Withdrawn Draft

Warning Notice

The attached draft document has been withdrawn and is provided solely for historical purposes. It has been followed by the document identified below.

Withdrawal Date August 1, 2025

Original Release Date August 21, 2024

The attached draft document is followed by:

Status	Final
Series/Number	NIST SP 800-63B-4
Title	Digital Identity Guidelines: Authentication and Authenticator Management
Publication Date	July 2025
DOI	https://doi.org/10.6028/NIST.SP.800-63B-4
CSRC URL	https://csrc.nist.gov/pubs/sp/800/63/b/4/final
Additional Information	https://www.nist.gov/identity-access-management/nist-special-publication-800-63-digital-identity-guidelines

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

Check for updates

NIST Special Publication
NIST SP 800-63B-4 2pd

Digital Identity Guidelines

Authentication and Authenticator Management

Second Public Draft

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

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

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

NIST Special Publication
NIST SP 800-63B-4 2pd

Digital Identity Guidelines

Authentication and Authenticator Management

Second Public Draft

David Temoshok
Naomi Lefkovitz
Ryan Galluzzo
Applied Cybersecurity Division
Information Technology Laboratory

Yee-Yin Choong
Information Access Division
Information Technology Laboratory

Andrew Regenscheid
Computer Security Division
Information Technology Laboratory

James L. Fenton
Altmode Networks

Justin P. Richer
Bespoke Engineering

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

August 2024

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

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

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

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

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

Authority

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

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

NIST Technical Series Policies

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

Publication History

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

How to Cite this NIST Technical Series Publication

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

Author ORCID iDs

David Temoshok: 0000-0001-6195-0331
James L. Fenton: 0000-0002-2344-4291
Yee-Yin Choong: 0000-0002-3889-6047
Naomi Lefkovitz: 0000-0003-3777-3106
Andrew Regenscheid: 0000-0002-3930-527X
Ryan Galluzzo: 0000-0003-0304-4239
Justin P. Richer: 0000-0003-2130-5180

Public Comment Period

August 21, 2024 - October 7, 2024

Submit Comments

mailto:dig-comments@nist.gov

Additional Information

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

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

Abstract

This guideline focuses on the authentication of subjects who interact with government information systems over networks to establish that a given claimant is a subscriber who has been previously authenticated. The result of the authentication process may be used locally by the system performing the authentication or may be asserted elsewhere in a federated identity system. This document defines technical requirements for each of the three authenticator assurance levels. The guidelines are not intended to constrain the development or use of standards outside of this purpose. This publication supersedes NIST Special Publication (SP) 800-63B.

Keywords

authentication; authentication assurance; credential service provider; digital authentication; digital credentials; electronic authentication; electronic credentials; passwords.

Reports on Computer Systems Technology

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

Note to Reviewers

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

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

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

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

- Address comments received in response to the IPD of Revision 4 of SP 800-63

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

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

1. Authentication and Authenticator Management
    - Are the syncable authenticator requirements sufficiently defined to allow for reasonable risk-based acceptance of syncable authenticators for public and enterprise-facing uses?
    - Are there additional recommended controls that should be applied? Are there specific implementation recommendations or considerations that should be captured?
    - Are wallet-based authentication mechanisms and "attribute bundles" sufficiently described as authenticators? Are there additional requirements that need to be added or clarified?
2. General
    - What specific implementation guidance, reference architectures, metrics, or other supporting resources could enable more rapid adoption and implementation of this and future iterations of the Digital Identity Guidelines?
    - What applied research and measurement efforts would provide the greatest impacts on the identity market and advancement of these guidelines?

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

Call for Patent Claims

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

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

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

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

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

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

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

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

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

Table of Contents

List of Tables

List of Figures

Preface

This publication and its companion volumes — [SP800-63], [SP800-63A], and [SP800-63C] — provide technical guidelines for organizations to implement digital identity services.

This document, SP 800-63B, provides requirements to credential service providers (CSPs) for remote user authentication at each of three Authentication Assurance Levels (AALs).

Acknowledgments

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

1. Introduction

This section is informative.

Authentication is the process of determining the validity of one or more authenticators used to claim a digital identity by establishing that a subject attempting to access a digital service is in control of the secrets used to authenticate. If return visits are applicable to a service, successful authentication provides reasonable risk-based assurance that the subject accessing the service today is the same as the one who previously accessed the service. One-time services (where the subscriber will only ever access the service once) do not necessarily require the issuance of authenticators to support persistent digital authentication.

The authentication of claimants is central to the process of associating a subscriber with their online activity as recorded in their *subscriber account*, which is maintained by a *credential service provider* (CSP). Authentication is performed by verifying that the claimant controls one or more *authenticators* (called *tokens* in some earlier editions of SP 800-63) associated with a given subscriber account. The authentication process is conducted by a *verifier*, which is a role of the CSP or — in federated authentication — of an identity provider (IdP). Upon successful authentication, the verifier asserts an *identifier* to the relying party (RP). Optionally, the verifier may assert additional attributes to the RP.

This document provides recommendations on types of authentication processes, including choices of authenticators, that may be used at various *Authentication Assurance Levels* (AALs). It also provides recommendations on events that may occur during the lifetime of authenticators, including initial issuance, maintenance, and invalidation in the event of loss or theft of the authenticator.

This technical guideline applies to the digital authentication of subjects to systems over a network. It also requires that verifiers and RPs participating in authentication protocols be authenticated to claimants to assure the identity of the services with which they are authenticating. It does not address the authentication of a person for physical access (e.g., to a building). However, some credentials used for digital access may also be used for physical access authentication as described in [SP800-116].

AALs characterizes the strength of an authentication transaction as an ordinal category. Stronger authentication (i.e., a higher AAL) requires malicious actors to have better capabilities and to expend greater resources to successfully subvert the authentication process. Authentication at higher AALs can effectively reduce the risk of attacks. A high-level summary of the technical requirements for each of the AALs is provided below; see Sec. 2 and Sec. 3 of this document for specific normative requirements.

Authentication Assurance Level 1: AAL1 provides basic confidence that the claimant controls an authenticator bound to the subscriber account being authenticated. AAL1 requires only single-factor authentication using a wide range of available authentication technologies. However, it is recommended that applications assessed at AAL1 offer multi-factor authentication options. Successful authentication requires that the claimant prove possession and control of the authenticator.

Authentication Assurance Level 2: AAL2 provides high confidence that the claimant controls one or more authenticators bound to the subscriber account being authenticated. Proof of the possession and control of two distinct *authentication factors* is required. Applications assessed at AAL2 must offer a phishing-resistant authentication option.

Authentication Assurance Level 3: AAL3 provides very high confidence that the claimant controls one or more authenticators bound to the subscriber account being authenticated. Authentication at AAL3 is based on the proof of possession of a key through the use of a public-key cryptographic protocol. AAL3 authentication requires a hardware-based authenticator with a non-exportable private key and a phishing-resistant authenticator (see Sec. 3.2.5); the same device may fulfill both requirements. To authenticate at AAL3, claimants are required to prove possession and control of two distinct authentication factors.

When a session has been authenticated at a given AAL and a higher AAL is required, an authentication process may also provide step-up authentication to raise the session's AAL.

1.1. Notations

This guideline uses the following typographical conventions in text:

- Specific terms in **CAPITALS** represent normative requirements. When these same terms are not in **CAPITALS**, the term does not represent a normative requirement.
  - The terms "**SHALL**" and "**SHALL NOT**" indicate requirements to be strictly followed in order to conform to the publication and from which no deviation is permitted.
  - The terms "**SHOULD**" and "**SHOULD NOT**" indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.
  - The terms "**MAY**" and "**NEED NOT**" indicate a course of action that is permissible within the limits of the publication.
  - The terms "**CAN**" and "**CANNOT**" indicate a material, physical, or causal possibility and capability or — in the negative — the absence of that possibility or capability.

1.2. Document Structure

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

- Section 1 introduces the document. This section is *informative*.
- Section 2 describes requirements for Authentication Assurance Levels. This section is *normative*.
- Section 3 describes requirements for authenticator and verifier requirements. This section is *normative*.
- Section 4 describes requirements for authenticator event management. This section is *normative*.
- Section 5 describes requirements for session management. This section is *normative*.
- Section 6 provides security considerations. This section is *informative*.
- Section 7 provides privacy considerations. This section is *informative*.
- Section 8 provides usability considerations. This section is *informative*.
- Section 9 provides equity considerations. This section is *informative*.
- The References section lists publications that are referred to in this document. This section is *informative*.
- Appendix A discusses the strength of passwords. This appendix is *informative*.
- Appendix B discusses syncable authenticators. This appendix is *normative*.
- Appendix C contains a selected list of abbreviations used in this document. This appendix is *informative*.
- Appendix D contains a glossary of selected terms used in this document. This appendix is *informative*.
- Appendix E contains a summarized list of changes in this document’s history. This appendix is *informative*.

2. Authentication Assurance Levels

This section is normative.

To satisfy the requirements of a given AAL and be recognized as a subscriber, a claimant **SHALL** authenticate to an RP or IdP as described in [SP800-63C] with a process whose strength is equal to or greater than the requirements at that level. The authentication process results in an identifier that uniquely identifies the subscriber each time they authenticate to that RP. The identifier **MAY** be pseudonymous. Other attributes that identify the subscriber as a unique subject **MAY** also be provided.

Detailed normative requirements for authenticators and verifiers at each AAL are provided in Sec. 3. See [SP800-63] Sec. 3 for details on how to choose the most appropriate AAL.

Personal information collected during and after identity proofing (described in [SP800-63A]) **MAY** be made available to the subscriber by the digital identity service through the subscriber account. The release or online availability of any personally identifiable information (PII) or other personal information by federal agencies requires multi-factor authentication in accordance with [EO13681]. Therefore, federal agencies **SHALL** select a minimum of AAL2 when PII or other personal information is made available online.

At all AALs, pre-authentication checks **MAY** be used to lower the risk of misauthentication. For example, authentication from an unexpected geolocation or IP address block (e.g., a cloud service) might prompt the use of additional risk-based controls. Where used, CSPs or verifiers **SHALL** assess their pre-authentication checks for efficacy and to identify and mitigate potential disparate impacts on their user populations. CSPs or verifiers **SHALL** include pre-authentication checks in the authentication privacy risk assessment. Pre-authentication checks do not impact or change the AAL of a transaction or substitute for an authentication factor.

Throughout this document, [FIPS140] requirements are satisfied by the latest edition of FIPS 140. Legacy FIPS 140 certifications **MAY** also be used while still valid.

2.1. Authentication Assurance Level 1

AAL1 provides basic confidence that the claimant controls an authenticator bound to the subscriber account. AAL1 requires either single-factor or multi-factor authentication using a wide range of available authentication technologies. Verifiers **SHOULD** make multi-factor authentication options available at AAL1 and encourage their use. Successful authentication requires that the claimant prove possession and control of the authenticator through a secure authentication protocol.

