Variable laser engraved images	Visual, Tilting	Laser-engraved images at different angles so that image view changes with tilting angle of viewing evidence.
Iridescent Inks and Custom Foil Stamping	Visual, Tilting	Custom designs and printing that will change color properties depending on the angle at which evidence is viewed.
Laser perforation	Visual, Light, Tactile	Perforated holes made by laser beam to form images. The images can be viewed under light source; image holes have tactile feel.
UV printing	Visual, UV Lighting	A UV image or text that can only be viewed with special lighting. UV images may appear on the front or back of the evidence.
Microprinting	Visual, Magnifier	Microtext of static or variable data that can be confirmed when viewed under a magnifier. Requires magnification of at least 10X to view.
Laser embossing	Tactile	Use of laser to emboss image or text for tactile feel on only one side of the evidence.
Barcode	Visual, Barcode Reader	Machine readable, encoded data (typically personalized printed data) for 2-D barcode, readable with barcode reader.
UV printing	Visual, UV Lighting	A UV image or text that can only be viewed with specialized lighting. UV images may appear on the front or back of a card.

SP 800-63A (5.2.2) also provides that the genuineness tests above for identity evidence validation may be performed through confirmation of cryptographic security features contained on the evidence in order to meet FAIR and STRONG validation strength; this is a requirement for SUPERIOR validation strength. Such cryptographic security features generally refer to cryptographically signed (e.g., digitally signed) data objects that are stored on an integrated circuit chip on the data evidence that can be used to compare and validate printed information on the evidence. The federal Personal Identity Verification

(PIV) Card is an example of this type of evidence. The cryptographically signed data objects on the chip can be used to confirm the personalized data, including facial image, printed on the evidence for evidence validation. Cryptographic security features require specialized equipment to access and validate cryptographically signed data objects on the evidence. Validation of the signed data objects requires verification of the digital signature on the signed data objects.

Unless identity evidence validation products and services as described above are used, CSP personnel will need to possess the capabilities to confirm correct information and format, detection of any tampering or counterfeiting, and presence and confirmation of security features for various types of identity evidence that may be presented by applicants. Due to the complexity of evidence validation, SP 800-63A (5.2.2) requires training for CSP personnel that are responsible for evidence validation:

> Training requirements for personnel validating evidence SHALL be based on the policies, guidelines, or requirements of the CSP or RP.

CSPs should determine the types and scopes of various types of evidence that may need to be validated and adjust training requirements to address those types of evidence as well as the policies and procedures that are established for the presentation and validation of identity evidence.

Most of the capabilities to confirm security features on identity evidence are dependent upon physically viewing the evidence directly, tactile feel of the evidence, and viewing the evidence under specialized lighting or through the use of specialized equipment. Therefore, the validation of evidence that may be submitted remotely for remote identity proofing methods is particularly challenging. For this reason, CSPs opting to provide remote identity proofing may find it most effective to use automated evidence validation products and services as described above which are permitted as "appropriate technologies" for evidence validation in SP 800-63A section 5.2.2. If such validation services are not used, operator training for evidence validation will depend on the CSP policies, guidelines and requirements. For this reason, training requirements for evidence validation requirements are not specified in SP 800-63A. Such training is especially important for CSPs that provide for IAL2 remote identity proofing or IAL3 supervised remote identity proofing. For these remote identity proofing methods, images of identity evidence are submitted remotely, but the capabilities for evidence validation are very limited as seen in the table of common types of security features presented above. If automated evidence validation solutions are not used, CSPs may choose to apply similar procedures for IAL2 remote proofing as are required for IAL3 supervised remote proofing. These procedures provide that a trained operator can remotely supervise the evidence collection process, require the applicant to turn or tilt evidence or apply lighting to be able to confirm security features on evidence that is presented for the identity proofing

encounter in a recorded video or webcast. Alternatively, a CSP may use an automated interface for the capture of identity evidence images that similarly can direct the applicant to turn, tilt or provide lighting on evidence presented for identity proofing purposes. Therefore, the training for personnel involved in the validation of evidence for remote proofing methods will depend on the CSPs' policies and procedures. Regardless, the confirmation of genuineness of identity evidence presented to support the claimed identity for identity proofing is critical and necessary for identity validation.

A.4.2 Evidence Information Validation

The second step in identity validation is to validate the correctness of information from the identity evidence against the issuing source for the evidence or an authoritative source that has linkage to the issuing source. This step applies to evidence validation at the STRONG and SUPERIOR Strengths (5.2.2):

> All personal details and evidence details have been confirmed as valid by comparison with information held or published by the issuing source or authoritative source(s).

It should be noted that the validation of all personal details and evidence details may not be possible for some types of common identity evidence. For example, state motor vehicle departments and driver's license verification services can typically verify issuing state and license number but may only be able to validate selected personal and document information from the license. Therefore, the CSP may not be able to validate all personal details and evidence information on the evidence but must validate all information that can be validated with the issuing or authoritative sources.

The results of identity evidence information validation and evidence genuineness validation should be recorded in enrollment records or audit logs as appropriate for the CSP.

A.5 Identity Verification

Identity verification represents the processes of confirming that the evidence, previously shown to be valid, actually refers to the applicant that is appearing for identity proofing. The objective of identity verification is to confirm a linkage between the validated evidence for the claimed identity and the physical, live existence of the person presenting the evidence. For IAL2 and IAL3 this binding is done by a physical or biometric comparison of the photograph on the strongest piece of evidence to the applicant or by a biometric comparison between information on the evidence and a biometric characteristic obtained from the applicant.

The following table presents verification methods that may be applied to achieve a verification level of strength of fair and higher. The requirements for these levels are presented in Table 5-3 in SP 800-63A (5.3.1). It should be noted that identity verification is performed against the strongest piece of identity evidence submitted and validated. For IAL2 and IAL3 the strongest piece of evidence will always be either STRONG or SUPERIOR evidence; therefore, verification of FAIR evidence binding will never be required. The KBV method for FAIR evidence verification is presented in the table below for information and use as additional binding strength as determined appropriate by the CSP.

Verification Strength	Verification Method	Description
SUPERIOR	Biometric Verification	Biometric comparison against biometric characteristics on the strongest piece(s) of evidence against live biometric capture for remote or in-person identity proofing. May be used for identity verification for FAIR, STRONG, and SUPERIOR strength.
STRONG	In-Person Physical Verification	Physical comparison of applicant to facial-image photograph on strongest piece(s) of validated evidence. May be used for identity verification for FAIR and STRONG strength.
STRONG	Remote Physical Verification	Physical comparison of applicant to facial-image photograph on strongest piece(s) of validated evidence. May be used for identity verification for FAIR and STRONG strength.
FAIR	Knowledge-Based Verification (KBV)	Comparison of challenge response to KBV questions provided by applicant. May be used for identity verification for FAIR strength only.

Table A-5-1. Verification Methods and Strengths

As indicated in the table above and SP 800-63A Table 5-3 (5.3.1), physical or biometric comparison is required for STRONG verification strength and biometric comparison is required for SUPERIOR verification strength against the strongest piece of validated identity evidence.

Physical comparison is a comparison by a person (i.e., CSP-trained personnel) of the applicant to the photograph (i.e., facial image) on any of the strongest piece(s) of validated identity evidence collected. This comparison can be an in-person comparison for in-person identity proofing processes or may be conducted remotely for remote identity proofing. In both cases, the operator must perform a physical comparison of the applicant to the facial image photograph on the evidence. That is, the in-person proofing personnel will physically compare the facial image of the live applicant to the photograph of facial image on the strongest piece of validated evidence. For remote physical comparison, the applicants' facial image may be captured by high resolution video or camera for physical comparison to the facial image photograph on the identity evidence. For remote facial image capture, the requirements of SP 800-63B, section 5.2.3. shall be applied and the methods for remote facial image collection and comparison are discussed below.

Biometric comparison is an automated comparison of a biometric characteristic (e.g., facial image, fingerprint, iris) collected and recorded as a reference to a live capture of the same biometric characteristic for comparison. For identity proofing verification, a biometric characteristic recorded on the strongest piece of identity evidence is compared to the corresponding biometric characteristic of the applicant captured live during the identity proofing session. For in-person biometric collection and comparison, the CSP must employ capabilities for biometric capture and comparison during the in-person session. Since most STRONG and SUPERIOR evidence contains a photographic image (i.e., facial image) on the evidence, the most common form of biometric collection for in-person proofing and biometric comparison will be facial image biometric matching. Automated biometric system matching capability must meet the requirements presented in SP 800-63B section 5.2.3. Biometric comparison is required for identity verification at SUPERIOR strength, which is required at IAL3.

For IAL2 remote identity proofing processes, either physical comparison or biometric comparison may be performed for identity verification based on the strongest piece of validated identity evidence. Unlike the in-person verification method described above, remote identity proofing requires the collection of both an image of the identity evidence and a live capture of the facial image of the applicant for physical or biometric comparison. The CSP must employ liveness detection capabilities to ensure that the applicant's facial image used for comparison is "live" and not a spoofing or presentation attack. There are considerable risks of impersonation, presentation and spoofing attacks without mitigating controls to ensure live capture of the applicants' facial image. Potential methods for the determination of live facial image capture for remote proofing involve

supervision by trained personnel and automated capabilities for liveness detection as presented below.

- A remote operator supervises the identity proofing session (similar to the processes of supervised remote identity proofing (5.3.3.2)) and may conduct a real-time physical comparison between the image of the identity evidence and a live video of the applicant. In order to confirm the video stream is live and not pre-recorded, the operator may direct the applicant to move their head in specific ways, raise or lower eyes, or ask the applicant questions requiring response during the live capture video. Once a positive confirmation is recorded from the operator, and all other requirements are met, the identity verification may be completed in a single session.
- The CSP employs automated capabilities which are specifically designed to compare the image of the identity evidence with the applicant and also employ liveness detection technologies. Pending a positive confirmation from the automated comparison, and the satisfaction of all other requirements, identity verification can be completed in a single session.
- The CSP employs liveness detection technology during the capture of the facial image and an off-line operator performs the physical comparison of images captured during the identity proofing session. The identity proofing process may require more than one session with the applicant and is not completed until the operator provides a positive confirmation of the comparison and the other requirements are met.

It is noted that liveness detection is a necessary control whether the identity verification is performed through physical comparison of the live capture of the applicants' facial image to the photograph on the strongest piece of identity evidence or through automated biometric facial image comparison.

A.6 Enrollment Codes

The use of an enrollment code for address confirmation is a requirement for IAL2 remote identity proofing and enrollment. Enrollment codes are not used for address confirmation for in-person identity proofing and enrollment but may be used for authenticator binding if one or more authenticators were not registered to the subscriber's account at the time of in-person identity proofing. This is discussed in more detail below. For either enrollment code use case – IAL2 remote identity proofing address confirmation or in-person proofing authenticator binding – enrollment codes must meet specified entropy requirements (4.6). Enrollment codes must be comprised of:

- a random six-character alphanumeric or equivalent entropy; or
- a machine-readable optical label, such as a QR Code, that contains data of similar or higher entropy as a random six character alphanumeric.

For IAL2 remote identity proofing address confirmation, the enrollment code may be sent to any address that was validated in the identity evidence validation step of identity proofing – postal, email, or telephone/SMS addresses. Enrollment codes used for address confirmation have specified validity periods depending on the type of address where the enrollment code is sent:

- 10 days, when sent to a postal address of record within the contiguous United States;
- 30 days, when sent to a postal address of record outside the contiguous United States;
- 10 minutes, when sent to a telephone of record (SMS or voice);
- 24 hours, when sent to an email address of record;

The IAL2 remote identity proofing and enrollment process is not complete until the applicant provides confirmation of the enrollment code within the specified validity period – through confirmation of the enrollment code or scanning and confirmation of the optical label/QR code.

Enrollment codes may also be used for in-person proofing and enrollment processes if an authenticator(s) is not registered to the subscribers' account at the time of in-person identity proofing and, therefore, the authenticator binding would need to occur at a later time. Enrollment codes may be used for authenticator binding to subscribers' accounts in such circumstances. Enrollment codes used for this purpose must meet the entropy requirements presented above and have a maximum validity period of 7 days. It is intended that enrollment codes used for this purpose would be provided to the applicant during the in-person proofing session and would not be mailed to the validated address of record.

A.7 Biometrics Collection

SP 800-63A presents two use cases for the collection of biometrics for purposes of identity proofing and enrollment: biometric matching of biometric data objects contained on identity evidence for the purpose of identity verification; and enrollment and registration of biometric characteristics as an authentication factor in the subscribers' enrollment account for purposes of account recovery and non-repudiation.

Biometric matching of biometric data objects contained on identity evidence for identity verification may be performed to provide binding of the evidence to the applicant for FAIR, STRONG, and SUPERIOR evidence strengths. Biometric collection for this purpose may be performed for in-person or remote identity proofing processes. Biometric matching is one of the optional methods for identity verification of the binding of the applicant to the evidence for FAIR and STRONG evidence verification; biometric matching is required for SUPERIOR verification binding. Therefore, biometrics collection is required for biometrics matching for SUPERIOR evidence verification binding and may be performed for binding FAIR and STRONG evidence whether the identity proofing process is in-person or remote.

Biometrics collection for enrollment and registration of biometric characteristics as an authentication factor in the subscriber's enrollment account for purposes of account recovery and non-repudiation is a requirement for IAL3 enrollment; this is optional for IAL2 account enrollment whether identity proofing is performed in-person or through remote processes.