2.1.1. Permitted Authenticator Types

AAL1 authentication **SHALL** use any of the following authentication types, which are further defined in Sec. 3:

- Password (Sec. 3.1.1): A memorizable secret typically chosen by the subscriber
- Look-up secret (Sec. 3.1.2): A secret determined by the claimant by looking up a prompted value in a list held by the subscriber
- Out-of-band device (Sec. 3.1.3): A secret sent or received through a separate communication channel with the subscriber
- Single-factor one-time password (OTP) (Sec. 3.1.4): A one-time secret obtained from a device or application held by the subscriber
- Multi-factor OTP (Sec. 3.1.5): A one-time secret obtained from a device or application held by the subscriber that requires activation by a second authentication factor
- Single-factor cryptographic authentication (Sec. 3.1.6): Proof of possession and control via an authentication protocol of a cryptographic key held by the subscriber.
- Multi-Factor cryptographic authentication (Sec. 3.1.7): Proof of possession and control via an authentication protocol of a cryptographic key held by the subscriber that requires activation by a second authentication factor

2.1.2. Authenticator and Verifier Requirements

Authenticators used at AAL1 **SHALL** use *approved cryptography*. In other words, they must use approved algorithms, but the implementation need not be validated under [FIPS140].

Communication between the claimant and verifier **SHALL** occur via one or more authenticated protected channels.

Cryptography used by verifiers operated by or on behalf of federal agencies at AAL1 **SHALL** be validated to meet the requirements of [FIPS140] Level 1.

2.1.3. Reauthentication

These guidelines provide for two types of timeouts, which are further described in Sec. 5.2:

1. An overall timeout limits the duration of an authenticated session to a specified period following authentication or a previous reauthentication.
2. An inactivity timeout terminates a session that has not had activity from the subscriber for a specified period.

Periodic reauthentication of subscriber sessions **SHALL** be performed, as described in Sec. 5.2. A definite reauthentication overall timeout **SHALL** be established, which **SHOULD** be no more than 30 days at AAL1. An inactivity timeout **MAY** be applied but is not required at AAL1.

2.2. Authentication Assurance Level 2

AAL2 provides high confidence that the claimant controls one or more authenticators that are bound to the subscriber account. Proof of possession and control of two distinct authentication factors is required through the use of secure authentication protocols. Approved cryptographic techniques are required.

2.2.1. Permitted Authenticator Types

At AAL2, authentication **SHALL** use either a multi-factor authenticator or a combination of two single-factor authenticators. A multi-factor authenticator requires two factors to execute a single authentication event, such as a cryptographically secure device with an integrated biometric sensor that is required to activate the device. Authenticator requirements are specified in Sec. 3.

When a multi-factor authenticator is used, any of the following **MAY** be used:

- Multi-factor Out-of-band authenticator (Sec. 3.1.3.4)
- Multi-factor OTP (Sec. 3.1.5)
- Multi-factor cryptographic authentication (Sec. 3.1.7)

When a combination of two single-factor authenticators is used, the combination **SHALL** include a password (Sec. 3.1.1) and one *physical authenticator* (i.e., “something you have”) from the following list:

- Look-up secret (Sec. 3.1.2)
- Out-of-band device (Sec. 3.1.3)
- Single-factor OTP (Sec. 3.1.4)
- Single-factor cryptographic authentication (Sec. 3.1.6)

A biometric characteristic is not recognized as an authenticator by itself. When biometric authentication meets the requirements in Sec. 3.2.3, a physical authenticator is authenticated along with the biometric. The physical authenticator then serves as “something you have,” while the biometric match serves as “something you are.” When a biometric comparison is used as an activation factor for a multi-factor authenticator, the authenticator itself serves as the physical authenticator.

2.2.2. Authenticator and Verifier Requirements

Authenticators used at AAL2 **SHALL** use approved cryptography. Cryptographic authenticators procured by federal agencies **SHALL** be validated to meet the requirements of [FIPS140] Level 1. At least one authenticator used at AAL2 **SHALL** be replay-resistant, as described in Sec. 3.2.7. Authentication at AAL2 **SHOULD** demonstrate authentication intent from at least one authenticator, as discussed in Sec. 3.2.8.

Communication between the claimant and verifier **SHALL** occur via one or more authenticated protected channels.

Cryptography used by verifiers operated by or on behalf of federal agencies at AAL2 **SHALL** be validated to meet the requirements of [FIPS140] Level 1.

When a biometric factor is used in authentication at AAL2, the performance requirements stated in Sec. 3.2.3 **SHALL** be met, and the verifier **SHALL** determine that the biometric sensor and subsequent processing meet these requirements.

Verifiers **SHALL** offer at least one phishing-resistant authentication option at AAL2, as described in Sec. 3.2.5. Federal agencies **SHALL** require their staff, contractors, and partners to use phishing-resistant authentication to access federal information systems. In all cases, verifiers **SHOULD** encourage the use of phishing-resistant authentication at AAL2 whenever practical since phishing is a significant threat vector.

2.2.3. Reauthentication

Periodic reauthentication of subscriber sessions **SHALL** be performed as described in Sec. 5.2. A definite reauthentication overall timeout **SHALL** be established, which **SHOULD** be no more than 24 hours at AAL2. The inactivity timeout **SHOULD** be no more than 1 hour. When the inactivity timeout has occurred but the overall timeout has not yet occurred, the verifier **MAY** allow the subscriber to reauthenticate using only a successful password or biometric comparison in conjunction with the session secret.

2.3. Authentication Assurance Level 3

AAL3 provides very high confidence that the claimant controls authenticators that are bound to the subscriber account. Authentication at AAL3 is based on the proof of possession of a key through the use of a cryptographic protocol along with either an activation factor or a password. AAL3 authentication requires the use of a hardware-based authenticator that provides phishing resistance. Approved cryptographic techniques are required.

2.3.1. Permitted Authenticator Types

AAL3 authentication **SHALL** require one of the following authenticator combinations:

- Multi-factor cryptographic authentication (Sec. 3.1.7)
- Single-factor cryptographic authentication (Sec. 3.1.6) used in conjunction with a password (Sec. 3.1.1)

2.3.2. Authenticator and Verifier Requirements

Authenticators used at AAL3 **SHALL** use approved cryptography. Communication between the claimant and verifier **SHALL** occur via one or more authenticated protected channels. The cryptographic authenticator used at AAL3 **SHALL** be hardware-based and **SHALL** provide phishing resistance, as described in Sec. 3.2.5. The cryptographic authentication protocol **SHALL** be replay-resistant as described in Sec. 3.2.7. All authentication and reauthentication processes at AAL3 **SHALL** demonstrate authentication intent from at least one authenticator as described in Sec. 3.2.8.

Multi-factor authenticators used at AAL3 **SHALL** be hardware cryptographic modules that are validated at [FIPS140] Level 2 or higher overall with at least [FIPS140] Level 3 physical security. Single-factor cryptographic authenticators used at AAL3 **SHALL** be validated at [FIPS140] Level 1 or higher overall with at least [FIPS140] Level 3 physical security. AAL3 protects the verifier from compromise through the use of public-key cryptography since the verifier does not possess the private key required to authenticate.

Cryptography used by verifiers at AAL3 **SHALL** be validated at [FIPS140] Level 1 or higher.

Hardware-based authenticators and verifiers at AAL3 **SHOULD** resist relevant side-channel (e.g., timing and power-consumption analysis) attacks.

When a biometric factor is used in authentication at AAL3, the verifier **SHALL** determine that the biometric sensor and subsequent processing meet the performance requirements stated in Sec. 3.2.3.

2.3.3. Reauthentication

Periodic reauthentication of subscriber sessions **SHALL** be performed, as described in Sec. 5.2. At AAL3, the overall timeout for reauthentication **SHALL** be no more than 12 hours. The inactivity timeout **SHOULD** be no more than 15 minutes. Unlike AAL2, AAL3 reauthentication requirements are the same as for initial authentication at AAL3.

2.4. General Requirements

The following requirements apply to authentication at all AALs.

2.4.1. Security Controls

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

2.4.2. Records Retention Policy

The CSP **SHALL** comply with its respective records retention policies in accordance with applicable laws, regulations, and policies, including any National Archives and Records Administration (NARA) records retention schedules that may apply. If the CSP opts to retain records in the absence of mandatory requirements, the CSP **SHALL** conduct a risk management process, including assessments of privacy and security risks, to determine how long records should be retained and **SHALL** inform the subscriber of that retention policy.

2.4.3. Privacy Requirements

The CSP **SHALL** employ appropriately tailored privacy controls defined in [SP800-53] or an equivalent industry standard.

If CSPs process attributes for purposes other than identity service (i.e., identity proofing, authentication, or attribute assertions), related fraud mitigation, or compliance with laws or legal process, CSPs **SHALL** implement measures to maintain predictability and manageability commensurate with the privacy risks that arise from the additional processing. Examples of such measures include providing clear notice, obtaining subscriber consent, and enabling the selective use or disclosure of attributes. When CSPs use consent measures, CSPs **SHALL NOT** make consent for the additional processing a condition of the identity service.

Regardless of whether the CSP is an agency or private-sector provider, the following requirements apply to a federal agency that offers or uses the authentication service:

1. The agency **SHALL** consult with their Senior Agency Official for Privacy (SAOP) and conduct an analysis to determine whether the collection of PII to issue or maintain authenticators triggers the requirements of the *Privacy Act of 1974* [PrivacyAct] (see Sec. 7.4).
2. The agency **SHALL** publish a System of Records Notice (SORN) to cover such collections, as applicable.

3. The agency **SHALL** consult with its SAOP and conduct an analysis to determine whether the collection of PII to issue or maintain authenticators triggers the requirements of the *E-Government Act of 2002* [E-Gov].

4. The agency **SHALL** publish a Privacy Impact Assessment (PIA) to cover such collection, as applicable.

2.4.4. Redress Requirements

The CSP and verifier **SHALL** provide mechanisms for the redress of subscriber complaints and for problems that arise from subscriber authentication processes as described in Sec. 5.6 of SP 800-63. These mechanisms **SHALL** be easy for subscribers to find and use. The CSP **SHALL** assess the mechanisms for efficacy in resolving complaints or problems.

2.5. Summary of Requirements

Figure 1 provides a non-normative summary of the requirements for each of the AALs.

Requirement	AAL1	AAL2	AAL3
Permitted Authenticator Types	• *Any AAL2 or AAL3 authenticator* • Password • Look-up secret • Out-of-band • SF OTP • SF cryptographic	• *Any AAL3 authenticator* • MF out-of-band • MF OTP • Password plus: * Look-up secret * Out-of-band * SF OTP	• MF cryptographic • SF cryptographic plus password
FIPS 140 Validation	Level 1 (Government agency verifiers)	Level 1 (Government agency authenticators and verifiers)	• Level 3 physical security • Level 2 overall (MF cryptographic) • Level 1 overall (verifiers and SF cryptographic)
Reauthentication	30 days overall	24 hours overall 1 hour inactivity	12 hours overall 15 minutes of inactivity
Phishing Resistance	Not required	Recommended; must be available	Required
Replay Resistance	Not required	Required	Required
Authentication Intent	Not required	Recommended	Required

Fig. 1. Summary of requirements by AAL

3. Authenticator and Verifier Requirements

This section is normative.

This section provides detailed requirements that are specific to each type of authenticator. With the exception of the reauthentication requirements specified in Sec. 2 and the requirement for phishing resistance at AAL3 described in Sec. 3.2.5, the technical requirements for each authenticator type are the same regardless of the AAL at which the authenticator is used.

In many circumstances, users need to share devices that are used in authentication processes, such as a family phone that receives OTPs. In public-facing applications, CSPs **SHOULD NOT** prevent a device from being registered as an authenticator by multiple subscribers. However, they **MAY** establish appropriate restrictions to prevent large-scale fraud or misuse.

Authentication, authenticator binding (see in Sec. 4), and session maintenance (see in Sec. 5) are based on proof of possession of one or more types of secrets, as shown in Table 1.

Table 1. Summary of Secrets (non-normative)

Type of Secret	Purpose	Reference Section
Password	A subscriber-chosen secret used as an authentication factor	3.1.1
Look-up secret	A secret issued by a verifier and used only once to prove possession of the secret	3.1.2
Out-of-band secret	A short-lived secret generated by a verifier and independently sent to a subscriber's device to verify its possession	3.1.3
One-time passcodes (OTP)	A secret generated by an authenticator and used only once to prove possession of the authenticator	3.1.4, 3.1.5
Activation secret	A password that is used locally as an activation factor for a multi-factor authenticator	3.2.10

Long-term authenticator secret	A secret embedded in a physical authenticator to allow it to function for authentication	4.1
Recovery code	A secret issued to the subscriber to allow them to recover an account at which they are no longer able to authenticate	4.2
Session secret	A secret issued by the verifier at authentication and used to establish the continuity of authenticated sessions	5.1

3.1. Requirements by Authenticator Type

The following requirements apply to specific authenticator types.

3.1.1. Passwords

A password (sometimes referred to as a *passphrase* or, if numeric, a *PIN*) is a secret value intended to be chosen and either memorized or recorded by the subscriber. Passwords must be of sufficient complexity and secrecy that it would be impractical for an attacker to guess or otherwise discover the correct secret value. A password is "something you know".

The requirements in this section apply to centrally verified passwords that are used as independent authentication factors and sent over an authenticated protected channel to the verifier of a CSP. Passwords used locally as an activation factor for a multi-factor authenticator are referred to as *activation secrets* and discussed in Sec. 3.2.10.

Passwords are not phishing-resistant.

3.1.1.1. Password Authenticators

Passwords **SHALL** either be chosen by the subscriber or assigned randomly by the CSP.

If the CSP disallows a chosen password because it is on a blocklist of commonly used, expected, or compromised values (see Sec. 3.1.1.2), the subscriber **SHALL** be required to choose a different password. Other complexity requirements for passwords **SHALL NOT** be imposed. A rationale for this is presented in Appendix A, *Strength of Passwords*.

3.1.1.2. Password Verifiers

The following requirements apply to passwords:

1. Verifiers and CSPs **SHALL** require passwords to be a minimum of eight characters in length and **SHOULD** require passwords to be a minimum of 15 characters in length.
2. Verifiers and CSPs **SHOULD** permit a maximum password length of at least 64 characters.
3. Verifiers and CSPs **SHOULD** accept all printing ASCII [RFC20] characters and the space character in passwords.
4. Verifiers and CSPs **SHOULD** accept Unicode [ISO/ISC 10646] characters in passwords. Each Unicode code point **SHALL** be counted as a single character when evaluating password length.
5. Verifiers and CSPs **SHALL NOT** impose other composition rules (e.g., requiring mixtures of different character types) for passwords.
6. Verifiers and CSPs **SHALL NOT** require users to change passwords periodically. However, verifiers **SHALL** force a change if there is evidence of compromise of the authenticator.
7. Verifiers and CSPs **SHALL NOT** permit the subscriber to store a hint that is accessible to an unauthenticated claimant.
8. Verifiers and CSPs **SHALL NOT** prompt subscribers to use knowledge-based authentication (KBA) (e.g., "What was the name of your first pet?") or security questions when choosing passwords.
9. Verifiers **SHALL** verify the entire submitted password (i.e., not truncate it).

If Unicode characters are accepted in passwords, the verifier **SHOULD** apply the normalization process for stabilized strings using either the NFKC or NFKD normalization defined in Sec. 12.1 of *Unicode Normalization Forms* [UAX15]. This process is applied before hashing the byte string that represents the password. Subscribers choosing passwords that contain Unicode characters **SHOULD** be advised that some endpoints may represent some characters differently, which would affect their ability to authenticate successfully.

When processing a request to establish or change a password, verifiers **SHALL** compare the prospective secret against a blocklist that contains known commonly used, expected, or compromised passwords. The entire password **SHALL** be subject to comparison, not substrings or words that might be contained therein. For example, the list **MAY** include but is not limited to:

- Passwords obtained from previous breach corpuses
- Dictionary words
- Context-specific words, such as the name of the service, the username, and derivatives thereof

If the chosen password is found on the blocklist, the CSP or verifier **SHALL** require the subscriber to select a different secret and **SHALL** provide the reason for rejection. Since the blocklist is used to defend against brute-force attacks and unsuccessful attempts are rate-limited, as described below, the blocklist **SHOULD** be of sufficient size to prevent subscribers from choosing passwords that attackers are likely to guess before reaching the attempt limit.

> Excessively large blocklists are of little incremental security benefit because the blocklist is used to defend against online attacks, which are already limited by the throttling requirements described in Sec. 3.2.2.

Verifiers **SHALL** offer guidance to the subscriber to assist the user in choosing a strong password. This is particularly important following the rejection of a password on the blocklist as it discourages trivial modification of listed weak passwords [Blocklists].

Verifiers **SHALL** implement a rate-limiting mechanism that effectively limits the number of failed authentication attempts that can be made on the subscriber account, as described in Sec. 3.2.2.

Verifiers **SHALL** allow the use of password managers. Verifiers **SHOULD** permit claimants to use the “paste” functionality when entering a password to facilitate their use. Password managers have been shown to increase the likelihood that users will choose stronger passwords, particularly if the password managers include password generators [Managers].

To assist the claimant in successfully entering a password, the verifier **SHOULD** offer an option to display the secret — rather than a series of dots or asterisks — while it is entered and until it is submitted to the verifier. This allows the claimant to confirm their entry if they are in a location where their screen is unlikely to be observed. The verifier **MAY** also permit the claimant’s device to display individual entered characters for a short time after each character is typed to verify the correct entry. This is common on mobile devices.

Verifiers **MAY** make allowances for mistyping, such as removing leading and trailing whitespace characters before verification or allowing the verification of passwords with differing cases for the leading character, provided that passwords remain at least the required minimum length after such processing.

Verifiers and CSPs **SHALL** use approved encryption and an authenticated protected channel when requesting passwords.

Verifiers **SHALL** store passwords in a form that is resistant to offline attacks. Passwords **SHALL** be salted and hashed using a suitable password hashing scheme. Password hashing schemes take a password, a salt, and a cost factor as inputs and generate a password hash. Their purpose is to make each password guess more expensive for an attacker who has obtained a hashed password file, thereby making the cost of a guessing attack high or prohibitive. The chosen cost factor **SHOULD** be as high as practical without negatively impacting verifier performance. It **SHOULD** be increased over time to account for increases in computing performance. An approved password hashing scheme published in the latest revision of [SP800-132] or updated NIST guidelines on password hashing schemes **SHOULD** be used. The chosen output length of the password verifier, excluding the salt and versioning information, **SHOULD** be the same as the length of the underlying password hashing scheme output.

The salt **SHALL** be at least 32 bits in length and chosen to minimize salt value collisions among stored hashes. Both the salt value and the resulting hash **SHALL** be stored for each password. A reference to the password hashing scheme used, including the work factor, **SHOULD** be stored for each password to allow migration to new algorithms and work factors. For example, for the Password-Based Key Derivation Function 2 (PBKDF2) [SP800-132], the cost factor is an iteration count: the more times that the PBKDF2 function is iterated, the longer it takes to compute the password hash.

In addition, verifiers **SHOULD** perform an additional iteration of a keyed hashing or encryption operation using a secret key known only to the verifier. If used, this key value **SHALL** be generated by an approved random bit generator, as described in Sec. 3.2.12. The secret key value **SHALL** be stored separately from the hashed passwords. It **SHOULD** be stored and used within a hardware-protected area, such as a hardware security module or trusted execution environment (TEE). With this additional iteration, brute-force attacks on the hashed passwords are impractical as long as the secret key value remains secret.

3.1.2. Look-Up Secrets

A look-up secret authenticator is a physical or electronic record that stores a set of secrets shared between the claimant and the CSP. The claimant uses the authenticator to look up the appropriate secrets needed to respond to a prompt from the verifier. For example, the verifier could ask a claimant to provide a specific subset of the numeric or character strings printed on a card in table format. A typical application of look-up secrets is for one-time saved recovery codes (see Sec. 4.2.1.1) that the subscriber stores for use if another authenticator is lost or malfunctions. A look-up secret is "something you have."

Look-up secrets are not phishing-resistant.

3.1.2.1. Look-Up Secret Authenticators

CSPs that create look-up secret authenticators **SHALL** use an approved random bit generator, as described in Sec. 3.2.12, to generate the list of secrets and **SHALL** deliver the authenticator list securely to the subscriber (e.g., in an in-person session, via a session authenticated by the subscriber at AAL2 or higher, or through the postal mail to a contact address). Look-up secrets **SHALL** be at least six decimal digits (or equivalent) in length. Additional requirements described in Sec. 3.1.2.2 may also apply, depending on their length.

Look-up secrets **MAY** be distributed by the CSP in person, by postal mail to a contact address for the subscriber, or by online distribution. If distributed online, look-up secrets **SHALL** be distributed over a secure channel in accordance with the post-enrollment binding requirements in Sec. 4.1.2.