In-person identity proofing biometrics collection requirements are presented in SP 800-63A section 5.3.3.1. These requirements provide controls against impersonation, presentation, and spoofing attacks. These requirements are also applicable to supervised remote identity proofing processes for IAL3 in-person identity proofing comparability. The in-person biometrics collection requirements of section 5.3.3.1 apply for both use cases described above. While it is envisioned that biometrics collection for remote identity proofing and enrollment for either use cases would principally involve facial image capture, biometrics collection for remote identity proofing and enrollment can be performed for any biometric modality. Remote identity proofing biometrics collection of any modality requires controls against impersonation, presentation, and spoofing attacks. The controls for supervised remote identity proofing presented in SP 800-63A section 5.3.3.2 allow the remote operator to view the applicant for the entire proofing session and inspect the biometric source (e.g., facial image, fingerprint) to detect attempts at spoofing or presentation attack.

A.8 Supervised Remote Identity Proofing

SP 800-63A section 5.3.3.2 provides for supervised remote identity proofing. Supervised remote identity proofing is intended to provide controls for comparable levels of confidence and security to the in-person identity proofing process for identity proofing processes that are performed remotely. Supervised remote identity proofing is optional for CSPs; that is, if a CSP chooses to use supervised remote identity proofing, then the requirements of section 5.3.3.2 would apply. It should be noted that the term "supervised remote identity proofing" has specialized meaning in SP 800-63A and is used only to refer to the specialized equipment and controls required in section 5.3.3.2.

Supervised remote identity proofing involves the use of a CSP-controlled station at a remote location that is connected to a trained operator at a central location. The goal of this arrangement is to permit identity proofing of individuals in remote locations where it is not practical for them to travel to the CSP for in-person identity proofing. Supervised remote identity proofing may also be used for achieving comparability with in-person requirements when face-to-face (i.e., in-person) encounters may present health risks to the applicant, CSP personnel or both. This may be necessary due to circumstances such as the COVID-19 pandemic where face-to-face encounters may present health risks. In these circumstances, supervised remote identity proofing may be used in a common facility where the applicant and CSP are in different locations in the facility but not actually interacting face-to-face. In such circumstances supervised remote identity proofing processing may be used.

Supervised remote identity proofing processes take advantage of improvements in sensor technology (cameras and biometric sensors) and communications bandwidth to closely duplicate the security of in-person identity proofing, which has been the requirement for high-assurance identity proofing in the past. This can be done through the use of a remote identity proofing station (or kiosk) which is under the control of the CSP or a third party that is trusted by the CSP to maintain its integrity.

The integrity of supervised remote identity proofing depends upon the applicant being continuously present and observed by the CSP operator during the entire session. An applicant who steps away from an in-process session may do so to alter their biometric source or substitute a different person to complete the identity proofing process.

The camera(s) a CSP employs to monitor the actions taken by a remote applicant during the identity proofing session should be positioned in such a way that the upper body, hands, and face of the applicant are visible at all times. Additionally, the components of the remote identity proofing station (including such things as keyboard, fingerprint capture device, signature pad, and scanner, as applicable) should be arranged such that all interactions with these devices is within the field of view. This may require more than one camera to view both the applicant and the room itself.

Technologies exist that allow for the digital validation of identity evidence via electronic means (such as RFID to read the data directly from e-passports and chip readers for smartcards). The scanners and sensors employed to access these features should be integrated into the remote identity proofing stations in order to reduce the likelihood of being tampered with, removed, or replaced. To be integrated means the devices themselves are a component of the workstation (i.e., smartcard readers or fingerprint sensors built into a laptop) or the devices, and their connections, are secured in a protective case or locked box.

For example, a kiosk located in a restricted area or one where it is monitored by a trusted individual requires less tamper detection than one that is located in a semi-public area such as a shopping mall concourse. (5.3.3.2 #6)

Requirements for protection and integrity of the kiosk depend on the specific kiosk capabilities (e.g., anti-tamper features). In most (perhaps all) cases, the kiosk will be overseen by a human attendant that can supplement the security features and protect the integrity of the kiosk. Between the attendant and the kiosk, the forms of protection provided may include (but are not limited to):

- Ensuring that only a single individual (applicant) interacts with the kiosk during any identity proofing session;
- Ensuring that the physical integrity of the kiosk and its sensors is maintained at all times;
- Verifying that the applicant is not using any devices to spoof biometric sensors (finger covers, for example); and
- Reporting any problems with the kiosk to the CSP

Supervised remote identity proofing stations/kiosks are required to employ mutual authentication where both the station/kiosk and server authenticate to each other. This is most often accomplished through the use of mutual TLS. Upon successful mutual authentication, an encrypted communication channel is established between the workstation/kiosk and the server which protects the data exchanged between them.

A.9 Use of Trusted Referees

SP 800-63A section 5.3.4 provides for the use of Trusted Referees in identity proofing and enrollment processes. The use of trusted referees is optional for CSPs; that is, if a CSP chooses to allow the use of trusted referees for identity proofing and enrollment, then the requirements of section 5.3.4 would apply. The use of trusted referees is intended to assist in the identity proofing and enrollment for populations that are unable to meet IAL2 and IAL3 identity proofing requirements or otherwise would be challenged to perform identity proofing and enrollment process requirements. Such populations include, but are not limited to:

- disabled individuals,
- elderly individuals,
- homeless individuals,
- individuals with little or no access to online services or computing devices,
- unbanked and individuals with little or no credit history,
- victims of identity theft,
- children under 18, and
- immigrants.

SP 800-63A section 5.3.4 intentionally avoids presenting overly prescriptive requirements in order to allow CSPs flexibility in establishing processes for trusted referees that can best meet the needs, use cases, and operational environment for the target populations. CSPs are required to establish written documentation of the policies and procedures for the use of trusted referees, both for the determination that such policies and procedures can meet applicable SP 800-63A IAL2 and IAL3 requirements and so that the use of trusted referees can be understood to external entities. Such CSP documentation for the use of trusted referees may include:

- types of trusted referees permitted,
- use(s) of referees,
- trusted referee enrollment procedures,
- identity proofing processes for trusted referees and the applicants they represent,
- trusted referee relationship to applicants,
- procures for recording trusted referees in enrollment records and logs,
- contact and communication procedures for trusted referees and the applicants they represent.

Trusted referees may be notaries, legal guardians, medical professionals, conservators, persons with power of attorney, or other qualified individuals that may act on behalf of or otherwise represent the applicant.

A.10 IAL2 Remote Identity Proofing

> Note: This section of the SP 800-63A Implementation Resources repeats some of the text of other sections of the Implementation Resources as it is anticipated that this section for IAL2 remote identity proofing may be used as a stand-alone resource.

Identity proofing of applicants without requiring them to physically meet in person with CSP personnel is an important but challenging capability. It is important in providing access to CSP services to a larger portion of the population and in reducing the costs to both the applicant and the CSP. It is challenging because many of the identity proofing methods available to the CSP in a face-to-face interaction, such as detailed inspection of evidence documents, are difficult to perform with comparable security when conducted remotely. The requirements in NIST SP 800-63A for remote identity proofing attempt to strike a pragmatic balance between availability and convenient access to identity proofing services and security of the associated processes.

There are two methods of remote identity proofing that are defined in SP 800-63A.

- Conventional remote identity proofing represents the processes and controls for CSPs to identity proof and enroll applicants remotely at IAL2.
- Supervised remote identity proofing represents the processes and controls for CSPs to provide comparable levels of confidence and security to in-person IAL3 identity proofing for identity proofing processes that are performed remotely. Supervised remote identity proofing requires the use of specialized equipment under the CSPs' control that is deployed to a remote location and specific controls and specialized requirements for comparability to in-person IAL3 proofing processes. Detailed guidance for supervised remote identity proofing is provided in a separate section of the Implementation Resources.

Note that "unsupervised" (conventional) remote identity proofing is not intended to imply the lack of supervision for the identity proofing process, but rather that the specific requirements of supervised remote identity proofing for IAL3 are not required.

Conventional remote identity proofing, which may be used at IAL2, generally involves the applicant (the person undergoing identity proofing) using their own hardware to complete the proofing process. This will typically involve the use of a camera to capture images of the applicant and the evidence they are presenting. When available, devices such as scanners may be used to capture higher-resolution images of the evidence being presented. However, the use of separate devices like scanners may make it more difficult to securely associate the image of the captured evidence with the primary (webcam) session.

A.10.1 Identity Resolution

Identity proofing begins with the resolution process. The applicant provides attribute information (e.g., name, physical address, date of birth, email address, phone number) to the CSP and one to three forms of identity evidence. In rare cases, the attribute information provided may not resolve to a unique individual; if this is the case, additional attributes may be requested to resolve the ambiguity. If necessary, ambiguities can be resolved through the use of knowledge-based verification (KBV).

A.10.1.1 Identity Evidence Collection

Several combinations of evidence quality are accepted at IAL2 as shown in the table below.

IAL2

- One piece of SUPERIOR or STRONG evidence depending on strength of original proof and validation occurs with the issuing source, or
- Two pieces of STRONG evidence, or
- One piece of STRONG evidence plus two (2) pieces of FAIR evidence

A single piece of SUPERIOR or STRONG evidence can be used for identity proofing at IAL2 if the evidence itself was issued pursuant to a sufficiently strong identity proofing process and if the CSP validates the evidence directly with the issuing source. See the Notional Strength of Evidence table in the Strength of Evidence section of these Implementation Resources. STRONG evidence types that may be considered to meet this level of quality are presented as STRONG+ in that table.

Additional evidence strength combinations at IAL2 are: two pieces of STRONG evidence. or a single piece of STRONG evidence along with two pieces of FAIR evidence.

A.10.2 Identity Validation

The objective of identity validation is to determine the authenticity, integrity and accuracy of identity evidence collected from the applicant to support the claimed identity for identity proofing. Identity validation is made up of two process steps: confirming the evidence is authentic and confirming that the data on the identity evidence is valid, current, and related to an actual, live individual.

Evidence validation for authenticity involves examining the evidence for:

- Confirmation of required information completeness and format for the identity evidence type.

- Detection of evidence tampering or the creation of counterfeit or fraudulent evidence.
- Confirmation of security features.

SP 800-63A Table 5-2 *Validating Identity Evidence* (5.2.2) presents validation techniques for 5 levels of validation strength, ranging from UNACCEPTABLE to SUPERIOR. One of the validation techniques that may be used for evidence validation at FAIR, STRONG, and SUPERIOR strength is to confirm that the evidence is genuine using "appropriate technologies". In this case, "appropriate technologies" refers to identity document validation products and services with the capability to perform one or more of the tests for authenticity listed below for the types of identity evidence presented. Such evidence validation products and services may be used for either in-person or remote identity proofing methods. Therefore, such products may be used when the identity evidence is physically presented for in-person proofing or submitted via video or images that are captured via scanner, webcam, or mobile phone camera for remote identity proofing. Such products and services should conduct one or more of the following necessary evidence authenticity tests:

- Test identity evidence for authenticity against document type libraries for information completeness, format, and correctness;
- Test identity evidence for authenticity through tamper and counterfeit detection; and
- Test identity evidence for authenticity by confirming presence and verification of security features for the type of evidence presented.

There are multiple commercial products that can perform these types of document validation capabilities at varying degrees of accuracy and reliability. If a single product cannot perform each of the three genuineness validation tests above, then other products should be used in combination to perform these validation tests or manual intervention and examination would be necessary. SP 800-63A Table 5-2 *Validating Identity Evidence* (5.2.2) could be interpreted that use of appropriate technologies as described above alone would be sufficient for evidence validation at validation strengths of FAIR and STRONG. However, in practice most document validation products require some degree of manual intervention to resolve data collisions and evidence conflicts in order to proceed with identity proofing processes. Manual intervention to resolve collisions and conflicts most likely would require trained personnel from the product vendor or CSP personnel, depending on the type of product or service and any associated service level agreements.

Manual validation of identity evidence, particularly the confirmation of integrity of physical security features of the evidence, is particularly challenging when done remotely without specialized equipment. For example, some security features involving the texture of a printed medium and verification of the existence and quality of microprinting may not be remotely verifiable. Others like holographic coatings and color-shifting inks may be dynamically verifiable on a live video connection between applicant and proofing agent.

Therefore, the validation of evidence that may be submitted remotely for remote identity proofing methods is particularly challenging. For this reason, CSPs opting to provide remote identity proofing may find it most effective to use automated evidence validation products and services as described above which are permitted as "appropriate technologies" for evidence validation in SP 800-63A section 5.2.2. If such validation services are not used, operator training for evidence validation will depend on the CSP policies, guidelines and requirements. For this reason, training requirements for evidence validation requirements are not specified in SP 800-63A. Such training is especially important for CSPs that provide for IAL2 remote identity proofing or IAL3 supervised remote identity proofing For these remote identity proofing methods, images of identity evidence are submitted remotely, but the capabilities for evidence validation are very limited as seen in the table of common types of security features presented in the Identity Validation section of these Implementation Resources. If automated evidence validation solutions are not used, CSPs may choose to apply similar procedures for IAL2 remote proofing as are required for IAL3 supervised remote proofing. These procedures provide that a trained operator can remotely supervise the evidence collection process, require the applicant to turn or tilt evidence or apply lighting to be able to confirm security features on evidence that is presented for the identity proofing encounter in a recorded video or webcast. Alternatively, a CSP may use an automated interface for the capture of identity evidence images that similarly can direct the applicant to turn, tilt or provide lighting on evidence presented for identity proofing purposes. Therefore, the training for personnel involved in the validation of evidence for remote proofing methods will depend on the CSPs' policies and procedures. Regardless, the confirmation of genuineness of identity evidence presented to support the claimed identity for identity proofing is critical and necessary for identity validation.