3.1.2.2. Look-Up Secret Verifiers

Verifiers of look-up secrets **SHALL** prompt the claimant for the next secret from their authenticator or a specific (e.g., numbered) secret. A secret from a look-up secret authenticator **SHALL** be used successfully only once. If the look-up secret is derived from a grid card, each grid cell **SHOULD** be used only once, which limits the number of authentications that can be accomplished using look-up secrets. Otherwise, a very long list of secrets is required.

Verifiers **SHALL** store look-up secrets in a form that is resistant to offline attacks. All look-up secrets **SHALL** be stored in a hashed form using an approved hashing function.

Look-up secrets **SHALL** be at least six decimal digits (or equivalent) in length, as specified in Sec. 3.1.2.1. Look-up secrets that are shorter than specified lengths have additional verification requirements as follows:

- Look-up secrets that are shorter than the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication) **SHALL** be stored in a salted and hashed form using a suitable password hashing scheme, as described in Sec. 3.1.1.2. The salt value **SHALL** be at least 32 bits in length and arbitrarily chosen to minimize salt value collisions among stored hashes. Both the salt value and the resulting hash **SHALL** be stored for each look-up secret. Because look-up secrets are generated using a random bit generator, the work factor for the password hashing scheme **MAY** be small.

- The verifier **SHALL** implement a rate-limiting mechanism that effectively limits the number of failed authentication attempts that can be made on the subscriber account, as described in Sec. 3.2.2.

The verifier **SHALL** use approved encryption and an authenticated protected channel when requesting look-up secrets.

3.1.3. Out-of-Band Devices

An out-of-band authenticator is a physical device that is uniquely addressable and can communicate securely with the verifier over a distinct communications channel, referred to as the secondary channel. The device is possessed and controlled by the claimant and supports private communication over this secondary channel, which is separate from the primary channel for authentication. An out-of-band authenticator is "something you have."

Out-of-band authentication uses a short-term secret generated by the verifier. The secret securely binds the authentication operation on the primary and secondary channels and establishes the claimant's control of the out-of-band device.

Out-of-band authentication is not phishing-resistant.

The out-of-band authenticator can operate in one of the following ways:

- The claimant transfers a secret received by the out-of-band device via the secondary channel to the verifier using the primary channel. For example, the claimant may receive the secret (typically a 6-digit code) on their mobile device and type it into their authentication session. This method is shown in Fig. 2.

- The claimant transfers a secret received via the primary channel to the out-of-band device for transmission to the verifier via the secondary channel. For example, the claimant may view the secret on their authentication session and either type it into an app on their mobile device or use a technology such as a barcode or QR code to effect the transfer. This method is shown in Fig. 3.

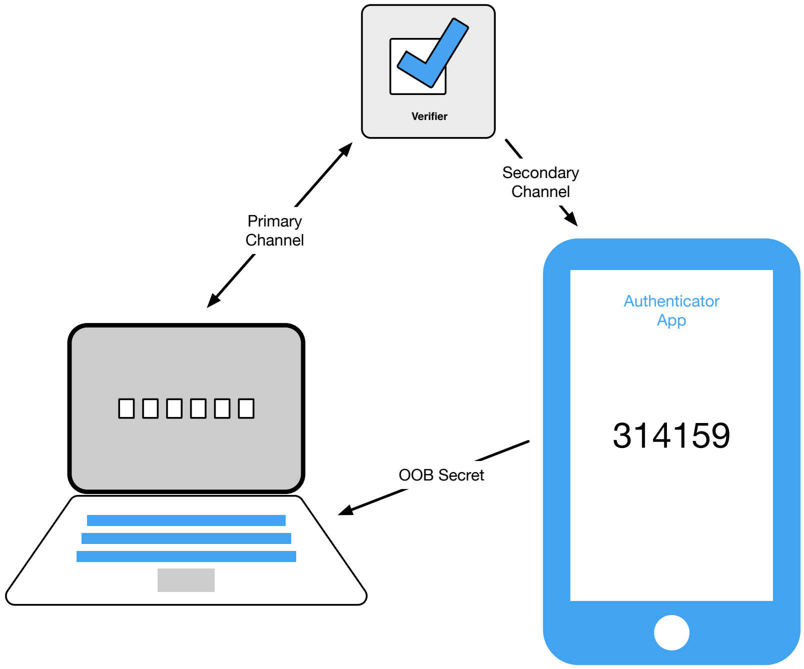

Fig. 2. Transfer of Secret to Primary Device

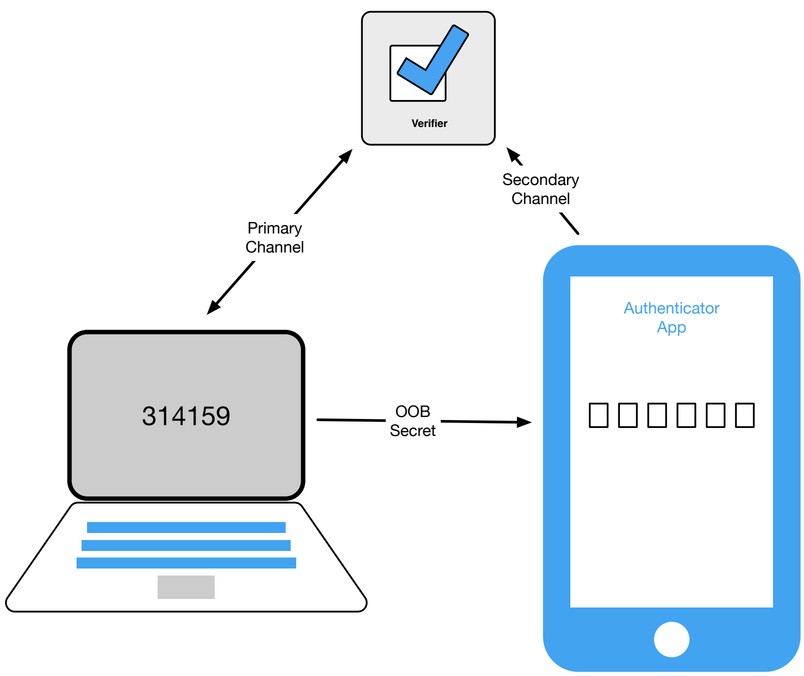

Fig. 3. Transfer of Secret to Out-of-band Device

> A third method of out-of-band authentication compares secrets received from the primary and secondary channels and requests approval on the secondary channel. This method is no longer considered acceptable because it increased the likelihood that the subscriber would approve an authentication request without actually comparing the secrets as required. This has been observed with "authentication fatigue" attacks where an attacker (claimant) would generate many out-of-band authentication requests to the subscriber, who might approve one to eliminate the annoyance. For this reason, an authenticator that receives a push notification from the verifier and simply asks the claimant to approve the transaction (even if they provide some additional information about the authentication) does not meet the requirements of this section.

3.1.3.1. Out-of-Band Authenticators

The out-of-band authenticator **SHALL** establish a separate channel with the verifier to retrieve the out-of-band secret or authentication request. This channel is considered to be out-of-band with respect to the primary communication channel (even if it terminates on the same device), provided that the device does not leak information from one channel to the other without the claimant's authorization.

The out-of-band device **SHOULD** be uniquely addressable by the verifier. Communication over the secondary channel **SHALL** use approved encryption unless sent via the public switched telephone network (PSTN). For additional authenticator requirements that are specific to using the PSTN for out-of-band authentication, see Sec. 3.1.3.3.

Email **SHALL NOT** be used for out-of-band authentication because it may be vulnerable to:

- Accessibility using only a password
- Interception in transit or at intermediate mail servers
- Rerouting attacks, such as those caused by DNS spoofing

The out-of-band authenticator **SHALL** uniquely authenticate itself in one of the following ways when communicating with the verifier:

- Using approved cryptography, establish a mutually authenticated protected channel (e.g., client-authenticated TLS) to the verifier. Communication between the out-of-band authenticator and the verifier **MAY** use a trusted intermediary service to which each authenticates. The key **SHALL** be provisioned in a mutually authenticated session during authenticator binding, as described in Sec. 4.1.

- Authenticate to a public mobile telephone network using a SIM card or equivalent secret that uniquely identifies the subscriber. This method **SHALL** only be used if a secret is sent from the verifier to the out-of-band device via the PSTN (SMS or voice) or an encrypted instant messaging service.
- Use a wired connection to the PSTN that the verifier can call and dictate the out-of-band secret. For purposes of this definition, “wired connection” includes services such as cable providers that offer PSTN services through other wired media and fiber via analog telephone adapters.

For single-factor out-of-band authenticators, if a secret is sent by the verifier to the out-of-band device, the device **SHOULD NOT** display the authentication secret while it is locked by the owner (i.e., the device **SHOULD** require the presentation and verification of a PIN, passcode, or biometric characteristic to view). However, authenticators **SHOULD** indicate the receipt of an authentication secret on a locked device.

If the out-of-band authenticator requests approval over the secondary communication channel rather than by presenting a secret that the claimant transfers to the primary communication channel, it **SHALL** accept a transfer of the secret from the primary channel and send it to the verifier over the secondary channel to associate the approval with the authentication transaction. The claimant **MAY** perform the transfer manually and with the assistance of a representation, such as a barcode or QR code.

3.1.3.2. Out-of-Band Verifiers

For additional verification requirements that are specific to the PSTN, see Sec. 3.1.3.3.

The verifier waits for an authenticated protected channel to be established with the out-of-band authenticator and verifies its identifying key. The verifier **SHALL NOT** store the identifying key itself but **SHALL** use a verification method (e.g., an approved hash function or proof of possession of the identifying key) to uniquely identify the authenticator. Once authenticated, the verifier transmits the authentication secret to the authenticator. The connection with the out-of-band authenticator **MAY** be either manually initiated or prompted by a mechanism such as a push notification.

Depending on the type of out-of-band authenticator, one of the following **SHALL** take place:

Transfer of the secret from the secondary to the primary channel

As shown in Fig. 2, the verifier **MAY** signal the device that contains the subscriber’s authenticator to indicate a readiness to authenticate. It **SHALL** then transmit a random secret to the out-of-band authenticator and wait for the secret to be returned via the primary communication channel.

Transfer of the secret from the primary to the secondary channel

As shown in Fig. 3, the verifier **SHALL** display a random authentication secret to the claimant via the primary channel. It **SHALL** then wait for the secret to be returned via the secondary channel from the claimant's out-of-band authenticator. The verifier **MAY** additionally display an address, such as a phone number or VoIP address, for the claimant to use in addressing its response to the verifier.

In all cases, the authentication **SHALL** be considered invalid unless completed within 10 minutes. Verifiers **SHALL** accept a given authentication secret as valid only once during the validity period to provide replay resistance, as described in Sec. 3.2.7.