A.10.3 Evidence Information Validation

The second step in identity validation is to validate the correctness of information from the identity evidence against the issuing source for the evidence or an authoritative source that has linkage to the issuing source. This step applies to evidence validation at the STRONG and SUPERIOR Strengths (5.2.2): *All personal details and evidence details have been confirmed as valid by comparison with information held or published by the issuing source or authoritative source(s).* It should be noted that the validation of all personal details and evidence details may not be possible for some types of common identity evidence. For example, state motor vehicle departments and driver's license verification services can typically verify issuing state and license number but may only be able to validate selected personal and document information from the license. Therefore, the CSP may not be able to validate all personal details and evidence information on the evidence but must validate all information that can be validated with the issuing or authoritative sources.

one or more authenticators. A credential is established as a result of identity proofing and authenticator binding. The authoritative instance of a credential is a data structure that is securely maintained by the CSP.

In some cases—notably, with PIV cards—copies of subject (user) attributes are stored on an authenticator, in most cases cryptographically signed by the CSP or other authority. This is useful when it isn't possible to communicate with the CSP, e.g., in disaster situations. However, attributes can change so such copies, even if accompanied by valid signatures, might be considered less authoritative if they can't be verified online with the CSP.

It isn't possible to entirely avoid the usage of *credential* as a physical object held by the claimant. SP 800-63B attempts be consistent in its use of the term in the above-described way, rather than as a user-retained physical credential.

B.3 Authenticator Assurance Levels

The following sections provide some further description of the three authenticator assurance levels (AALs) and in particular how the authenticator combinations permitted at each AAL were arrived at. As with the rest of these implementation resources, these descriptions are informative; refer to SP 800-63B for normative guidelines.

Authenticator assurance levels are associated with interactive sessions and not with the authenticators themselves. This is because combinations of authenticators, used together, can achieve a higher AAL than individually. On the other hand, some requirements, such as reauthentication time, that are more stringent at higher AALs can limit a given session to a lower AAL than the authenticators themselves might be able to support. So while a multi-factor cryptographic authenticator might be characterized as AAL3-capable, that doesn't mean that any session it is used to authenticate is necessarily AAL3.

B.3.1 Authenticator Assurance Level 1

AAL1 permits single-factor authentication using a wide variety of authenticators listed in SP 800-63B Section 4.1.1. By far the most common authenticator at AAL1 is the memorized secret, but from the standpoint of meeting AAL1 requirements it is equally acceptable to use a physical authenticator such as an OTP device. Physical authenticators and memorized secrets are, of course, susceptible to different types of threats. When multifactor authenticators are used at AAL1, the nature of those devices requires that the additional factor (a memorized secret or biometric) be provided to allow those authenticators to operate.

Biometrics by themselves are not considered authenticators in SP 800-63B; they must always be strongly bound to a physical authenticator and are considered an activation factor for that authenticator. This mitigates the relatively high false acceptance rate for biometrics and the risks associated with disclosure and non-revocability of biometric data. For that reason, a biometric cannot be used alone for authentication, even at AAL1.

B.3.2 Authenticator Assurance Level 2

AAL2 requires the use of two authentication factors, either (1) a physical authenticator and a memorized secret, or (2) a physical authenticator and a biometric that has been associated with it. Multi-factor authentication can be performed using either a multi-factor authenticator or through the use of two independent authenticators.

As detailed below, there are restrictions on the use of biometrics, in particular that they must be securely bound to a specific physical authenticator. For this reason, a memorized secret plus a biometric is not an acceptable combination for authentication.

In addition to the requirement for two authentication factors at AAL2, there are additional requirements relating to the authentication and the session. These include:

- shorter reauthentication time,
- replay resistance,
- FIPS 140 Level 1 for authenticators supplied by government agencies, and
- authentication intent (recommended).

Multi-factor authenticators use an additional factor, either something you know or something you have, to unlock a secret that is stored in the (physical) authenticator.

B.3.3 Authenticator Assurance Level 3

AAL3 introduces several new requirements beyond AAL2, the most significant being the use of a hardware-based authenticator. There are several additional authentication characteristics that are required:

- verifier impersonation resistance,
- verifier compromise resistance, and
- authentication intent.

Some of these characteristics are satisfied jointly by the authenticator and verifier, while others are primarily authenticator characteristics. When multiple authenticators are used, these requirements are satisfied by the use of at least one authenticator with the required characteristic. For example, if a hardware-based authenticator that is not verifier impersonation resistant is used, a software-based authenticator that provides verifier impersonation resistance will satisfy that requirement.

B.3.3.1 Permitted Authenticator Types

SP 800-63B Section 4.3.1 identifies six combinations of authenticators that can meet the requirements of AAL3. There might be additional combinations that work, such as combinations of four or more authenticators to meet all of the AAL3 requirements, but these are unlikely to be used because of the complexity of the user experience.

Even though two authentication factors are required at AAL3, one combination of authenticators (Hardware Single-Factor OTP Device plus a Single-Factor Cryptographic Software Authenticator plus a Memorized Secret) consists of three authenticators. This combination stems from the fact that the hardware-based Single-Factor OTP Devices do not provide verifier impersonation resistance, so a Single-Factor Cryptographic Software Authenticator can satisfy that requirement. But since both of those authenticators are something you have, a Memorized Secret is required to satisfy the requirement for two different authentication factors.

Use of an authenticator or combination of authenticators on this list is not itself sufficient to meet the requirements of AAL3. For example, a multi-factor cryptographic device does not necessarily provide verifier impersonation resistance nor establish authentication intent. When an authentication system to meet AAL3 is designed, all of the AAL3 requirements need to be examined and satisfied, in addition to the choice of authenticator type(s).

B.3.4 Privacy Requirements

While the privacy requirements in SP 800-63B Section 4.4 are expressed primarily in wording that applies to federal agencies, the requirements are relevant for other uses of authentication as well. A key requirement is that data that is collected be limited to its intended use (authentication) unless the subscriber consents to additional use. Any such additional use must be voluntary, and not be a condition for the use of the service without a strong justification.

B.4 Authenticators and Verifiers

See SP 800-63B Section 5 for normative requirements.

B.4.1 Authenticator Types

There are nine recognized authenticator types.

Pre-registered knowledge tokens—sometimes referred to as security questions or knowledge-based authentication (KBA)—an authenticator (token) type that existed in SP 800-63-2, has been withdrawn in SP 800-63B because they often rely on information that is private but not secret. They also encourage the use of the same answers to authenticate on multiple sites, which is a problem if any of them is compromised. In addition, they often must be stored in an unhashed form, introducing a further vulnerability because the recalled answers may be approximate (e.g., "Central High" vs. "Central High School" or "Central HS"). The use of hints in prompts for memorized secrets has also been prohibited because of similar security concerns and the possible use of hints as a work-around to support security questions.

The single-factor cryptographic software authenticator, discussed in SP 800-63B Section 5.1.6, is a new authenticator type introduced in SP 800-63B.

B.4.1.2 Memorized Secrets

The memorized secret is by far the most common type of authenticator. It is also the only authenticator that is a *something you know* factor (pre-registered knowledge tokens were also something you know in SP 800-63-2 and earlier editions).

The term *memorized secret* was chosen as a single term encompassing passwords, passphrases, and PINs. The intent of a memorized secret is that it be potentially memorable to a subscriber, even if not chosen by the subscriber. This differentiates it from a key, which is never chosen by the subscriber, typically has at least 112 bits of entropy, and therefore is not expected to be memorized nor entered by the average subscriber.

One of the significant changes in SP 800-63B is a rethinking of the role of memorized secrets and minimization of their burden on subscribers. In accordance with Executive Order 13681, transactions involving any significant risk, including any which involve the release of personal information, require multi-factor authentication. As a result, memorized secrets will be used alone only when a low level of security is required.

Research has shown that there is a significant gap between the requirement for memorized secrets that must protect against an offline attack as compared with those that only protect against throttled online attacks. For memorized secrets to be considered secure against current offline attacks, a considerably higher minimum length would be required. Even so, there is no assurance that subscribers would pick memorized secrets that don't lend themselves to automated guessing attacks. Accordingly, a two-pronged approach was adopted:

- Set minimum memorized secret requirements to protect against online attacks only, accept the risk of offline attacks, and throttle online attempts.
- Require verifiers to implement secure hashing of memorized secrets, including iterated hashing with a salt, and recommend hashing with a secret value as well.

This puts the burden on the verifier, rather than the subscriber, to the maximum extent possible.

At the same time, SP 800-63B attempts to make it as easy as possible for a subscriber to choose a memorized secret that is as secure as possible. Because memorized secrets are required to be hashed before storage by the verifier, the length of the stored value is independent of the length of the memorized secret. There is no good reason, therefore, to prevent memorized secrets from being almost arbitrarily long, nor to prohibit the use of spaces and of certain special characters. Since non-English speakers might more readily memorize a secret in their own language, Unicode characters should also be permitted (not just to permit the creation of emoji passwords as some have suggested).

It is nevertheless desirable to provide some degree of protection against subscribers who choose frequently used memorized secrets. SP 800-63B requires the use of a blocklist to prevent subscribers from choosing such secrets.

No size is specified for the blocklist. While it might be tempting to use lists of millions of compromised passwords (such lists are readily available on the internet), it is really only the ones that are fairly commonly used, a much shorter list, that represent a significant risk of online attack. Excessively long lists are also likely to be frustrating to the subscriber, as are the composition rules (inclusion of specific character classes in memorized secrets) currently in common use. Bear in mind that common passwords are not just words, but sometimes typing patterns such as "qwertyuiop".

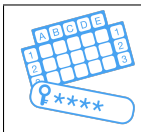
In addition to common memorized secrets perhaps obtained elsewhere, it is useful to include other things on the blocklist that might be relevant to the specific service being authenticated, such as the agency name or domain name. Items on the blocklist can also be common constructions including those terms.

When a subscriber attempts to choose a blocklisted memorized secret, it is helpful to give additional guidance to them. Measures like strength indicators (password meters) may encourage them not to choose a memorized secret that is a trivial modification of one on the blocklist.

B.4.1.2.1 Examples

As mentioned above, memorized secrets include passwords, passphrases, and PINs. The term passphrase is often used when the expectation is that the secret will be longer than a password, and when spaces may be included, but otherwise the terms are equivalent. PINs normally denote a numeric secret that is often randomly chosen by the CSP/verifier and assigned to the user. The length requirement for randomly chosen memorized secrets is shorter than for user-chosen secrets because they would be expected to be uniformly distributed and therefore have more entropy than a user-chosen secret of the same length and composition.

B.4.1.3 Look-up Secrets



Look-up secrets are secrets that are issued by the CSP to the subscriber each of which can be used for one successful authentication. They are considered *something you have*, the "something" being the printed or other media containing a set of these secrets. They are well suited for use as a backup authenticator to be used when a primary authenticator is lost, stolen, or malfunctions.

The primary disadvantage of look-up secrets is that they can only be used for a specific number of authentications, after which a new set of look-up secrets needs to be issued to the subscriber. However, they are among the lowest-cost authenticators to issue. Issuance of look-up secrets can occur in person (typically at the end of an in-person identity proofing session), via postal mail, or in a mutually-authenticated protected session where the subscriber authentication also included something you have.

Look-up secrets must, of course, be protected from disclosure. While storage requirements for look-up secrets are not specified in SP 800-63B, look-up secrets that are used as backup authenticators would normally be stored in a locked container on the subscriber's premises. Issuance of look-up secrets should be accompanied by suitable advice on protecting the secrets, as well as procedures for revoking the secrets should they be lost or stolen. As noted in SP 800-63B Section 5.1.2.1, look-up secrets may be issued online over a secure channel; this normally requires a mutually-authenticated session at AAL2 or higher. Non-secure mechanisms such as email are unsuitable for the distribution of look-up secrets.

In some cases, look-up secrets are issued in a form suitable for the subscriber to carry with them, e.g., in a wallet. While something carried in a wallet is probably more likely to be lost or stolen, that theft or loss is more likely to be detected quickly. Accordingly, issuers of look-up secret authenticators that are designed to be carried should have procedures in place to allow rapid reporting and revocation of authenticators that are no longer under the subscriber's control.

Look-up secrets need to be protected by the verifier as well. While Section 5.1.2.1 of SP 800-63B permits look-up secrets to have as little as 20 bits of entropy, their use as backup authenticators makes usability less of a concern and permits the use of look-up secrets with sufficient entropy to resist offline attacks. The use of high-entropy look-up secrets is highly encouraged.

B.4.1.3.1 Examples

The most common form of look-up secret authenticator is a printed list of secrets. These secrets are generated using an approved random bit generator, and may be expressed in any encoding that provides acceptable usability. This often includes grouping the secret into a number of sections to enhance its readability from the media on which it is delivered. One popular format is the version 4 (random) UUID, because of the wide support available for rendering UUID values.

One-time secrets can be used sequentially or a particular order specified by the verifier (e.g., "Enter OTP #4:"). This gives a bit of a challenge/response characteristic to the transaction. However, look-up secrets are required to be used only once, so "OTP #4" in this example would not be reused. This requirement is meant to ensure that an attacker with pervasive access to the authentication session (e.g., a key logger) would not be able to exploit the authenticator output in the future. The use of a specified order (verifier challenge) is acceptable but not required.

A third common example of a look-up secret authenticator is a secret grid. In this arrangement, the verifier gives the coordinates for squares in a grid under the control of the subscriber. Again, the grid squares can be used only once to meet SP 800-63B requirements. This also has the disadvantage that it is difficult to store the values in the individual squares securely: if the squares contain short values, hashed values stored by the verifier would be easily dictionary attacked. This form of look-up secret authenticator, while permissible, does not have any particular advantages and, being more susceptible to dictionary attacks, is discouraged.

B.4.1.4 Out-of-Band Devices

Out-of-band authenticators use a private communication channel that is separate from the channel being authenticated to establish the claimant's control of a specific physical device. An out-of-band authenticator is *something you have*.

While there are many different implementations of out-of-band authenticators, it is important to remember that the primary objective is to establish that the claimant controls

a specific device associated with the subscriber—that the claimant and subscriber are the same person. To the extent that devices can be substituted without re-enrollment or more than one device can be used for a given out-of-band authentication, the authenticator is weaker, or in some cases unsuitable for use. Accordingly, (1) email services and (2) telephony that terminates in a voice-over-IP (VoIP) endpoint are not acceptable for out-of-band authentication because these often can be received by more than one endpoint. If the registration of an out-of-band device is rejected because it is a VoIP endpoint, it is helpful to explain the rationale for this to the subscriber.

It is also important to ensure that the activity on the out-of-band device be associated with a specific session on the primary channel. The transfer or verification of a secret between the primary and secondary channels avoids the opportunity for an attacker with good timing to obtain authentication of a different session controlled by them.

By far, the most common authentication flow for out-of-band authenticators is for the relying party to send a secret to the subscriber's device via a secondary channel and request that value be returned over the primary channel. However, at least two other out-of-band authentication flows are possible. As described in SP 800-63B Section 5.1.3.1, the secret can also be output via the primary channel and returned over the secondary channel. This may permit the use of a QR code or similar mechanism to transfer the secret to the out-of-band device (often a mobile smartphone), potentially improving usability. The primary channel and out-of-band device may also display a secret and prompt the claimant to compare the consistency of those secrets to ensure that the claimant is authenticating the correct session. The verifier then accepts a yes/no response from the out-of-band device. This is a less effective method, because it depends on the subscriber actually making the comparison and not just selecting "yes".

In all of these situations, it is important that the out-of-band device be securely and uniquely authenticated. This requires the use of a secret in the device, perhaps in the form of a client certificate or, if using the telephone network, a SIM card.

It should be noted that the authenticator is not required to be a physically separate device from that on which the authentication is occurring. The requirement for separation applies to the communication channel, not the device. Therefore, it is permissible to authenticate a browser session on a mobile device using an application resident on the same device, provided they use separate communication channels (e.g., TLS sessions) and provided that the application uniquely identifies itself to the verifier.

B.4.1.4.1 Examples

The most common example of an out-of-band device is also a restricted authenticator: the use of SMS to send a random secret to the subscriber's mobile telephone. Many security weaknesses with this have been identified, including SS7 (telephone signaling)

vulnerabilities and the possibility of the telephone number being reassigned to a different device, perhaps by social engineering of a carrier or retail representative by an attacker.

Some secure communications apps, such as Signal, create a fingerprint that changes if the device on which the app is running ever changes. This allows the verifier to securely detect a change in endpoint. The combination of strong device binding and the end-to-end encryption of the secondary channel permits the authentication secret to prove possession of a specific device, making this a fully acceptable alternative to the use of SMS.

A verifier-specific application can also be used to terminate the user side of the secondary communication channel. This application would need to maintain a secret (probably a key pair) that it uses to authenticate to the verifier in accepting or providing the secret being exchanged.

B.4.1.5 Single-Factor OTP Device

A single-factor OTP device is something that is in the possession of the subscriber that generates one-time passwords that are displayed and manually entered by the claimant. Even though it is referred to as a "device", this authenticator can be either a distinct physical device or a software application running on a general-purpose device such as a smartphone. A single-factor OTP device is *something you have*.

Single-factor OTP devices that are not time-based usually operate based on the pressing of a button to obtain a single one-time password. While it is important that a one-time password be accepted only once, non-time-based devices might be operated by mistake, as a test, or in a session that authenticates unsuccessfully due to a communications error. Accordingly, the verifier should accept any of several possible future one-time passwords, and advance its state to the authenticator output most recently used when a successful authentication is performed.

Time-based OTP devices maintain an internal clock that must be kept in relatively close synchronization with the verifier. This can be difficult because the OTP device, under control of the subscriber, may be expected to operate for several years despite being subjected to temperature changes and other environmental factors that contribute to clock drift. The verifier needs to consider possible clock drift in its determination whether to accept a given OTP value. These devices are usually shipped from manufacturers with their clocks pre-synchronized, and the manufacturer may provide a verification service for their use. As in any case when authentication is outsourced, verifiers need to consider the security practices of the manufacturer when assessing overall misauthentication risk.

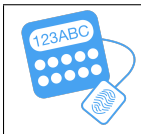
Unlike earlier editions of SP 800-63, SP 800-63B treats devices that are connected directly to the endpoint as crypto devices rather than as OTP devices, even if they only

supply a one-time password. The authenticator output for OTP devices is defined to be manually transferred from the OTP device to the application being authenticated. For this reason, OTP devices are never considered verifier-impersonation resistant as described in SP 800-63B Section 5.2.5. The goal of verifier-impersonation resistance is to not depend on the claimant detecting a phishing attack, and an OTP authenticator cannot control where its output is entered.

B.4.1.5.1 Examples

A number of readily-available commercial OTP products, both hardware and software, are available on the market. The Initiative for Open Authentication (OATH) is an industry consortium promoting the use of OTP authenticators.

B.4.1.6 Multi-Factor OTP Devices



Multi-Factor OTP Devices are similar to Single-Factor OTP devices, but require activation by input of a memorized secret or the successful presentation of a biometric in order to obtain a one-time password. A multi-factor OTP device is *something you have* and is activated by *something you know* or *something you are*.

Many of the same considerations associated with single-factor OTP devices apply to these authenticators as well.

When the wrong memorized secret is entered, the authenticator can take one of two actions. One is to generate an intentionally incorrect output; this allows the verifier to implement a throttling strategy to discourage guessing attacks on the memorized secret. Another possibility is to display an error indication on the device. This avoids the usability impact if the user mis-enters the secret, but requires that the authenticator implement the throttling strategy described in SP 800-63B section 5.2.2, which may be challenging on some devices.

Because of the significant false reject rates associated with biometrics, the generation of an intentionally incorrect output is likely to have a greater impact on devices activated by a biometric. In using biometric-activated OTP devices, the severe throttling requirements described in SP 800-63B Section 5.2.3 should be considered, and alternatives provided if the user is unable to successfully complete biometric authentication. These alternatives could include the use of a memorized secret for activation, or use of a completely different authenticator.

B.4.1.7 Single-Factor Cryptographic Software

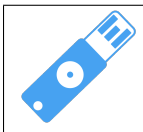
A single-factor cryptographic software authenticator is a secret cryptographic key and associated software stored on a software-accessible medium. Authentication is accomplished by proving possession of the embedded key. A single-factor cryptographic software authenticator is *something you have*.

The characteristics of cryptographic authenticators depend on the method by which the authenticator output is generated. One such method is the generation of a one-time password; this is different from an OTP device because the authenticator output is directly supplied to the application by the authenticator. This makes a larger (higher entropy) authenticator output practical, but does not provide the additional security benefits of a challenge-response protocol.

B.4.1.7.1 Examples

The classic example of a single-factor cryptographic software authenticator is the use of a client X.509 (TLS) certificate. The certificate (signed public key) is accompanied by a private key that is held securely by the subscriber. The verifier needs to have some basis for associating the public key with the subscriber. This may be accomplished by a certificate that is signed by a certificate authority accepted by the verifier (in some cases, by the verifier itself) associating the certificate's common name with the subscriber. Alternatively, the verifier may directly associate the certificate's public key with the subscriber. Because the verifier only needs to associate specific certificates with subscribers, the use of generally-recognized root certificate authorities is often not required.

B.4.1.8 Single-Factor Cryptographic Devices



Single-factor cryptographic devices are similar to single-factor cryptographic software authenticators, except that the private key is contained within a hardware device and cannot be exported in normal operation. This means that the hardware device also performs the cryptographic operations associated with authentication. A single-factor cryptographic device is *something you have*.

As with cryptographic software authenticators, cryptographic device authenticators have capabilities that range from one-time password generation (not challenge-response, and not verifier-impersonation resistant) to others having many of the supplementary characteristics described in Section 5.2.

B.4.1.8.1 Examples

Single-factor cryptographic devices exist in a wide range of shapes and sizes. "Smart cards" with an embedded processor in a credit card form factor are quite popular, and may be read either via a dedicated device associated with the endpoint or through a USB adapter. Other devices, notably FIDO U2F authenticators, have direct USB interfaces and may be designed to be kept on a subscriber's (physical) keychain or for semi-permanent installation in an endpoint such as a laptop computer.

Single-factor cryptographic devices may also be embedded in a user endpoint, such as in a hardware TPM in a user device. Cryptographic devices with wireless interfaces, particularly NFC, are also emerging and may prove popular, particularly for mobile devices that may lack USB and similar hardware interfaces.

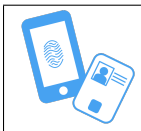
Some single-factor cryptographic devices operate in more than one mode, and it is important to consider the capabilities of the particular mode being used. For example, some authenticators that implement FIDO U2F (a challenge-response protocol that may be verifier impersonation resistant) also implement a legacy one-time password mode, which is not verifier impersonation resistant.

B.4.1.9 Multi-Factor Cryptographic Software

Multi-factor cryptographic software authenticators are similar to single-factor cryptographic software authenticators except that they require the input of a memorized secret in order to access the private key for authentication. Multi-factor cryptographic software authenticators are *something you have* and are activated by *something you know*.

One of the operational problems associated with multi-factor cryptographic software authenticators is in determining whether a multi-factor authentication has in fact taken place. Since the encrypted private key is available to the subscriber's software, a non-cooperative subscriber could decrypt and store the key, degrading authentication to single-factor (but less effort for the subscriber) without the verifier's knowledge or consent. Since there is less opportunity to extract and decrypt the private keys on some platforms (particularly some mobile devices), these authenticators are more certain to be effective on these than on general-purpose devices.

B.4.1.10 Multi-Factor Cryptographic Devices



Multi-factor cryptographic device authenticators are similar to single-factor cryptographic device authenticators except that they require activation by the entry of a memorized secret or verification of a biometric. Multi-factor cryptographic device authenticators are *something you have* and are activated by either *something you know* or *something you are*.

Since the private key (authentication secret) associated with the device is embedded in a hardware device with security requirements (depending on the AAL at which it is used), activation of the authenticator can cause decryption of the secret key, as in the case of a multi-factor cryptographic software authenticator. It can also simply make the key available to an authentication operation. The latter is the mode in which biometric activation usually operates.

The activation factor can be provided to the authenticator directly (i.e., by keyboard input or biometric sensor directly on the device). More frequently, the activation factor is provided by the host endpoint; this requires additional trust in the endpoint, e.g., to make sure that a keylogger is not installed or that a biometric sensor is not being spoofed. If the biometric sensor or endpoint is separate from the authenticator, the sensor or endpoint needs to be authenticated as described in SP 800-63 Section 5.2.3.

The activation factor, either a memorized secret or biometric, is subject to throttling on repeated unsuccessful attempts as described in SP 800-63B Section 5.2.2. The state information for this throttling can be kept by the authenticator or unsuccessful authentication attempts can be indicated to the verifier, which would then limit the number of attempts permitted.

As with other cryptographic authenticators, a range of capabilities is possible, including generation of one-time passwords and challenge-response. The primary distinguishing factor between a cryptographic device and an OTP device is that the former is directly connected to the endpoint and the latter requires manual entry of the authenticator output by the claimant.

B.4.1.10.1 Examples

The classic examples of multi-factor cryptographic authenticators are US Government PIV and Department of Defense CAC authenticators. Other “smart card” authenticators, such as the Estonian e-resident card, are also in this category. Multi-factor cryptographic devices can also be embedded in endpoints, as is the case of FIDO UAF authenticators.

B.4.2 General Authenticator Requirements

The subsections of Section 5.2 describe requirements applicable to multiple classes of authenticators, or in some cases supplemental requirements applicable at higher AALs. These are summarized in the tables below.

	Rate Limiting	Biometrics	Attestation	Intent
Memorized Secret	Required	N/A	N/A	Yes
Look-up Secret	Required if <64 bits	N/A	N/A	Yes
OOB	Not required	N/A	N/A	Yes
SF OTP	Required	N/A	N/A	Yes
MF OTP	Required	N/A	Offline	Yes
SF Crypto SW	Not required	N/A	N/A	Maybe
SF Crypto Dev	Not required	N/A	Issuance or certificate	Maybe
MF Crypto SW	Required for activation	N/A	Offline, procedures	Yes[4]
MF Crypto Dev	Required for activation	Required for biometric activation	Issuance or certificate	Yes[5]

Table B-4-1. General Authenticator Requirements (1)

	Verifier Impersonation Resistance	Verifier Compromise Resistance	Replay Resistance
Memorized Secret	No	No	No
Look-up Secret	No	Maybe	Yes
OOB	No	Yes	Yes
SF OTP	No	No	Yes
MF OTP	No	No	Yes
SF Crypto SW	Maybe	Maybe	Yes
SF Crypto Dev	Maybe	Maybe	Yes
MF Crypto SW	Maybe	Maybe	Yes
MF Crypto Dev	Maybe	Maybe	Yes

Table B-4-2. General Authenticator Requirements (2)

B.4.2.1 Physical Authenticators

This section addresses the need for physical authenticators (any authenticator that includes something you have) to be protected against theft and loss. The CSP needs to establish procedures to handle these situations, and needs to ensure that the subscriber knows what to do (how to report the event) when they occur.

In order to avoid denial-of-service attacks on subscribers, the CSP needs to identify the subscriber when accepting such a report. Typically, the cost associated with erroneously suspending a subscriber is lower than that associated with use of a stolen authenticator, so this identification can often be weaker than would be acceptable for authentication. However, in certain cases erroneous suspension could be very serious, so procedures for revocation need to be designed accordingly.