The verifier **SHALL** generate random authentication secrets that are at least six decimal digits (or equivalent) in length using an approved random bit generator as described in Sec. 3.2.12. If the authentication secret is less than 64 bits long, the verifier **SHALL** implement a rate-limiting mechanism that effectively limits the total number of consecutive failed authentication attempts that can be made on the subscriber account as described in Sec. 3.2.2. Generating a new authentication secret **SHALL NOT** reset the failed authentication count.

Out-of-band verifiers that send a push notification to a subscriber device **SHOULD** implement a reasonable limit on the rate or total number of push notifications that will be sent since the last successful authentication.

3.1.3.3. Authentication Using the Public Switched Telephone Network

Use of the PSTN for out-of-band verification is restricted as described in this section and Sec. 3.2.9. Setting or changing the pre-registered telephone number is considered to be the binding of a new authenticator and **SHALL** only occur as described in Sec. 4.1.2.

Some subscribers may be unable to use PSTN to deliver out-of-band authentication secrets in areas with limited telephone coverage (particularly without mobile phone service). Accordingly, verifiers **SHALL** ensure that alternative authenticator types are available to all subscribers and **SHOULD** remind subscribers of this limitation of PSTN out-of-band authenticators before binding one or more devices controlled by the subscriber.

Verifiers **SHOULD** consider risk indicators (e.g., device swap, SIM change, number porting, or other abnormal behavior) before using the PSTN to deliver an out-of-band authentication secret.

> Consistent with the restriction of authenticators in Sec. 3.2.9, NIST may adjust the restricted status of out-of-band authentication using the PSTN based on the evolution of the threat landscape and the technical operation of the PSTN.

3.1.3.4. Multi-Factor Out-of-Band Authenticators

Multi-factor out-of-band authenticators operate similarly to single-factor out-of-band authenticators (see Sec. 3.1.3.1). However, they require the presentation and verification of an activation factor (i.e., a password or a biometric characteristic) before allowing the claimant to complete the authentication transaction (i.e., before accessing or entering the authentication secret as appropriate for the authentication flow being used). Each use of the authenticator **SHALL** require the presentation of the activation factor.

Authenticator activation secrets **SHALL** meet the requirements of Sec. 3.2.10. A biometric activation factor **SHALL** meet the requirements of Sec. 3.2.3, including limits on the number of consecutive authentication failures. The password or biometric sample used for activation and any biometric data derived from the biometric sample (e.g., a probe produced through signal processing) **SHALL** be zeroized (erased) immediately after an authentication operation.

3.1.4. Single-Factor OTP

A single-factor OTP generates one-time passwords (OTPs). This category includes hardware devices and software-based OTP generators that are installed on devices such as mobile phones. These authenticators have an embedded secret that is used as the seed for generating OTPs and do not require activation through a second factor. The OTP is displayed on the authenticator and manually input for transmission to the verifier, thereby proving possession and control of the authenticator. A single-factor OTP authenticator is *something you have*.

Single-factor OTPs are similar to look-up secret authenticators except that the secrets are cryptographically and independently generated by the authenticator and the verifier and compared by the verifier. The secret is computed based on a nonce that may be time-based or from a counter on the authenticator and verifier.

OTP authentication is not phishing-resistant. [FIPS140] validation of OTP authenticators and verifiers is not required.

3.1.4.1. Single-Factor OTP Authenticators

Single-factor OTP authenticators and verifiers contain two persistent values: 1) a symmetric key that persists for the authenticator’s lifetime and 2) a nonce that is either changed each time the authenticator is used or is based on a real-time clock.

The secret key and its algorithm **SHALL** provide at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication). The nonce **SHALL** be of sufficient length to ensure that it is unique for each operation of the authenticator over its lifetime. If a subscriber needs to change the device on which a software-based OTP authenticator resides, they **SHOULD** bind the authenticator application on the new device to their subscriber account, as described in Sec. 4.1.2, and invalidate the authenticator application that will no longer be used.

The authenticator output is obtained using an approved block cipher or hash function to securely combine the key and nonce. In coordination with the verifier, the authenticator **MAY** truncate its output to as few as six decimal digits or equivalent.

If the nonce used to generate the authenticator output is based on a real-time clock, the nonce **SHALL** be changed at least once every two minutes.

3.1.4.2. Single-Factor OTP Verifiers

Single-factor OTP verifiers effectively duplicate the process of generating the OTP used by the authenticator. As such, the symmetric keys used by authenticators are also present in the verifier and **SHALL** be strongly protected against unauthorized disclosure by access controls that limit access to the keys to only those software components that require access.

When binding a single-factor OTP authenticator to a subscriber account, the verifier or associated CSP **SHALL** use approved cryptography for key establishment to generate and exchange keys or to obtain the secrets required to duplicate the authenticator output.

The verifier **SHALL** use approved encryption and an authenticated protected channel when collecting the OTP. Verifiers **SHALL** accept a given OTP only once while it is valid to provide replay resistance as described in Sec. 3.2.7. If a claimant's authentication is denied due to the duplicate use of an OTP, verifiers **MAY** warn the claimant if an attacker has been able to authenticate in advance. Verifiers **MAY** also warn a subscriber in an existing session of the attempted duplicate use of an OTP.

The verifier **SHOULD** implement or, if the authenticator output is less than 64 bits in length, **SHALL** implement a rate-limiting mechanism that effectively limits the number of failed authentication attempts that can be made on the subscriber account, as described in Sec. 3.2.2.

3.1.5. Multi-Factor OTPs

A multi-factor OTP generates one-time passwords for authentication following the input of an activation factor. This includes hardware devices and software-based OTP generators that are installed on mobile phones and similar devices. The second authentication factor may be provided through an integral entry pad, an integral biometric (e.g., fingerprint) reader, or a direct computer interface (e.g., USB port). The OTP is displayed on the authenticator and manually input for transmission to the verifier. The multi-factor OTP authenticator is "something you have" activated by either "something you know" or "something you are."

OTP authentication is not phishing-resistant. [FIPS140] validation of OTP authenticators and verifiers is not required.

3.1.5.1. Multi-Factor OTP Authenticators

Multi-factor OTP authenticators operate similarly to single-factor OTP authenticators (see Sec. 3.1.4.1), except they require the presentation and verification of either a password or a biometric characteristic to obtain the OTP from the authenticator. Each use of the authenticator **SHALL** require the input of the activation factor.

In addition to activation information, multi-factor OTP authenticators and verifiers contain two persistent values: 1) a symmetric key that persists for the authenticator's lifetime and 2) a nonce that is either changed each time the authenticator is used or based on a real-time clock.

The secret key and its algorithm **SHALL** provide at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication). The nonce **SHALL** be of sufficient length to ensure that it is unique for each operation of the authenticator over its lifetime. If a subscriber needs to change the device on which a software-based OTP authenticator resides, they **SHOULD** bind the authenticator application on the new device to their subscriber account, as described in Sec. 4.1.2, and invalidate the authenticator application that will no longer be used.

The authenticator output is obtained using an approved block cipher or hash function to securely combine the key and nonce. In coordination with the verifier, the authenticator **MAY** truncate its output to as few as six decimal digits or equivalent.

If the nonce used to generate the authenticator output is based on a real-time clock, the nonce **SHALL** be changed at least once every two minutes.

Authenticator activation secrets **SHALL** meet the requirements of Sec. 3.2.10. A biometric activation factor **SHALL** meet the requirements of Sec. 3.2.3, including limits on the number of consecutive authentication failures. The unencrypted key and activation secret or biometric sample and any biometric data derived from the biometric sample (e.g., a probe produced through signal processing) **SHALL** be zeroized (erased) immediately after an OTP has been generated.

3.1.5.2. Multi-Factor OTP Verifiers

Multi-factor OTP verifiers effectively duplicate the process of generating the OTP used by the authenticator without requiring a second authentication factor. As such, the symmetric keys used by authenticators **SHALL** be strongly protected against unauthorized disclosure by access controls that limit access to the keys to only those software components that require access.

When binding a multi-factor OTP authenticator to a subscriber account, the verifier or associated CSP **SHALL** use approved cryptography for key establishment to generate and exchange keys or to obtain the secrets required to duplicate the authenticator output.

The verifier **SHALL** use approved encryption and an authenticated protected channel when collecting the OTP. Verifiers **SHALL** accept a given OTP only once while it is valid to provide replay resistance as described in Sec. 3.2.7. If a claimant's authentication is denied due to the duplicate use of an OTP, verifiers **MAY** warn the claimant if an attacker has been able to authenticate in advance. Verifiers **MAY** also warn a subscriber in an existing session of the attempted duplicate use of an OTP.

Time-based OTPs [TOTP] **SHALL** have a defined lifetime that is determined by the expected clock drift in either direction of the authenticator over its lifetime plus an allowance for network delay and user entry of the OTP.

The verifier **SHALL** implement a rate-limiting mechanism that effectively limits the number of consecutive failed authentication attempts that can be made on the subscriber account, as required by Sec. 3.2.10.

3.1.6. Single-Factor Cryptographic Authentication

Single-factor cryptographic authentication is accomplished by proving the possession and control of a cryptographic key via an authentication protocol. Depending on the strength of authentication required, the private or symmetric key may be stored in a manner that is accessible to the endpoint being authenticated or in a separate, directly connected processor or device from which the key cannot be exported. The authenticator output is highly dependent on the specific cryptographic protocol used but is generally some type of signed message. A single-factor cryptographic authenticator is "something you have."

Cryptographic authentication is phishing-resistant if it meets the additional requirements in Sec. 3.2.5.

3.1.6.1. Single-Factor Cryptographic Authenticators

Single-factor cryptographic authenticators encapsulate one or more private or symmetric keys. The key **SHOULD** be stored in appropriate storage available to the authenticator (e.g., keychain storage), or if the key is to be non-exportable, it **SHALL** be stored in an isolated execution environment protected by hardware or in a separate processor with a controlled interface to the central processing unit of the user endpoint. If they are accessible to the endpoint being authenticated, the private or symmetric keys **SHALL** be strongly protected against unauthorized disclosure by using access controls that limit access to the key to only those software components that require access.

External (i.e., non-embedded) cryptographic authenticators **SHALL** meet the requirements for connected authenticators in Sec. 3.2.11.

As required by Sec. 2.3.2, single-factor cryptographic authenticators that are being used at AAL3 must meet the authentication intent requirements of Sec. 3.2.8.

3.1.6.2. Single-Factor Cryptographic Verifiers

Single-factor cryptographic verifiers generate a challenge nonce, send it to the corresponding authenticator, and use the authenticator output to verify possession of the authenticator. The authenticator output is highly dependent on the specific cryptographic authenticator and protocol used but is generally some type of signed message.

The verifier has either a symmetric or an asymmetric public cryptographic key that corresponds to each authenticator. While both types of keys **SHALL** be protected against modification, symmetric keys **SHALL** additionally be protected against unauthorized disclosure by access controls that limit access to the key to only those software components that require access.