B.4.2.2 Rate Limiting (Throttling)

Rate limiting, also referred to as throttling, is the primary defense against online attacks on the authenticator, authenticator output, or an activation factor used by a multi-factor authenticator. The throttling parameters have been chosen based on the value being guessed by the attacker having approximately 20 bits of entropy, or a likelihood of success of 1 in 1 million guesses. The 100 guesses permitted therefore gives an attacker approximately a 1 in 10 thousand chance of success.

Rate limiting, of course, is an opportunity for an attacker to be able to perform a denial-of-service attack on the subscriber. Several suggestions are made to mitigate that possibility. The use of a CAPTCHA tends to protect against automated attacks, and the use of delays increases the likelihood that the attack will be discovered before being complete. The scope of the rate limiting (such as by IP address) can also be limited, although it is important to consider the capabilities of potential attackers to launch a distributed attack from many IP addresses.

When multiple authentication factors are being used, it is sometimes possible to rate-limit only when one of the factors is successful. For example, an authentication using a memorized secret plus an OTP authenticator output might only throttle (or perhaps even prompt for) the OTP when the memorized secret is correct. It might also prompt for both and not indicate which factor, if any, had succeeded and thereby only throttle the unsuccessful factor, whichever it was, if the other factor was correct.

B.4.2.3 Use of Biometrics

Biometric authentication is a rapidly evolving area, and it is important to use biometric systems with actual measured performance characteristics. It is also important to work within the revocability and secrecy limitations of biometrics.

One of the primary limitations of biometrics is that they cannot be revoked: it isn't possible to change your fingerprint, iris pattern, or other modalities if your biometric becomes known to a potential attacker. This is addressed by the requirement that there be a strong binding between the biometric and a physical authenticator. A biometric is enrolled for use with a specific physical authenticator, and if there is a suspicion of misuse, it is the physical authenticator, not the biometric itself, that is revoked or suspended.

Biometrics are also not secret. High-resolution cameras have been shown to reveal a person's iris pattern in enough detail for authentication, and fingerprints are left behind on many things you touch. There is, of course, the challenge of producing a model that replicates the subscriber, a challenge that is made more difficult with the use of presentation attack detection (PAD) technology. But liveness detection and the need to replicate some aspect of the subscriber is only difficult if the attacker does not control the biometric sensor. "Skimmers" for credit cards and PINs are commonplace, and use of skimmers and devices that can spoof biometrics should be expected as well. If the sensor and processing cannot be trusted, a collected biometric could be substituted for that from an actual sensor. For this reason, the sensor (or endpoint with a tightly integrated sensor) needs to be authenticated to ensure that it is not an impostor.

Current performance of biometric sensors and processing leads to the requirement of a false-match rate of 1 in 1000 or better. Furthermore, this rate is measured under conditions of a zero-effort attack: biometrics from random people being tested, without intentionally picking biometrics that are more likely to be accepted. Because this rate is significantly lower than authenticators like memorized secrets and OTPs, more restrictive throttling requirements have been adopted. Depending on whether PAD is implemented, throttling begins at 5-10 failed attempts, and increases exponentially after that. For this reason, an alternate modality, or the use of a memorized secret as the second factor, is probably required in most situations.

Biometrics can be verified centrally, although increases in processor performance (e.g., in mobile devices) makes it increasingly practical to verify biometrics at the sensor location. If central verification is performed, additional requirements about the security of the biometric data in transit and authentication of the sensor/endpoint are imposed. In particular, use of a biometric is required to be tightly bound to specific device(s) for which the sensor and endpoint have been determined by the verifier to meet the required performance parameters.

B.4.2.4 Attestation

While verifiers must of course authenticate the claimant, they must also have some information about the manner in which that has occurred. In some cases, this may be obvious, e.g., the use of a memorized secret plus an OTP, with both authenticator

outputs being presented to the verifier. In others, the similarities between some multi-factor authenticators and their single-factor counterparts gives rise to a need to securely determine what type of authenticator is used. This need is particularly applicable for "bring your own authenticator" situations, where the subscriber uses an authenticator obtained elsewhere rather than one that has been provided by, and is known to, the CSP. On the other hand, when the CSP has issued the authenticator, or has an opportunity to examine it, the CSP can determine the nature of the authenticator directly.

One situation in which attestation or direct examination is needed is in determining whether the security requirements of the authenticator have been met. For example, at AAL3, multifactor authenticators are required to meet FIPS 140 Level 3 physical security and Level 2 overall. Attestation information describing the authenticator being used can allow the CSP or verifier to determine whether that requirement has been met.

Attestation usually is required only at the time the authenticator is bound to the subscriber's account and not in connection with each authentication, since it is expected that it will be difficult to move secrets from an acceptable authenticator to one that is less secure.

B.4.2.6 Verifier Impersonation Resistance

Verifier impersonation resistance is a characteristic of some cryptographic authenticators that bind the authenticator output to a specific authenticated protected session (usually a TLS session). Verifier impersonation resistance is effective against certain types of "phishing" attacks where the claimant is misdirected to a look-alike site where they are encouraged to authenticate.

When authentication is attempted at a phishing site operated by the attacker, the attacker can capture the authenticator output and initiate their own authentication session with the actual relying party. For example if a one-time password is used (such as from an OTP device), the attacker can use the authenticator output immediately after it is entered, so even a time-based OTP does not protect against the attack. Challenge-response protocols are similarly ineffective because the attacker can open an authentication session and obtain the challenge nonce, relay it to the claimant, and then have the necessary response to authenticate the attacker's own session.

Establishment of authenticated protected sessions creates encryption keys unique to the session using Diffie-Hellman key exchange. This key can be included in calculating the authenticator output, so that the output is not valid for any other authenticated protected session (such as that between the attacker and the actual relying party in the example above).

Note that verifier impersonation resistance requires both a directly-connected authenticator (a cryptographic device or cryptographic software authenticator) as well

as support in the application, such as a web browser, in which it will be used. Some cryptographic devices do not actually bind to the protected session secrets and are therefore not verifier impersonation resistant.

B.4.2.7 Verifier-CSP Communications

SP 800-63B assumes a very close relationship between the verifier and the CSP: that they are two different roles for the same entity or that they are very closely associated, perhaps under common administration. In the latter case, this section reinforces the requirement that all communications between the entities be strongly protected by a mutually-authenticated secure channel.

B.4.2.8 Verifier Compromise Resistance

A common form of authentication compromise is an attack on the verifier, which if successful may be able to harvest information that can later be used to authenticate to that verifier. Authentication protocols where the verifier has data that can only be used to verify, and not generate, the authenticator output are referred to as bring verifier compromise resistant.

Public keys used with approved algorithms and having at least the minimum security strength specified in SP 800-131A are considered to be verifier compromise resistant, as are hashed keys when the key being hashed has the necessary security strength. For example, look-up secrets that are sufficiently complex would be considered verifier compromise resistant when hashed with an approved algorithm.

Certain types of authenticators, notably OTP devices and cryptographic devices that generate one-time passwords, cannot be verifier compromise resistant because they need to share a secret with the authenticator in order to generate an authenticator output for comparison.

B.4.2.9 Replay Resistance

An authenticator output is considered replay resistant if its output can be used only once for authentication. Most authenticators specified are replay-resistant, with the notable exception of memorized secrets when they are used independently (other than as an activation factor for a multifactor authenticator).

B.4.2.10 Authentication Intent

One of the concerns with embedded and directly-connected authenticators (typically cryptographic device authenticators) is the question of whether malware in the endpoint device that hosts the authenticator can cause authentication to occur without the subscriber's knowledge or consent. In this situation, the malware could proxy authenticator challenges from the attacker if needed, and obtain the authenticator output needed to sign the attacker into a service of interest. In common use, some cryptographic authenticators are left connected for multiple authentications, and are subject to this concern.

All authenticators that require claimant intervention establish authentication intent, provided that there is no caching of claimant input and that the intervention cannot be spoofed by software (e.g., by simulating keyboard input). In most cases, multifactor authenticators also establish intent unless claimant input is cached, although certain biometric modalities such as facial recognition may require additional measures to establish intent. Caching of claimant input should be avoided because it weakens the establishment of intent.

Authentication intent does not require that the claimant be authenticated in any way, only that someone has taken an action requesting authentication at the endpoint. This could be through the pressing of a button or the insertion or connection of an authenticator that is capable of only one authentication per insertion, for example.

Wireless authenticators (e.g., NFC, Bluetooth, ISO 14443) require special consideration with respect to authentication intent. Since a connection with the authenticator could be established by an attacker who happens to be close to a subscriber carrying one of these authenticators, intent cannot be established by connection to the authenticator alone. These authenticators should require the pressing of a button or similar action to operate, even if a challenge/response protocol is being used (a mobile attacker could proxy an authentication challenge nonce and collect the result).

B.4.2.11 Restricted Authenticators

As both authentication technologies and attacker capabilities mature, some authenticator classes and sub-classes will inevitably become less effective. In severe cases, these authenticators will be removed from acceptable use. In SP 800-63B, this has been done with pre-registered knowledge tokens (knowledge-based authentication) and with the use of email as an out-of-band authentication technique.

When continued use represents a lower immediate risk, authenticators may be classed as "restricted authenticators". Use of restricted authenticators requires additional scrutiny as described in SP 800-63B, particularly that a risk assessment be performed by the implementing organization.

Restricted authenticators should not be used for new implementations; authenticators that are restricted may be removed from future editions of SP 800-63B as attacker capabilities mature further.

At present, the use of PSTN (SMS and voice) to deliver out-of-band secrets is restricted. This was prompted by several factors, including:

- The demonstrated ability of attackers to obtain reassignment of telephone numbers used for authentication to new devices they control.
- Weaknesses in SS7 security that provide attackers with the opportunity to intercept out-of-band secrets sent via text messages.
- Ability in many cases for subscribers or attackers to forward these notifications to a new device, breaking the ability to determine possession of a specific device.

B.5 Authenticator Lifecycle Management

See SP 800-63B Section 6 for normative requirements.

B.5.1 Authenticator Binding

One of the changes in SP 800-63B from previous editions of SP 800-63 is the explicit recognition of bring-your-own authenticators. It is no longer assumed that CSPs will issue authenticators, which is important in the case of physical authenticators and with the increased use of two-factor authentication: it should not be necessary for a subscriber to carry, manage, and protect a "keyring" of devices for authentication on multiple services. While this issue is mitigated greatly by the use of federated authentication, many subscribers will nevertheless have accounts at multiple CSPs. The use of multiple authenticators at each CSP, prompted by more secure account recovery procedures, also increases the number of authenticators that must be managed.

SP 800-63B uses the term *binding* rather than *issuance* to better accommodate bring-your-own authenticators since the authenticator(s) being used may have been issued elsewhere. At the same time, bring-your-own authenticators introduce a new problem: the need for the CSP to determine the type and strength of authenticators it binds to the account. This is discussed in the section on authenticator attestation.

The binding refers to the association between the subscriber's account at the CSP (the credential) and the authenticators that can be used to access it. While binding multiple authenticators does increase the attack surface of the subscriber's account, availability of a reasonable number of authenticators minimizes the need for account recovery, which can be made more secure if it is a rare event. It also accommodates the different interfaces available on different devices.

Each authenticator has a metadata record associated with it, including information on the binding that was established and unsuccessful authentications attempted with it. The latter includes state information that is needed to implement rate limiting of a specific authenticator as described in Section 5.2.2 without necessarily rate limiting the entire account.

B.5.1.1 Binding at Enrollment

Binding of one or more authenticators usually immediately follows an identity proofing transaction. It is important that the binding of authenticators be strongly associated with the identity proofing process to ensure that the subject associating an authenticator with a subscriber's credential is, in fact, that subscriber.

B.5.1.2 Post-Enrollment Binding

Post-enrollment binding includes the binding of additional authenticators for backup purposes as well as in response to the loss, theft, or damage to an existing authenticator. The latter situation, often referred to as "account recovery", has been the weak point of many authentication systems. All of the effort in strongly authenticating subscribers is moot if an attacker can successfully claim the loss of one or more authenticators and obtain the binding of new authenticators under their control. For this reason, binding of new authenticators requires either authentication with existing authenticators or a repeat of some or all of the identity proofing process.

B.5.1.2.1 Binding of an Additional Authenticator at Existing AAL

The most common binding situation is when a subscriber wants to bind an additional authenticator to their account. This may occur as a result of the loss, theft of, or damage to an existing authenticator. It might also be done to create an additional backup authenticator or one that is compatible with a different hardware device.

Associating a new authenticator to an account is a somewhat more sensitive transaction than a routine authentication, because a successful attack that is not detected might provide the attacker with ongoing access to the subscriber's account. However, authentication at a higher AAL is often not possible because of the limitations of authenticators that the subscriber already has. To address this concern, SP 800-63B recommends that a notification be sent to the subscriber when a new authenticator is bound to the account to increase the likelihood of detection of an unauthorized binding.

B.5.1.2.2 Adding an Additional Factor to a Single-Factor Account

A special case that happens at most once per subscriber account is the need to associate a second factor when one authentication factor is currently bound. Because the account has been only used with single-factor authentication, it is assumed that the subscriber has already accepted the risk associated with that use and that the binding of a new authentication factor doesn't represent an escalation of privilege (e.g., access to personal information that wasn't accessible before).

As noted in this section, it is possible for a subscriber to add an additional authentication factor if only one is currently bound. This is usually a physical authenticator being added to an account that currently only has a memorized secret authenticator. As above, the subscriber should be notified of this event.

B.5.1.2.3 Replacement of a Lost Authentication Factor

In a perfect world, subscribers would never lose authenticators, or would have another authenticator of the same factor(s) available as a backup. However, in practice subscribers lose or damage authenticators with some regularity. It is often possible to bind additional physical authenticators to mitigate loss of something you have, but it is considerably more difficult to recover from the common situation where a memorized secret has been forgotten. Having a secondary memorized secret as a backup is not a good response, since this secondary secret would be rarely used and more likely to be forgotten.

The most common and problematic situation is the loss of a memorized secret. A special provision is made for recovery in this case, involving the use of two physical authenticators and a recovery code that is sent by the CSP to the subscriber's address of record. The recovery code serves as both notice to the subscriber and protection against an attacker that is able to steal two or more of the subscriber's authenticators. While not fully two-factor, this procedure confirms the subscriber's ability to receive messages at the address of record.

B.5.1.3 Binding to a Subscriber-provided Authenticator

So-called "bring your own" authenticators, while desirable from a convenience and complexity point of view, introduce some new issues. As discussed above, the CSP needs some assurance as to the type and security of authenticators that are bound to the subscriber's account. The CSP should always make a default assumption of the weaker authenticator type (e.g., single-factor as opposed to multi-factor crypto device, or software-based as opposed to hardware-based authenticator) when it is not able to reliably establish the nature of the authenticator.

B.5.1.4 Renewal

The authenticator renewal process should begin well before the actual expiration of a previous authenticator. Lifetimes of physical authenticators should be chosen to balance the risk of the undetected loss of an authenticator and the cost and complexity of reissuance.

Authenticator expiration should not be seen as conflicting with the earlier guideline that memorized secrets should not have routine expiration. Expiration of memorized secrets (without some other indication of a security breach having occurred) should be avoided because of the users to choose poor memorized secrets when they know they will need to replace them soon.

B.5.2 Loss, Theft, Damage, and Unauthorized Duplication

The possibility of authenticator loss, theft, damage, and unauthorized duplication require that the CSP provide an effective means for reporting and requesting the suspension or revocation of an authenticator without creating a mechanism for denial-of-service attacks on the subscriber. Suspension gives the subscriber an opportunity to intervene before an authenticator is removed from the account entirely, thereby mitigating the severity of such denial-of-service attacks.

B.5.3 Expiration

Expiration of authenticators is permitted but not required by SP 800-63B. As discussed above, the decision whether and at what period to expire authenticators should be made by the CSP based on a risk analysis process. Expiration of some authenticators (certificate-based cryptographic authenticators) is sometimes useful to limit the growth of certificate revocation lists (CRLs) since revoked authenticators can be removed from the list once they have expired.

As discussed above, routine expiration of memorized secrets is discouraged because of the tendency for subscribers to choose weaker secrets when they have to change them periodically.

B.5.4 Revocation and Termination

Revocation and termination address situations when a subject ceases to be a subscriber of a given CSP. Procedures used by specific CSPs, such as in FIPS 201 with respect to PIV credentials, supplement the information in this section.

While the use of an authenticator by a specific CSP for online authentication is relatively easy to revoke, authenticators that contain user attributes or that can be used for physical authentication require much more emphasis on physical collection and destruction. Relying parties that use information contained on such authenticators need to consider the possibility that those attributes are stale, and should verify the attributes with the CSP online when this is practical, depending on the sensitivity of the application.

B.6 Session Management

See SP 800-63 B for normative requirements.

Session management comprises a number of mechanisms that are used following authentication to maintain continuity of state for a subscriber. Strength of session management procedures is as important as authentication, since the ability to hijack a session is as damaging as an authentication failure.

Sessions have well-defined maximum lifetimes. This lifetime can be extended through the reauthentication procedures outlined in Section 7.2 provided that reauthentication occurs before the previous session has expired.

B.6.1 Session Bindings

A session is maintained by a secret shared between the subscriber and host (CSP or RP). The host generates a random secret, and sends it to the subscriber over the authenticated protected channel used for subscriber authentication. It is very important that the secret not be guessable by an attacker; for this reason, the secret needs to have sufficient entropy to resist guessing attacks and needs to be generated by an approved random bit generator. Other methods for session secret leakage also need to be avoided, including possible extraction from log files if it is included as a URL parameter.

B.6.1.1 Browser Cookies

Browser cookies are by far the most common mechanism for session management. A number of requirements are given in this section to ensure that they are used in a secure manner, such as that they only be accessible in secure (HTTPS) sessions, so that they can't be intercepted in transit by an attacker and that they be inaccessible from (perhaps rogue) JavaScript.

Expiration of cookies used as session bindings depends on how long the CSP will accept the cookie as valid, which is determined by the reauthentication periods at each AAL. Cookies also have an expiration time, which primarily functions to allow the browser to discard cookies that will no longer work. This expiration time should be set slightly longer than the reauthentication period, and their expiration should be reset when reauthentication occurs.

B.6.2 Reauthentication

Reauthentication is the process by which the CSP reconfirms that a session is still under the control of the subscriber. Reauthentication occurs periodically depending on the AAL associated with the session and whether the session has actively been in use. It mitigates the risk that the authenticated endpoint leaves the subscriber's control and falls into the hands of an attacker. Even though session secrets are only a single factor, and two factors are required at AAL2 and AAL3, the short-term nature of these secrets and the requirement that they be sent only over an authenticated protected session mitigates the risk of compromise.

Reauthentication times are considerably shorter for sessions that are idle (without subscriber activity). This is because there is a greater risk of endpoint hijacking when there is no subscriber activity, e.g., when the subscriber goes to lunch. Session idle time is measured from the last user interaction with that specific session; other activity on the endpoint (e.g., user interaction with a different browser window or tab, or different application) does not reset the idle timer.

At AAL2 and above, reauthentication requires use of either a memorized secret or biometric associated with a physical authenticator. While possibly inconvenient, it is important to establish the presence of the subscriber, rather than a physical authenticator that may have been left at the endpoint. At AAL1, any authenticator may be used, but in practice that will usually be a memorized secret.

As noted, prior to reauthentication time it is acceptable for the RP to display a warning, such as "reauthentication will be required in 5 minutes" or "this session appears to be idle: reauthentication will be required in 30 seconds if there is no activity" to avoid unpleasant surprises for the subscriber.

B.6.2.1 Reauthentication from a Federation or Assertion

Federated authentication presents a new reauthentication issue: both the CSP (functioning as IdP) and RP may maintain reauthentication time. In most cases, it is the RP's reauthentication time that governs the timeout. If the IdP asserts the subscriber's identity to an RP based on an earlier authentication (which must have occurred within the reauthentication time), the IdP should assert the time of authentication and maximum authentication age to the RP, so that it can base its reauthentication timer on that information.

C.1 Introduction

Federated identity transactions allow for a more secure and more usable internet by allowing subscribers to have a smaller number of accounts that can be used across many sites and applications, without using the same authenticator at multiple sites or applications. There are several major protocols that enable federation transactions, and a multitude of software packages and libraries that implement them. This document outlines what to look for in software that enables federation and how to apply best practices to that software to meet the requirements in SP 800-63C.

This document is intended to provide more direct technology discussion than SP 800-63C, which was written to be intentionally technology-agnostic. While this choice makes the SP 800-63 guidelines applicable across a wide array of technologies and circumstances, the abstract nature can make it difficult for implementers to understand what was intended by the document with regard to specific protocols or products. This guide is intended to provide more concrete information for implementors of these systems.

This document contains no normative requirements.

Note: These resources use the term IdP in a manner consistent with the use of the terms in SP 800-63C. Specifically, the IdP role is fulfilled by the CSP, and the RP is the receiver of the federated assertion.

C.2 Choosing Security Parameters

Different federation protocols and implementations of those protocols have many options that lead to different outcomes in the security of a system. All of these options have trade-offs in terms of complexity, robustness, and other characteristics. Choosing the right set of options for a given situation helps ensure that transactions will be as secure, functional, and efficient as possible.

There must always be a balance between the complexity of a solution and the threats it protects against, and each deployment situation will lead to its own requirements. There is, unfortunately, not a one-size-fits-all approach that can be applied blindly to all situations.

C.2.1 Selecting a Protocol

A number of different federation protocols exist, but the two most common ones today are Security Assertion Markup Language (SAML) and OpenID Connect (OIDC). These protocols are not compatible with each other, but they offer some similar capabilities. For the most part, protocol selection will be based on the technology support available in the target environments. However, some core aspects of the protocols themselves lend them to different choices.

SAML is a protocol based on passing XML documents between different parties. The web single-sign-on profile for SAML allows it to be used as a federation protocol between websites. SAML has extensive support and deployment in some spaces, but lacks the flexibility that is often needed for modern systems. For instance, SAML is not well-suited for log in to a mobile application, nor is it a good fit for protecting API access. Additionally, most SAML federations are static in nature, or controlled by centralized federation authorities. While this is a valid model, it is limiting in terms of which applications can be used.

OIDC is a protocol based on the OAuth 2 delegation framework. While OIDC is primarily a web-focused protocol, it is also usable with mobile and native applications. Since it is built on OAuth 2, OIDC allows for delegated access to additional APIs alongside identity information, a feature that is often desirable in today's integrations.

C.2.2 Selecting a Federation Assurance Level (FAL)

The Federation Assurance Level (FAL) defined in SP 800-63C Section 4 provides a set of requirements for federation transactions. These requirements are grouped into an ascending scale of three levels: FAL1, FAL2, and FAL3. Each successive level includes all the features of lower levels and adds additional requirements on top of them. Each

level introduces protections against specific kinds of attacks, and these protections are applicable only in some situations.

FAL1 provides a solid basis for federation and is appropriate for the vast majority of use cases. Most off-the-shelf products operate at FAL1 today, including most common deployments of SAML and OIDC. FAL1 requires that all assertions be signed, time limited, and audience-restricted to prevent an assertion intended for one RP to be replayed at another RP. It is anticipated that most deployments will build out at FAL1 and move to FAL2 or FAL3 only when necessary.

FAL2 provides an extra layer of security by requiring that the assertion be encrypted so that the RP is the only party that can decrypt it. This level requires that the IdP manage an encryption key for the RP, which necessitates additional complexity for both parties. The keys could be managed with a traditional PKI infrastructure that relies on a trusted certificate authority, but with many protocols the keys can be instead registered directly between parties. The RP also needs to manage and protect its decryption keys in order to read the information in the assertion. If the RP's private decryption keys are leaked to another party, the additional protections provided by FAL2 are no longer in play.

FAL3 is intended to be a forward-looking requirement and is not yet readily available in off-the-shelf standards and products. FAL3 provides an additional layer in the form of a cryptographic key that is presented by the subscriber directly to the RP in addition to the signed and encrypted assertion itself. This level requires that the IdP manage references to keys representing the subscriber at each RP in addition to managing the keys for the RPs themselves. The IdP needs to correctly associate the subscriber's key to the correct RP in the assertion, and the RP needs to be able to process and validate the presentation of the key by the subscriber. This key could be the same key that's presented by the subscriber at the IdP, or it could be a separate key that's not used at the IdP. The key could be in a credential with its own attributes, such as an X.509 certificate, or it could be tied to an authenticator, such as a FIDO token. In all of these cases, the assertion needs to reference the key and the RP needs to ensure the correct key is presented by the subscriber.

C.2.2.1 Risk Management

Selecting and conforming to an FAL ought to be part of a larger risk management process and program. Conforming to FAL3 does not make an organization's security infallible, but instead provides protection against particular attacks while incurring certain costs to both the applications and the subscribers. Rather than attempting to make federation infrastructure conform to the highest standards available, it is recommended to analyze the inherent risks and choose how strongly to protect against them given their severity and likelihood of occurrence.

The additional information management and implementation complexity of higher FALs cannot be ignored, and the costs to all involved have to be weighed against perceived

benefits. Unless there is a compelling reason to use the features of higher FALs, FAL1 is the industry standard for most use cases. The risks of implementing a system at FAL1, when compared to higher FALs, may be negligible depending on relevant use cases and attack vectors.

Because it is the front door to many critical systems, authentication is a key piece of risk management strategy. Strong federation can protect against many potential subscriber impersonation and man-in-the-middle attacks. Instead of each RP needing to manage subscriber accounts and authenticators separately, creating many vulnerable surfaces, federation concentrates the key security practices in a dedicated component, the IdP. Upgrades to authenticators, software, and practices at the IdP automatically benefit the downstream RPs and the overall network.

C.2.3 Personally Identifiable Information (PII)

Personally Identifiable Information (PII) needs to be limited to only what's needed to perform a transaction as per Section 7.3. For many login transactions, the RP will need to know only an identifier for the current subscriber. After an initial login, this identifier is used by the RP to tie the subscriber to a record or account in the RP application, and this record often contains attributes collected from various sources including the IdP and direct interaction with the subscriber.

All assertions contain a subject identifier, which uniquely identifies the subscriber represented by the assertion to the RP. Since the subject identifier is required with every assertion, the subject identifier should not have any PII internally. For example, IdPs would not use usernames, employee numbers, simple sequences, or other easily predictable and correlatable information for the subject identifier. Instead, to prevent PII leakage, IdPs can use approved cryptography to assign random subject identifiers to all subscribers. Alternatively, IdPs could derive subject identifiers from PII using approved cryptography. For example, the IdP could run an internally unique identifier (like an employee number) through an approved hashing function. The output of the hashing function would be the subject identifier, and the PII used to generate it has been protected since it cannot be re-derived from the output of the hashing function.

Assertions at all levels can include additional attributes about the subscriber as per Section 6, potentially including PII. The RP might require some of this information to perform its function, such as an email address to contact a subscriber. However, the RP will not know if it needs information for a given subscriber before that subscriber has been logged in, since the required information could be available locally to the RP without requesting it from the IdP. This presents a dilemma for systems trying to practice data minimization. Some protocols, such as SAML, require all available attributes be present in the assertion itself. In such cases, the RP needs to be very judicious about which attributes it requests. In other protocols, such as OpenID Connect, attributes can

be sent through both the assertion and a secondary channel, the UserInfo endpoint. In OpenID Connect the ID Token serves as the assertion, and by default it contains a non-PII subject identifier for the subscriber. Additional information about the subscriber can be obtained through the UserInfo Endpoint by using an OAuth access token. Since this information is communicated in the back channel from the IdP to the RP over an authenticated protected channel, it need not be separately encrypted as it is not handled or presented by an intermediary party (though of course it can be encrypted as well).