The secret or symmetric key and its algorithm **SHALL** provide at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication). The challenge nonce **SHALL** be at least 64 bits in length and **SHALL** either be unique over the authenticator's lifetime or statistically unique (i.e., generated using an approved random bit generator, as described in Sec. 3.2.12). The verification operation **SHALL** use approved cryptography.

3.1.7. Multi-Factor Cryptographic Authentication

Multi-factor cryptographic authentication uses an authentication protocol to prove possession and control of a cryptographic key that requires activation through a second authentication factor. Depending on the strength of authentication needed, the private or symmetric key may be stored in a manner accessible to the endpoint being authenticated or in a separate, directly connected processor or device from which the key cannot be exported. The authenticator output is highly dependent on the specific cryptographic protocol used but is generally some type of signed message. A multi-factor cryptographic authenticator is "something you have" and is activated by an activation factor representing either "something you know" or "something you are."

Cryptographic authentication is phishing-resistant if it meets the additional requirements in Sec. 3.2.5.

3.1.7.1. Multi-Factor Cryptographic Authenticators

Multi-factor cryptographic authenticators encapsulate one or more private or symmetric keys that **SHALL** only be accessible through the presentation and verification of an activation factor (i.e., a password or a biometric characteristic). The key **SHOULD** be stored in appropriate storage available to the authenticator (e.g., keychain storage), or if the key is to be non-exportable, it **SHALL** be stored in an isolated execution environment protected by hardware or in a separate processor with a controlled interface to the central processing unit of the user endpoint. If accessible to the endpoint being authenticated, the key **SHALL** be strongly protected against unauthorized disclosure by using access controls that limit access to the key to only those software components that require access.

Some cryptographic authenticators, referred to as "syncable authenticators," can manage their private keys using a *sync fabric* (cloud provider). Additional requirements for using syncable authenticators are in Appendix B.

External (non-embedded) cryptographic authenticators **SHALL** meet the requirements for connected authenticators in Sec. 3.2.11.

Each authentication operation that uses the authenticator **SHALL** require the activation factor to be input.

Authenticator activation secrets **SHALL** meet the requirements of Sec. 3.2.10. A biometric activation factor **SHALL** meet the requirements of Sec. 3.2.3, including limits on the number of consecutive authentication failures.

The activation secret or biometric sample and any biometric data derived from the biometric sample (e.g., a probe produced through signal processing) **SHALL** be zeroized (erased) after an authentication transaction.

3.1.7.2. Multi-Factor Cryptographic Verifiers

The requirements for a multi-factor cryptographic verifier are identical to those for a single-factor cryptographic verifier, as described in Sec. 3.1.6.2. Verification of the output from a multi-factor cryptographic authenticator proves that the activation factor was used.

3.1.7.3. Usage With Subscriber-Controlled Wallets

A special-case usage of multi-factor cryptographic authentication is with subscriber-controlled wallets, described in Sec. 5 of [SP800-63C]. After the claimant first unlocks the wallet using an activation factor, the authentication process uses a federation protocol, as detailed in [SP800-63C]. The assertion contents and presentation requirements of the federation protocol provide the security characteristics required of cryptographic authenticators. As such, subscriber-controlled wallets can be considered multi-factor authenticators through the activation factor and the presentation and validation of an assertion generated by the wallet.

Access to the private key **SHALL** require an activation factor. Authenticator activation secrets **SHALL** meet the requirements of Sec. 3.2.10. Biometric activation factors **SHALL** meet the requirements of Sec. 3.2.3, including limits on the number of consecutive authentication failures. The password or biometric sample used for activation and any biometric data derived from the biometric sample **SHALL** be zeroized (erased) immediately after an authentication transaction.

Authentication processes using subscriber-controlled wallets **SHALL** be used with a federation process as detailed in Sec. 5 of [SP800-63C]. Signed audience-restricted assertions generated by subscriber-controlled wallets are considered phishing-resistant because they prevent an assertion presented to an impostor RP from being used by the legitimate one. Assertions that lack a valid signature from the wallet or an audience restriction **SHALL NOT** be considered phishing-resistant.

3.1.7.4. Syncable Authenticators

Some multifactor cryptographic authenticators allow the subscriber to copy (clone) the authentication secret to additional devices, usually via a sync fabric. This eases the burden for subscribers who want to use additional devices to authenticate. Specific requirements for syncable authenticators and the sync fabric are given in Appendix B.

3.2. General Authenticator Requirements

The following requirements apply to all types of authentication.

3.2.1. Physical Authenticators

CSPs **SHALL** provide subscriber instructions for appropriately protecting the authenticator against theft or loss. The CSP **SHALL** provide a mechanism to invalidate[1] the authenticator immediately upon notification from a subscriber that the authenticator's loss, theft, or compromise is suspected.

Possession and control of a physical authenticator are based on proof of possession of a secret associated with the authenticator. When an embedded secret (typically a certificate and associated private key) is in the endpoint, its "device identity" can be considered a physical authenticator. However, this requires a secure authentication protocol that is appropriate for the AAL being authenticated. Browser cookies do not satisfy this requirement except at AAL1 or as short-term secrets for session maintenance (not authentication) as described in Sec. 5.1.1.

3.2.2. Rate Limiting (Throttling)

When required by the authenticator type descriptions in Sec. 3.1, the verifier **SHALL** implement controls to protect against online guessing attacks. Unless otherwise specified in the description of a given authenticator, the verifier **SHALL** limit consecutive failed authentication attempts using one or more specific authenticators on a single subscriber account to no more than 100.

[1] Invalidation can take several forms, including revocation of a PKI-based authenticator and removal from the subscriber account.

> The limit of 100 attempts is an upper bound; agencies **MAY** impose lower limits. The limit of 100 was chosen to balance the likelihood of a correct guess (e.g., 100 attempts against a six-digit decimal OTP authenticator output) versus the potential need for account recovery when the limit is exceeded.

Additional techniques **MAY** be used to reduce the likelihood that an attacker will lock the legitimate claimant out due to rate limiting. These include:

- Requiring the claimant to complete a bot-detection and mitigation challenge before attempting authentication
- Requiring the claimant to wait after a failed attempt for a period of time that increases as the subscriber account approaches its maximum allowance for consecutive failed attempts (e.g., 30 seconds up to an hour)
- Accepting only authentication requests from IP addresses from which the subscriber has been successfully authenticated before
- Leveraging other risk-based or adaptive authentication techniques to identify user behavior that falls within or outside typical norms (e.g., the use of the claimant's IP address, geolocation, timing of request patterns, or browser metadata)

When the subscriber successfully authenticates, the verifier **SHOULD** disregard any previous failed attempts for the authenticators used in the successful authentication.

Following successful authentication at a given AAL, the verifier **SHOULD** reset the retry count of an authenticator that has been locked out due to excessive retries. If this is provided, the maximum AAL of the authenticator being reset **SHALL** not exceed the AAL of the session from which it is being reset. If the subscriber cannot authenticate at the required AAL, the account recovery procedures in Sec. 4.2 **SHALL** be used.

3.2.3. Use of Biometrics

The use of biometrics (i.e., something you are) in authentication includes both the measurement of physical characteristics (e.g., fingerprint, iris, facial characteristics) and behavioral characteristics (e.g., typing cadence). Both classes are considered biometric modalities, although modalities may differ in the extent to which they establish authentication intent as described in Sec. 3.2.8.

For a variety of reasons, this document supports only a limited use of biometrics for authentication. These reasons include:

- The biometric false match rate (FMR) does not provide sufficient confidence in the subscriber's authentication by itself. In addition, FMR does not account for spoofing attacks.

- Biometric comparison is probabilistic, whereas the other authentication factors are deterministic.
- Biometric template protection schemes provide a method for revoking biometric characteristics comparable to other authentication factors (e.g., PKI certificates and passwords). However, the availability of such solutions is limited.
- Biometric characteristics do not constitute secrets. They can often be obtained online or, in the case of a facial image, by taking a picture of someone with or without their knowledge. Latent fingerprints can be lifted from objects someone touches, and iris patterns can be captured with high-resolution images. While presentation attack detection (PAD) technologies can mitigate the risk of these types of attacks, additional trust in the sensor or biometric processing is required to ensure that PAD is operating in accordance with the needs of the CSP and the subscriber.

Therefore, the limited use of biometrics for authentication is supported with the following requirements and guidelines.

Biometrics **SHALL** be used only as part of multi-factor authentication with a physical authenticator (i.e., “something you have”). The biometric characteristic **SHALL** be presented and compared for each authentication operation. An alternative non-biometric authentication option **SHALL** always be provided to the subscriber. Biometric data **SHALL** be treated and secured as sensitive PII.

The biometric system **SHALL** operate with an FMR [ISO/IEC2382-37] of one in 10000 or better. This FMR **SHALL** be achieved under the conditions of a conformant attack (i.e., zero-effort impostor attempt) as defined in [ISO/IEC30107-1]. The biometric system **SHOULD** demonstrate a false non-match rate (FNMR) of less than 5 %. Biometric performance **SHALL** be tested in accordance with [ISO/IEC19795-1].

Biometric authentication technologies **SHALL** provide similar performance for subscribers of different demographic types (e.g., racial background, gender, ethnicity).

The biometric system **SHOULD** implement PAD. Testing the biometric system for deployment **SHOULD** demonstrate an impostor attack presentation accept rate (IAPAR) of less than 0.15. Presentation attack resistance **SHALL** be tested in accordance with Clause 13 of [ISO/IEC30107-3]. The PAD decision **MAY** be made either locally on the claimant’s device or by a central verifier.

The biometric system **SHALL** allow no more than five consecutive failed authentication attempts or 10 consecutive failed attempts if PAD is implemented and meets the above requirements. Once that limit has been reached, the biometric authenticator **SHALL** impose a delay of at least 30 seconds before each subsequent attempt, with an overall limit of no more than 50 consecutive failed authentication attempts or 100 if PAD is implemented due to the mitigation of presentation attacks. Once the overall limit is reached, the biometric system **SHALL** disable biometric user authentication and offer another factor (e.g., a different biometric modality or an activation secret if it is not a required factor) if such an alternative method is already available. These limits are upper bounds, and agencies **MAY** make risk-based decisions to impose lower limits.

The verifier **SHOULD** determine the performance and integrity of the sensor and its associated endpoint. Acceptable methods for making this determination include but are not limited to:

- Use of a known sensor, as determined by sensor authentication
- First- or third-party testing against biometric performance standards
- Runtime interrogation of signed metadata (e.g., attestation), as described in Sec. 3.2.4

Biometric comparison can be performed locally on a device being used by the claimant or at a central verifier. Since the potential for attacks on a larger scale is greater at central verifiers, comparison **SHOULD** be performed locally.