FAL2 is required if PII will be sent in an identity assertion that is passed through an intermediary such as a browser, as personally identifiable information needs to be protected in transit. If the PII is sent over the back channel instead, either in the assertion or in a separate request, then FAL2 is not required in order for the information to be sufficiently protected in transit.

C.2.4 Selecting a Presentation Mechanism

Assertions can be sent either over the back channel between the IdP and RP as per Section 7.1 or over the front channel using the subscriber and their browser as an intermediary as per Section 7.2. While both methods are allowed at all FALs, back channel presentation has a number of advantages and is preferred where possible.

Since the assertion is transmitted directly from the IdP to the RP over an authenticated protected channel, the RP has a high level of assurance that the assertion is from the IdP in question. The RP can also be sure that the assertion is generated in response to a specific authentication request since the RP needs to present an assertion reference to retrieve it.

Front channel presentation systems are simpler for RPs to implement, as the RP does not have to handle assertion references or take the extra step of trading the reference for an assertions. As a consequence of this simplicity, front channel presentation systems can be more susceptible to assertion injection, whereby an attacker can present a valid assertion to an unsuspecting RP in order to force a login at that RP, potentially taking over a session. A back channel presentation system is subject only to injection of an assertion reference, which can be more strongly tied to an authentication request.

To enforce least-privilege and least-knowledge security principles, it is preferable to have each RP request its own assertion instead of re-using one assertion for multiple RPs. With front-channel presentation, it is tempting for a system to create a single assertion to be presented to multiple RPs by the browser. RPs in such a system would be configured to accept an assertion that they did not explicitly request, in order to facilitate a more seamless user experience. The subscriber's browser then needs to determine which sites to present the assertion to, and which to request a new assertion for. With a back channel presentation mechanism, only the assertion reference is passed to the RP from the browser.

Since the assertion reference is one-time use and limited to a single RP as in Section 7.1, it cannot be used accidentally multiple times at multiple RPs.

Additionally, assertions passed in the front channel are visible to an intermediary party, the browser. The assertion's payload, which is intended for consumption by the RP, can include PII attributes, internal security information, or other sensitive data. To avoid unintended data leakage, the IdP can employ several techniques:

1. Encrypt the payload to the RP's key, as is required at FAL2 and FAL3.
2. Limit the information contained in the assertion payload to non-sensitive information such as identifiers, and use a secondary mechanism (such as OIDC's UserInfo Endpoint) to convey sensitive details.
3. Use a back-channel presentation mechanism to prevent the assertion itself from being seen by the intermediary.

C.3 Guidance for Relying Parties

While it is the responsibility of the IdP to provide strong and trustworthy federation assertions, relying parties need to validate the elements of an assertion during a federated transaction as in Section 6.2.

Relying parties can be a valuable target for attackers to impersonate valid subscribers or gain valuable information about them. If relying parties do not check the validity of the information they receive, attackers can gain access to the various services that subscribers are logging in to.

If all of these checks are performed properly, the compromise of a single relying party does not threaten the rest of the network. This is in contrast to systems with some kinds of individual authenticators at each site, where the theft of a subscriber's password from one site often leads to the compromise of other sites in the network due to password reuse.

C.3.1 General Guidance

Relying parties need to do two major checks against incoming assertions:

- check that the assertion itself is internally consistent, and
- check that the assertion is verifiably from a trusted source.

Relying parties need to validate IdP signatures, assertion expirations, and audience parameters within an assertion to validate that the assertion itself is internally consistent. Additionally, RPs need to test that these validation checks are working at all times because there will be no outward indication that something is wrong with the system until an attack occurs. In other words, RPs need to ensure that they are rejecting invalid assertions just as much as they are accepting valid assertions.

RPs need to also verify the origin of the information that they receive, as an attacker might try to inject a valid assertion from another subscriber in order to take over an account. RPs can do this by making sure that the assertion is signed by a trusted IdP's key and that the assertion is not being replayed. RPs also need to check the source of the assertion. This most often means that they will accept an assertion only if it is presented in response to a direct request for one, and not at any other time. This also means that an RP will only accept assertions generated and signed by the IdP they were connecting to.

C.3.1.1 Validating IdP Signatures

At all FALs, an identity assertion is signed by an IdP so that it cannot be forged by an attacker as per Section 6.2.2. The IdP is the only entity with access to its private key (detailed in Section 4.1), so a valid signature indicates that the assertion is from the

IdP itself and not an attacker, and that it has not been modified by an attacker. If an RP does not check the validity of the IdP signature, attackers will be able to forge identity assertions and gain access to protected systems without authorization. Additionally, if the relying party does not check the signature, an attacker could modify an otherwise valid assertion in transit, associating attributes and access rights to the current subscriber that were not asserted by the IdP.

Some protocols cover the entire assertion with a signature, while others cover only portions of it. The relying party has to take care to not accept unsigned portions of the assertion as validated even when presented alongside a signed assertion.

Validating the signature is not enough. The RP also needs to make sure it is using the correct key for the claimed IdP, especially if the RP accepts assertions from multiple IdPs. In OpenID Connect, for example, the IdP is identified by the "iss" field of the ID Token's payload, and the signing key is identified by the "kid" field in the ID Token's header. The RP will accept the token if and only if the signature validates using the identified key from the identified issuer, and then only if the issuer is trusted to provide identities to this RP. Additionally, the RP will accept the assertion only if it is issued by the IdP that the RP is currently communicating with. Otherwise, a rogue IdP could replay an assertion issued by another IdP in an attempt to grant an attacker access to the RP.

Testing whether RPs will reject unsigned assertions or assertions with invalid signatures is critical, though not an obvious test to do. Properly authorized transactions will still work even if an RP is not checking assertion signatures, since the RP will accept the (valid) assertion whether or not it has a valid signature. Therefore, in such cases there is no outward indication of a problem in the system and there will be no error messages or login failures to indicate that something is wrong. Only a failure from a negative test – that is to say, the explicit rejection of an unsigned assertion or an assertion with an invalid signature – will indicate that a relying party is properly checking keys and signatures.

C.3.1.1.1 Retrieving IdP Keys

The RP can trust the assertion's signature only as much as it can trust that the keys used to verify the signature are associated with the IdP as per Section 6.2.2. The keys need to be retrieved in a secure fashion, such as over an authenticated protected channel or having been pre-placed by a systems administrator. Only the keys identified in the assertion can be used to evaluate the signature of an assertion.

C.3.1.2 Checking Assertion Expirations

Federated identity assertions are intended to be short-lived, since they are used to establish a session at the RP and not to manage a full session at the RP (see Section 5.2). While details vary per protocol family, an assertion lasting a small number of minutes

will in most cases give the system ample time to process the assertion and create a session for the subscriber. Since federation assertions are passed between different systems on the network, it is reasonable to allow a small amount of padding to the time checks to account for clock skew. This skew ought to be very short, such as a few seconds, so as to not inadvertently open the attacker's window for using expired assertions. A time synchronization protocol such as NTP can be used on all systems on the network if possible to ensure the system clocks are as accurate as possible.

An identity assertion which expires quickly makes it difficult for attackers to misuse the assertion and also ensures that any identity or authorization information included in the assertion is not out-of-date. RPs need to be tested to ensure they do not accept expired assertions, which can be done by presenting the RP with an expired but otherwise valid assertion and seeing if the RP accepts or rejects it.

Some assertions also contain a timestamp indicating when the assertion was issued, and an RP should not accept any assertion that claims to have been issued in the future. Some assertions will also have a timestamp indicating when the assertion is not to be used before, which an RP can process to ensure it is not accepting an assertion too early. The use of the "not-before" processing mechanism is relatively rare in modern federation protocols, as the assertions are created in response to specific login requests.

All of the date fields have to be covered by the assertion's signature.

C.3.1.3 Checking Audience Restrictions

Common attacks include taking an assertion intended for one RP and presenting it at another RP, with or without modification. When an IdP creates an assertion, it includes an audience field indicating which RP requested the assertion as in Section 6.2.4. By checking the audience field of the assertion, an RP can detect when an attacker is presenting an assertion intended for a different RP.

If an RP does not check for a matching audience parameter, it is possible for an attacker to get a valid assertion from any RP registered with the IdP and replay it at the target RP to gain unauthorized access.

An RP that is not checking audience parameters will still accept a valid authorization with no outward indication of a problem. Therefore, it is important to test the RP with an assertion containing an errant or missing audience field.

The audience field has to be covered by the assertion's signature.

C.3.1.4 Checking Assertion Uniqueness

An attacker that gains possession of a bearer assertion could try to replay that assertion at an RP in order to take over a subscriber's session. To prevent this, an IdP is required to make each assertion unique as per Section 6.2.1. The RP consequently needs to check the assertion for uniqueness within the assertion's expiry window by checking any unique identifiers within the assertion and accepting each unique assertion identifier once and only once to establish a session with a single subscriber. If an assertion is seen multiple times by an RP, especially from multiple connections, the RP can consider this assertion stolen.

The RP ought to remember the identifiers of assertions as long as those identifiers are valid. Since assertions have a relatively short lifespan, this can be accomplished without large storage requirements by remembering only otherwise-valid assertion IDs within their validity window. If an assertion is replayed after it has expired, it will be rejected based on its expiration.

C.3.2 Guidance by Product Family

This document covers two main product families that enable federated identity transactions - SAML and OpenID Connect, the latter of which is built on top of OAuth. Other protocols and approaches are possible to use while fulfilling the requirements of the guidelines.

C.3.2.1 SAML

All parties need to be careful about passing and validating metadata. Incorrectly communicated or configured metadata could leak information about a subscriber that was not approved for distribution. Metadata that is not validated could have been tampered with by an attacker to gain access to valuable personal information.

The RP has to always check certificates before accepting identity assertions. Attackers can forge certificates and phish subscribers in an attempt to impersonate them.

SAML is not well-suited for use when the RP is a mobile or desktop application. Additionally, SAML does not provide a good means for protecting APIs. These limitations should be considered when choosing a product family.

C.3.2.2 OpenID Connect

Different OAuth grant types or "flows" are appropriate for different kinds of applications at different FALs.

The authorization code flow is a back-channel presentation mechanism and ought to be used whenever possible, particularly for web server, native, or mobile applications. It is the most common and most secure way to implement OAuth, the underlying protocol of OpenID Connect. It can accommodate all three FALs depending on the exact configuration of the application. The authorization code flow makes use of back channel assertion presentation, which reduces the attack surface of the RP significantly by sending the assertion directly from the IdP to the RP without an intermediary party touching it. The RP ought to authenticate itself when presenting the authorization code to the IdP.

If the RP is a native or mobile application, it can use the PKCE extension or dynamic client registration to ensure that different copies of the client software can not impersonate each other at the IdP. The best current practices for OAuth 2 mobile applications specification provides additional guidance.

In-browser applications, sometimes known as Single Page Applications (SPAs), are particularly challenging. The implicit grant type is a front-channel presentation mechanism and is applicable for applications which are implemented entirely in front-end code and have the capability to store secrets outside of the subscriber's web browser. However, best current practice is to use PKCE and the authorization code grant type for an SPA, or to use the OIDC hybrid flows to protect information. Best practices for these applications are currently under development by the OAuth working group in draft-ietf-oauth-browser-based-apps.

The lack of ability to store secrets means that these sorts of applications can usually only function at FAL1 because they have no method of private key management which would enable encryption of identity assertions. Modern browsers could allow a dynamic registration of the SPA and an in-browser protected keypair, but this is not a common deployment pattern at this time.

The client credentials and resource owner credentials grant types are not allowed at any FAL.

C.4 Guidance for Identity Providers

While every RP is responsible for its own internal security, the nature of federation protocols allows the IdP to specialize in security in a way that benefits all RPs that connect to it. With traditional application security authentication methods, security breaches can cascade between systems. Common practices like password reuse allow the compromise of a single RP to lead to the compromise of many other RPs for a given account. In a federation network, the identity provider (IdP) is the only party that can assert the presence and validity of subscribers and their attributes. The compromise of a single RP does not cascade through the network.

As the linchpin of security in a federation network, IdPs have the difficult task of keeping track of both subscribers and RPs as well as connecting them in a secure fashion with a federation protocol. Compromise of the IdP will affect all downstream RPs. However, unlike RPs that are trying to provide a service or application, the IdP's primary purpose is to act as a security component for the rest of the federation. As such, it is vitally important that the IdP be held to the highest of security standards in implementation and deployment. As a specialty service, it makes sense to invest heavily in good security practices.

C.4.1 General Guidance

IdPs manage the primary authenticators and authentication processes for subscribers in a federation. Guidance for managing such authentication can be found in SP 800-63B, all of which applies to the IdP. In particular, IdPs need to manage and store subscriber credentials appropriately for the types of credentials in use. IdPs also ought to implement phishing-resistant technologies in subscriber-facing pages and may want to use risk-based security methods for all connections, including any hosted identity APIs.

Additionally, the attributes and identities asserted by the IdP are subject to whatever verification practices the IdP uses. Guidelines for such identity proofing and verification are found in SP 800-63A. Since IdPs manage subscriber attributes, including PII, IdPs need to protect all such attributes to ensure they are not divulged to attackers or other unintended parties. An IdP must also follow data minimization practices and not divulge more attributes about a subscriber than are necessary to fulfill the identity request and affect a successful login.

Much of the technical friction in setting up a federation stems from IdPs which are built and configured in such a way that onboarding new RPs requires a significant amount of manual human intervention (see Section 5.1.1). Any time there is unwarranted friction in a security process, the consumers of that process (in this case, RP implementors) will often find creative and usually-insecure workarounds to that process. Much of this friction is removed when IdPs support automated discovery mechanisms and simple automated

registration (see Section 5.1.2). In order to ease RP onboarding, IdPs ought to make their configurations discoverable in a machine-readable format over a secure protected channel, as appropriate to the protocol in use. Registration of new RPs can be streamlined through developer portals and dynamic registration systems at the IdP.

IdPs need to use approved cryptographic systems to generate all key material as per Section 4.1. IdPs also need to securely store all private key material in such a way that attackers, RPs, end users, and other parties do not have access to the private key. An IdP can use a single asymmetric key pair across different RPs on the network, and the keys can be rotated on a regular basis to further prevent the chance of forgery. The IdP's public keys ought to be made available to RPs over authenticated protected channels to allow RPs to fetch the keys when needed, especially if the IdP rotates its keys. The public keys can also be transferred via a trusted out of band processes, such as hand configuration by a systems administrator.

The IdP's private keys, which are used to sign assertions, need to be protected from subscribers, RPs, and other unintended parties. If the IdP's private keys are compromised, an attacker could generate arbitrary assertions and impersonate any subscriber on the network at any RP. While an RP's keys also need to be protected, the possible damage is much less. If an RP's keys are compromised, an attacker could impersonate a request from that RP but not impersonate any other RPs or the IdP itself.

IdPs have to securely store any symmetric secrets used by RPs in a fashion that reduces the likelihood of their capture, such as by storing a hash of the secret instead of the secret itself. All symmetric secrets need to be generated using approved cryptography, and a different secret needs to be generated for every RP that the IdP associates with. Similarly, if an RP talks to multiple IdPs, it has to have a separate secret for each IdP and not re-use them.

If an IdP provides public and private keypairs to subscribers or RPs, the IdP needs to store only the public portion of the RP's key. This practice prevents the IdP from impersonating the client and becoming a potential attack target for abuse of this key material.

C.4.2 Guidance by Product Family

This document covers two main product families that enable federated identity transactions - SAML and OpenID Connect, the latter of which is built on top of OAuth. Other protocols and approaches are possible to use while fulfilling the requirements of the guidelines.

C.4.2.1 SAML

Both IdPs and RPs ought to publish metadata in a well-known location. While there is no widely accepted standard for SAML metadata exchange, it is advisable to use a well-documented metadata endpoint to serve the IdPs metadata in the form of a single XML file to any RP who wishes to consume it.

The IdP needs to always check signatures on metadata and to only accept metadata that has been signed by the presenting RP.

Identity federations like InCommon share the metadata of hundreds of IdPs and RPs in a structured manner as per section 5.1.3. Adding an IdP's metadata to such federations will help RPs to find it easily.

Apply best practices to protect subscriber information. All SAML assertions containing personally identifiable information ought to be encrypted to the relying party to protect the PII from being leaked to the browser. Assertions containing only authentication information and no personally identifiable information can relax this encryption requirement.

C.4.2.2 OpenID Connect

IdPs can use OpenID Connect's discovery mechanism, published in JSON format at an HTTPS location ending in `/.well-known/openid-configuration` as specified in the OpenID Connect discovery specification. The discovery document contains all of the information that an RP would need to interact with the server. This document is usually made available in a location based on the IdP's unique issuer URL, and a single discovery location should be considered canonical for a given IdP.

If personally identifiable information is bundled with authentication information in an ID Token, it ought to be protected through encryption of the ID Token or use of a back-channel presentation mechanism. If personally identifiable information is made available at the UserInfo Endpoint, the ID Token need not be encrypted. All back-channel communications have to pass over an authenticated protected channel, such as HTTPS over TLS with server certificate validation.

It is recommended that OpenID Connect IdPs support a dynamic client registration to make it easy for RPs to register without manual intervention. Note that dynamic registration does not release any subscriber data to the RP, it merely allows the RP to ask for login to be authorized at runtime (See Section 4.2).

The OpenID Foundation has made a test suite available for OpenID Connect providers to verify whether their instance of OpenID Connect is compliant with the standard, available from http://openid.net/certification/testing/.

The OpenID Connect community has reviewed libraries in several different languages to search for bugs and non-compliant processes, available at http://openid.net/developers/libraries/. Whenever possible, developers should leverage these proven libraries in development.

OpenID Connect relies heavily on the JOSE standard, particularly JWT and JWK. All tokens an keys in an implementation have to conform to those standards. It is recommended that IdPs use an established JOSE and JWT library to ensure all appropriate checks have been made during implementation.

IdPs should implement additional security standards such as MTLS for OAuth 2 and Proof Key for Code Exchange (PKCE) to enable higher security interactions. While such standards are not factors in determining the FAL of an OpenID Connect IdP, they considered to be best practice in the industry.

C.5 Example Scenarios

This section describes some common scenarios in use across different protocols and deployment patterns.

C.5.1 Shibboleth and SAML

SAML Federations like InCommon can operate at FAL1 or FAL2. Most InCommon IdPs are running on a Shibboleth identity provider. They pass assertions through a response to an authentication event. Most often, those assertions are not encrypted to the RP and therefore conform to FAL1. For a Shibboleth IdP, either encrypt all assertions to the RP or refrain from sending personally identifiable information such as `eduPersonPrincipalName` (or `eppn`) over the wire as an unencrypted SAML assertion.

SAML can reach FAL3 by providing an attribute within the SAML assertion that references a cryptographic key to be presented by the subscriber at the RP. The subscriber would then need to present proof of possession of that key directly to the RP in order to reach FAL3.

C.5.2 OpenID Connect

Typically, OpenID Connect Providers interact with OpenID Connect relying parties by providing a signed authentication assertion (the ID Token) which is separate from the transfer of personally identifiable information (from the UserInfo Endpoint). As such, these providers can safely operate at FAL1 because they are not bundling identity assertions with authentication information. This characterization is true for both the authorization code and implicit client types.

If the ID Token contains PII and is passed on the front channel (through the implicit or hybrid flows), it needs to be encrypted to the RP at FAL2 to protect the PII. This is accomplished by using the JSON Web Encryption (JWE) specification and a key registered to the RP.

OpenID Connect can reach FAL3 by providing a claim within the ID Token that references a cryptographic key to be presented by the subscriber at the RP. The subscriber would then need to present proof of possession of that key directly to the RP in order to reach FAL3.

C.5.3 Personal Identity Verification (PIV) Card

PIV cards are considered an authentication technology by the SP 800-63 guidelines, not a federation technology. Therefore, using a PIV card during a login determines the AAL, as well as the IAL that was used to issue the PIV card. As FALs are independent of AALs, any authentication technology can be used to start a federation transaction at any FAL. Therefore, the use of a PIV card does not imply any particular FAL.

Since they are authenticators, PIV cards can be used to authenticate to an IdP and start a federation transaction. This approach allows the complex processing and validation of the PIV certificate chain to be handled by a specialized security component, the IdP, and identity information be sent to the downstream RPs by the federation protocol.

Recall that in a federation protocol, the subscriber's attributes are contained in the assertion or in an associated identity API. Therefore, the attributes contained in the PIV certificates can then be transmitted by the IdP to the RP through those mechanisms, assuming all consent and privacy considerations around attribute release have been followed as usual. However, not all attributes in the certificate need be sent, allowing an IdP to tailor the information that it discloses to a given RP. Additional attributes can also be sent in the federation transaction that are not included in the certificate itself. Additionally, the attributes sent through federation are significantly easier to update than those within the PIV card's certificates, which necessitate issuance of a new certificate in its entirety.

Federated identity protocols allow subscribers to authenticate at an RP regardless of which authenticator they use at their IdP. This allows an RP to support all derived PIV credentials that an IdP has associated with the subscriber without having to verify the derived PIV credentials directly.

A PIV card can also be used to reach FAL3 by acting as the secondary authenticator alongside the assertion. If the assertion contains a reference to the PIV authentication certificate, and the RP directly verifies that the subscriber can present that certificate, then FAL3 can be reached.

C.5.4 Privacy-enhancing Federated Identity

In many cases, RPs do not need to know the full set of attributes available for a subscriber. RPs need to request only as much information as they need to complete the transaction requested by the subscriber, and IdPs need to limit what information RPs have access to within a transaction as per Section 5.2. Furthermore, with protocols like OpenID Connect, the attributes of the subscriber can be sent separately from the assertion itself, limiting leakage of this information.

Pairwise identifiers ought to be used in place of persistent or correlatable identifiers whenever possible (See Section 6.3). This limits relying parties in attempts of tracking or identifying individual subscribers across different systems.

When possible, claim references ought to be used to communicate identity information rather than raw data (See Section 7.3). For example, if a relying party needs to know whether a subscriber is over eighteen years old, the IdP can respond that the subscriber is over eighteen without sharing the subscriber's age or birthdate.

C.5.5 Parallel Authentication

In some cases a relying party may wish to confirm certain aspects of a subscriber's identity above and beyond what the IdP provides. For example, a relying party could log in a subscriber using an IdP, receive a picture of the subscriber from the IdP, and require that an in-person agent verify that the picture matches the identity of the person authenticating. This use case is known as "parallel authentication" because two authentication events are happening next to each other: the assertion, and the verification of the biometric (photo) by a trusted agent. The focus of FAL is primary on the assertions being passed from the IdP to the RP, so most authentication events occurring at the RP would not affect the FAL of the transaction.

At FAL3, holder-of-key transactions occur by verifying both the assertion from the IdP as well as the subscriber's presentation of proof of their personal key attested to in the assertion, which is another form of parallel authentication.

C.5.6 Brokered Identity Management

Some federated identity architectures are based on brokered identity management described in Section 5.1.4, where a single broker intermediates transactions between registered IdPs and RPs. In this architecture, each entity in the system only has to register with one broker in order to interoperate with everyone else in the system.

Recent advances in automated registration processes have made IdP/RP integrations much less onerous than they used to be. Since it is now more possible for an IdP and RP to register with each other in a very short amount of time, or without any manual intervention, the value of a broker solely as an integration point is much less.

The use of a broker can also blind the participants of the transaction from each other. Specifically, an IdP can authenticate a subscriber without knowledge of which RP requested the authentication event, and that an RP need not know which IdP a subscriber used to authenticate to the broker. While brokered identity management systems may appear to protect privacy by blinding an IdP from an RP, keep in mind that the broker

itself is aware of all the parties involved in the transaction, and in some cases can see personally identifiable information about subscribers. Brokered identity management systems do not prevent subscriber tracking all together, they merely shift the ability to track subscribers away from the IdPs and RPs and on to the broker.

Additionally, because brokers have access to active and valid identity assertions, they are capable of impersonating subscribers at RPs. The broker can also effectively impersonate any RP to an IdP, and any IdP to an RP. This power increases the risk inherent in the entire architecture, since the broker represents a single point of failure which, if it is compromised, can in turn compromise every participant in the system.

NIST has been promoting privacy-enhancing technology in the brokered identity management space through the Privacy-Enhanced Identity Federation project. This NIST building block outlines a set of goals which would constitute a new kind of brokered architecture. This architecture leverages a broker which cannot impersonate or track subscribers. This architecture is still theoretical and may allow for a privacy-preserving and secure version of brokered identity management in the future.

C.5.7 Communicating xAL

The value of the FAL for a given federation transaction should be inherently detectable by the nature of the transaction itself. Namely, the RP can tell if the assertion is encrypted and it will know if it has prompted for a secondary key-based authenticator. However, only the IdP inherently knows the IAL and AAL for the subscriber. The IdP can communicate that information to the RP in the assertion by using a format such as Vectors of Trust or the SAML Authentication Context, in combination with an appropriate trust framework. Since the RP has no way of directly verifying the IAL or AAL being asserted, it must trust that the IdP is asserting accurate and valid information regarding the subscriber.

It is still up to the RP to decide whether any given subscriber can perform an action within the RP's system. An RP could decide that most operations are allowable for people logging in at AAL2, for instance, but certain sensitive applications would require AAL3. Or an RP could request an FAL3 assertion, along with its key reference, but only prompt the subscriber for their key if a privileged operation required such escalation. Until such time that the secondary key is proofed by the RP, the subscriber is operating at FAL2.

C.6 Educational Resources

All specifications for identity federation standards mentioned in this resource guide are freely available online:

OAuth 2 (RFC 6749)
OpenID Connect
Security Assertion Markup Language

These specifications outline multiple, sometimes mutually exclusive, ways to implement federated identity. Therefore, it's important to read the specifications in their entirety before creating an implementation and to follow community best practices.

Federation standards communities actively track known vulnerabilities in existing standards.

The IETF lists OAuth Security Concerns in in RFC 6819 and hosts a working group to track OAuth standards and vulnerabilities.

The OpenID Foundation lists OpenID Connect security concerns within the specification itself and hosts a working group which actively tracks vulnerabilities.

OASIS has published SAML Privacy and Security Considerations and hosts a mailing list to track SAML vulnerabilities.

C.6.1 Communicating with Stakeholders

Stakeholders need to be aware that selecting an FAL is part of a larger risk- and resource-management process. While it is tempting for stakeholders to request the highest level of security, that is not always in the best interest of the organization. Federated identity projects at higher FALs can be long and complicated, and such complications can take resources away from other work that a security team could be doing that would be of greater benefit to the organization.

Many organizations today operate at FAL1, which is sufficient for most use cases. FAL1 is the industry standard, and there are many libraries and off-the-shelf products that can help an organization implement an FAL1 conformant federated identity system.

Conformance to FAL2 or FAL3 is appropriate for some business cases where there is a risk of fraudulent activity which would be prevented by token encryption, or when the transactions protected by the login is of particularly high value to warrant the additional complexity.