The presentation of a biometric factor for authenticator activation **SHALL** be a separate operation from unlocking the host device (e.g., smartphone). However, the same activation factor used to unlock the host device **MAY** be used in the authentication operation. Agencies **MAY** lower this requirement for authenticators that are managed by or on behalf of the CSP (e.g., via mobile device management) and constrained to have short agency-determined inactivity timeouts and biometric systems that meet the above requirements.

If the comparison is performed centrally:

- The sensor or endpoint **SHALL** be authenticated before capturing the biometric sample from the claimant. The verifier **MAY** limit the use of the centrally stored biometric template to particular sensors or sensor classes (e.g., sensor manufacturers or models).
- Appropriate controls (e.g., encryption and access controls) for sensitive PII **SHALL** be implemented.
- An authenticated protected channel between the sensor (or an endpoint containing a sensor that resists sensor replacement) and the verifier **SHALL** be established. All transmission of biometric information **SHALL** be conducted over that authenticated protected channel.

Biometric samples collected in the authentication process **MAY** be used to train comparison algorithms (e.g., updating templates to address changes in subscriber characteristics) or — with subscriber consent — for other research purposes. Biometric samples and any biometric data derived from the biometric sample **SHALL** be zeroized (erased) immediately after any training or research data has been derived.

3.2.4. Attestation

The CSP needs to have a reliable basis for evaluating the characteristics of the authenticator, such as the inclusion of a signed attestation. An attestation is information conveyed to the CSP, generally when an authenticator is bound, regarding a connected authenticator or the endpoint involved in an authentication operation. Information conveyed by attestation **MAY** include but is not limited to:

- The provenance (e.g., manufacturer or supplier certification), health, and integrity of the authenticator and endpoint
- Security features of the authenticator
- Security and performance characteristics of biometric sensors
- Sensor modality

Attestations **SHALL** be signed using a digital signature that provides at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication).

Verifiers in federal enterprise systems[2] **SHOULD** use attestation features to verify the capabilities and source of authenticators. In other applications, attestation information **MAY** be used as part of a verifier's risk-based authentication decisions.

3.2.5. Phishing (Verifier Impersonation) Resistance

Phishing attacks, previously referred to in SP 800-63B as "verifier impersonation," are attempts by fraudulent verifiers and RPs to fool an unwary claimant into presenting an authenticator to an impostor. In some prior versions of SP 800-63, protocols resistant to phishing attacks were also referred to as "strongly MitM-resistant."

The term *phishing* is widely used to describe a variety of similar attacks. In this document, phishing resistance is the ability of the authentication protocol to prevent the disclosure of authentication secrets and valid authenticator outputs to an impostor verifier without relying on the vigilance of the claimant. How the claimant is directed to the impostor verifier is not relevant. For example, regardless of whether the claimant was directed there via search engine optimization or prompted by email, it is considered to be a phishing attack.

Approved cryptographic algorithms **SHALL** be used to establish phishing resistance where required. Keys used for this purpose **SHALL** provide at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication).

[2] Federal enterprise systems include those considered in scope for PIV guidance, such as government contractors, government employees, and mission partners. It does not include government-to-consumer or public-facing use cases.

Phishing resistance requires single- or multi-factor cryptographic authentication. Authenticators that involve the manual entry of an authenticator output (e.g., out-of-band and OTP authenticators) are not phishing-resistant because the manual entry does not bind the authenticator output to the specific session being authenticated. For example, an impostor verifier could relay an authenticator output to the verifier and successfully authenticate.

Two methods of phishing resistance are recognized: channel binding and verifier name binding. Channel binding is considered more secure than verifier name binding because it is not vulnerable to the misissuance or misappropriation of verifier certificates, but both methods satisfy the requirements for phishing resistance.

3.2.5.1. Channel Binding

An authentication protocol with channel binding **SHALL** be used to establish an authenticated protected channel with the verifier. The protocol **SHALL** then strongly and irreversibly bind a channel identifier negotiated in establishing the authenticated protected channel to the authenticator output (e.g., by signing the two values together using a private key controlled by the claimant for which the public key is known to the verifier). The verifier **SHALL** validate the signature or other information used to prove phishing resistance. This prevents an impostor verifier — even one that has obtained a certificate representing the actual verifier — from successfully relaying that authentication on a different authenticated protected channel.

An example of a phishing-resistant authentication protocol that uses channel binding is client-authenticated TLS [TLS] because the client signs the authenticator output along with earlier messages from the protocol that are unique to the particular TLS connection being negotiated.

3.2.5.2. Verifier Name Binding

An authentication protocol with verifier name binding **SHALL** be used to establish an authenticated protected channel with the verifier. The protocol **SHALL** then generate an authenticator output that is cryptographically bound to a verifier identifier that is authenticated as part of the protocol. In the case of domain name system (DNS) identifiers, the verifier identifier **SHALL** be either the authenticated hostname of the verifier or a parent domain that is at least one level below the public suffix [PSL] associated with that hostname. The binding **MAY** be established by choosing an associated authenticator secret, deriving an authenticator secret using the verifier identifier, cryptographically signing the authenticator output with the verifier identifier, or using similar cryptographically secure means.

W3C WebAuthn [WebAuthn], which is used by authenticators that implement the FIDO2 specifications [FIDO2], is an example of a standard that provides phishing resistance through verifier name binding.

3.2.6. Verifier-CSP Communications

If the verifier and CSP are separate entities (as shown by the dotted line in Fig. 3 of [SP800-63]), communications between the verifier and CSP **SHALL** occur through a mutually authenticated secure channel (e.g., a client-authenticated TLS connection) using approved cryptography.

3.2.7. Replay Resistance

An authentication process resists replay attacks if it is impractical to achieve a successful authentication by recording and replaying a previous authentication message. Replay resistance is in addition to the replay-resistant nature of authenticated protected channel protocols since the output could be stolen before entry into the protected channel. Protocols that use nonces or challenges to prove the “freshness” of the transaction are resistant to replay attacks since the verifier will easily detect when old protocol messages are replayed because they will not contain the appropriate nonces or timeliness data.

Examples of replay-resistant authenticators include OTP authenticators, cryptographic authenticators, and look-up secrets.

In contrast, passwords are not considered replay-resistant because the authenticator output — the secret itself — is provided for each authentication.

3.2.8. Authentication Intent

An authentication process demonstrates intent if it requires the claimant to respond explicitly to each authentication or reauthentication request. The goal of authentication intent is to make it more difficult for authenticators (e.g., multi-factor cryptographic authenticators) to be used without the claimant’s knowledge, such as by malware on the endpoint. The authenticator itself **SHALL** establish authentication intent, although multi-factor cryptographic authenticators **MAY** establish intent by reentry of the activation factor for the authenticator.

Authentication intent **MAY** be established in several ways. Authentication processes that require the claimant’s intervention can be used to prove intent (e.g., a claimant entering an authenticator output from an OTP authenticator). Cryptographic authenticators that require user action for each authentication or reauthentication operation can also be used to establish intent (e.g., by pushing a button or reinsertion).

The presentation of biometric characteristics does not always establish authentication intent. For example, using a front-facing camera on a mobile phone to capture a face biometric does not constitute intent, as it can be reasonably expected to capture a face image while the device is used for other non-authentication purposes. In these scenarios, an explicit mechanism (e.g., tapping a software or physical button) **SHALL** be provided to establish authentication intent.

3.2.9. Restricted Authenticators

As threats evolve, authenticators' ability to resist attacks typically degrades. Conversely, the performance of some authenticators may improve, such as when changes to their underlying standards increase their ability to resist particular attacks.

To account for these changes in authenticator performance, NIST places additional restrictions on authenticator types or specific classes or instantiations of an authenticator type. Although they represent a less secure approach to multi-factor authentication, *restricted authenticators* remain necessary for some government-to-public applications.

The acceptance of a restricted authenticator requires the implementing organization to assess, understand, and accept the risks associated with that authenticator and acknowledge that risks will likely increase over time. It is the RP's responsibility to determine the level of acceptable risk for their systems and associated data, to define any methods for mitigating excessive risks, and to communicate those determinations to the verifier. If the RP determines that the risk to any party is unacceptable, the restricted authenticator **SHALL NOT** be used, and an alternative authenticator type **SHALL** be used.

Furthermore, the risk of an authentication error is typically borne by multiple parties, including the implementing organization, organizations that rely on the authentication decision, and the subscriber. Because the subscriber may be exposed to additional risks when an organization accepts a restricted authenticator and the subscriber may have a limited understanding of and ability to control that risk, the CSP **SHALL** do all of the following:

1. Offer subscribers at least one alternative authenticator that is not restricted and can be used to authenticate at the required AAL
2. Provide subscribers with meaningful notice regarding the restricted authenticator's security risks and the availability of unrestricted alternatives
3. Address any additional risks to subscribers and RPs in its risk assessment
4. Develop a migration plan for the possibility that the restricted authenticator is no longer acceptable in the future and include this migration plan in its Digital Identity Acceptance Statement (see Sec. 3.4.4 of [SP800-63])

3.2.10. Activation Secrets

A password used locally as an activation factor for a multi-factor authenticator is referred to as an *activation secret*. An activation secret is used to obtain access to a stored authentication key. In all cases, the activation secret **SHALL** remain within the authenticator and its associated user endpoint.

Authenticators that use activation secrets **SHALL** require the secrets to be at least four characters in length and **SHOULD** require the secrets to be at least six characters in length. Activation secrets **MAY** be entirely numeric (i.e., a PIN). If alphanumeric values are permitted, all printing ASCII [RFC20] characters and the space character, **SHOULD** be allowed. Unicode [ISO/ISC 10646] characters **SHOULD** also be permitted in alphanumeric secrets. The authenticator or its management tools **SHOULD** implement a blocklist to discourage users from selecting commonly used activation secrets (e.g., 123456).

The authenticator or verifier **SHALL** implement a retry-limiting mechanism that limits the number of consecutive failed activation attempts using the authenticator to no more than 10. If an incorrect activation secret entry causes the authenticator to provide an invalid output to the central verifier, the verifier **MAY** implement this retry-limiting mechanism. Otherwise, retry limiting **SHALL** be implemented in the authenticator. Once the limit of attempts is reached, the authenticator **SHALL** be disabled, and a different authenticator **SHALL** be required for authentication.

For authenticators that are usable at AAL3, verification of activation secrets **SHALL** be performed in a hardware-protected environment (e.g., a secure element, TPM, or TEE). At AAL2, if a hardware-protected environment is not used, the authenticator **SHALL** use the activation secret to derive a key used to decrypt the authentication key.

Submitting the activation factor **SHALL** be a separate operation from unlocking the host device (e.g., smartphone). However, the same activation factor used to unlock the host device **MAY** be used in the authentication operation. Agencies **MAY** lower this requirement for authenticators and that are managed by or on behalf of the CSP (e.g., via mobile device management) that are constrained to have short agency-determined inactivity timeouts and device activation factors that meet the corresponding requirements in this section.

3.2.11. Connected Authenticators

Cryptographic authenticators require a trustworthy connection between the authenticator and the endpoint being authenticated that provides resistance to eavesdropping, injection, and relay attacks. This connection **SHALL** be made using a wired connection (e.g., USB or direct connection with a smartcard), a wireless technology, or a hybrid of those technologies, including network connections.

Approved cryptography **SHALL** be used for all cases in which cryptographic operations are required. All communication of authentication data between authenticators and endpoints **SHALL** occur directly between those devices or through an authenticated protected channel between the authenticator and endpoint.

3.2.11.1. Wired Connections

Wired connections, including those with embedded authenticators, **MAY** be assumed to be trustworthy because their attack surface is minimal. Claimants **SHOULD** be advised to use trusted hardware (e.g., cables, adapters, etc.) to ensure that they have not been compromised.

3.2.11.2. Wireless and Hybrid Connections

Wireless and network-based authenticator connections are potentially vulnerable to threats, including eavesdropping, injection, and relay attacks. The potential for such attacks on wireless connections depends on the technology's effective range. To minimize the attack surface for threats to the authenticator-endpoint connection, the authentication process **SHALL** require physical proximity between the authenticator and endpoint by establishing a wireless connection with a range of no more than 200 meters.

Wireless and hybrid connections **SHALL** establish a key for encrypted communication between the authenticator and endpoint in one of the following ways:

1. Through a temporary wired connection between the devices.
2. Through an association process (similar to a pairing process but not requiring a persistent relationship between devices) to establish a key for encrypted communication between the authenticator and endpoint. The association process **SHALL** employ a pairing code[3] or other shared secret between the devices. Either the authenticator or endpoint **SHALL** have a pairing code that **MAY** be printed on the device. The pairing code **SHALL** be at least six decimal digits (or equivalent) in length. It **SHALL** be conveyed between the devices by manual entry or using a QR code or similar representation that is optically communicated.

When using a wireless technology with an effective range of less than 1 meter (e.g., NFC), any activation secret transmitted from the endpoint to the authenticator **SHALL** be encrypted using a key established between the devices. An authenticated connection **SHOULD** be used. A pairing code **SHALL** be used if the authenticator is configured to require authenticated pairing.

> Encrypting only the activation secret and not the entire authentication transaction may expose sensitive information, such as the identity of the RP, although this would require the attacker to be very close to the subscriber. Special care should be taken with authenticators that contain PII and that do not require authenticated pairing. Encryption **SHOULD** be used to protect that information against "skimming" and eavesdropping attacks.

Wireless technologies with an effective range of 1 meter or more (e.g., Bluetooth LE) and network connections **SHALL** use an authenticated encrypted connection between the authenticator and endpoint. The entire authentication transaction **SHALL** be encrypted. Examples of this include the pairing code used with the virtual contact interface specified in [SP800-73] and the hybrid transport specified by the [CTAP2.2] protocol.

[3] As used in this section, the term *pairing code* does not imply that a persistent pairing process (e.g., Bluetooth) is necessarily used.

The key established by the association process may be either temporary (i.e., valid for a limited number of transactions or time-limited) or persistent. A mechanism for endpoints to remove persistent keys **SHALL** be provided.

3.2.12. Random Values

Random values are extensively used in authentication processes, such as nonces and authentication secrets. Unless otherwise specified, random values that reference this section **SHALL** be generated by an approved random bit generator [RBG][4] that provides at least the minimum security strength specified in the latest revision of [SP800-131A] (112 bits as of the date of this publication).

3.2.13. Exportability

Exportability is the ability of an authenticator to share its authentication secret (either a private or symmetric key) with another endpoint or authenticator. Generally, endpoints with access to the authentication secret are considered exportable since software (perhaps malware) on the endpoint could access and leak the authentication secret. Non-exportable authenticators are considered more secure, and accordingly, a non-exportable cryptographic authenticator is required at AAL3. Syncable authenticators are inherently exportable (see Appendix B).

To be considered non-exportable, an authenticator **SHALL** either be a separate piece of hardware or an embedded processor or execution environment (e.g., secure element, TEE, or trusted platform module). These hardware authenticators and embedded processors are separate from a host processor, such as the CPU on a laptop or mobile device. A non-exportable authenticator **SHALL** be designed to prohibit the export of the authentication secret to the host processor and **SHALL NOT** be capable of being reprogrammed by the host processor to allow the secret to be extracted. The authenticator is subject to applicable [FIPS140] requirements of the AAL at which the authenticator is being used, including applicable tamper resistance requirements.

[4] Detailed information on generating random values may be found in the NIST SP 800-90 document suite comprising [SP800-90A], [SP800-90B], and [SP800-90C].

4. Authenticator Event Management

This section is normative.

Events can occur over the lifetime of a subscriber's authenticator and affect its use. These events include binding, maintenance, loss, theft, compromise, unauthorized duplication, expiration, and revocation. This section describes the actions to be taken in response to those events.

4.1. Authenticator Binding

Authenticator binding refers to establishing an association between a specific authenticator and a subscriber account to enable the authenticator to authenticate for that subscriber account, possibly in conjunction with other authenticators.

Authenticators **SHALL** be bound to subscriber accounts by either:

- Being issued by the CSP as part of enrollment or
- Using a subscriber-provided authenticator that is acceptable to the CSP.

The SP 800-63 suite of guidelines refers to the *binding* rather than the issuance of authenticators to accommodate both options.

Throughout the lifetime of a digital identity, CSPs **SHALL** maintain a record of all authenticators that are or have ever been bound to each subscriber account. The CSP **SHALL** determine the characteristics of the authenticator being bound (e.g., single-factor versus multi-factor, phishing-resistant or not) so that verifiers can assess compliance with the requirements at each AAL. This determination **MAY** be based on strong evidence (e.g., authenticator attestation), direct information from having issued the authenticator, or typical characteristics of authenticator implementations (e.g., whether a user verification bit is set).

The CSP **SHALL** also maintain other state information required to meet the authenticator verification requirements. For example, the throttling of authentication attempts described in Sec. 3.2.2 requires the CSP or verifier to maintain state information on recent failed authentication attempts, except for activation factors verified at the authenticator.

The record created by the CSP **SHALL** contain the date and time of significant authenticator life cycle events (e.g., binding to the subscriber account, renewal, update, expiration). The record **SHOULD** include information about the source of the binding (e.g., IP address, device identifier) of any device associated with the event.

As part of the binding process, the CSP **MAY** require additional information about the new authenticator or its associated endpoint to determine whether it is suitable for the requested AAL.

4.1.1. Binding at Enrollment

Binding at the time of enrollment is considered to be part of the enrollment process and is discussed in [SP800-63A].

4.1.2. Post-Enrollment Binding

4.1.2.1. Binding an Additional Authenticator

To minimize the need for account recovery, CSPs and verifiers **SHOULD** encourage subscribers to maintain at least two separate means of authentication. For example, a subscriber who usually uses an OTP authenticator as a physical authenticator **MAY** also be issued look-up secret authenticators or register a device for out-of-band authentication to be used if the physical authenticator is lost, stolen, or damaged. See Sec. 4.2 for more information on replacing passwords.

Accordingly, CSPs **SHOULD** permit the binding of multiple authenticators to a subscriber account. When any new authenticator is bound to a subscriber account, the CSP **SHALL** ensure that the process requires authentication at either the maximum AAL currently available in the subscriber account or the maximum AAL at which the new authenticator will be used, whichever is lower. For example, binding an authenticator that is suitable for use at AAL2 requires authentication at AAL2 unless the subscriber account currently has only AAL1 authentication capability. When an authenticator is added, the CSP **SHALL** notify the subscriber via a mechanism independent of the transaction binding the new authenticator, as described in Sec. 4.6.

4.1.2.2. External Authenticator Binding

External authenticator binding refers to binding an authenticator to a subscriber account when it is not connected to or embedded in the authenticated endpoint. This process is typically used when adding authenticators that are embedded in a new endpoint or when connectivity limitations prevent the newly bound authenticator from being connected to an authenticated endpoint.

The binding process **SHALL** proceed in one of the following ways:

- An endpoint that has authenticated to the CSP requests a binding code from the CSP. The binding code is input into the endpoint associated with the new authenticator and sent to the CSP.
- The endpoint associated with the new authenticator obtains a binding code from the CSP. The binding code is input to an authenticated endpoint and sent to the CSP.

In addition to the requirements in Sec. 4.1.2.1 and Sec. 4.2, the following requirements **SHALL** apply when binding an external authenticator:

- An authenticated protected channel **SHALL** be established by the endpoint associated with the new authenticator and the CSP.
- The subscriber **MAY** be prompted to enter an identifier by which the CSP knows them on the endpoint associated with the new authenticator.
- The CSP **SHALL** generate a *binding code* using an approved random bit generator as described in Sec. 3.2.12 and send it to either the new authenticator endpoint or the authenticated endpoint approving the binding. The binding code **SHALL** be at least 40 bits in length if used with an identifier entered in the previous step. Otherwise, a binding code of at least 112 bits in length **SHALL** be required.
- The subscriber **SHALL** transfer the binding code to the other endpoint. This transfer **SHALL** either be manual or via a local out-of-band method (e.g., QR code). The binding code **SHALL NOT** be communicated over any insecure channel (e.g., email).
- The binding code **SHALL** be usable only once and **SHALL** be valid for a maximum of 10 minutes.
- Following the binding of the new authenticator (or issuance of a certificate, in the case of PKI-based authenticators), the CSP **SHOULD** encourage the subscriber to authenticate with the new authenticator to confirm that the process has been completed successfully.
- The CSP **SHALL** provide clear instructions on what the subscriber should do in the event of an authenticator binding mishap (e.g., making a button available to be pressed or a contact address to be used to allow a misbound authenticator to be quickly invalidated), as appropriate. This **MAY** be provided in the authenticated session in addition to the binding notification described in Sec. 4.6.

The binding of an external authenticator may introduce risks due to the potential for the subscriber to be tricked into using a binding code by an attacker or supplying a binding code to an attacker. In some cases, representations (e.g., QR codes) obtained from a trusted source (e.g., an authenticated session, especially when that authentication is phishing-resistant) are considered to be more robust against such attacks because they typically contain the URL of the CSP in addition to the binding code. As a result, there is less potential for the subscriber to be fooled into entering a binding code at a phishing site.

4.1.3. Binding to a Subscriber-Provided Authenticator

A subscriber may already possess authenticators that are suitable for authentication at a particular AAL. For example, they may have a multi-factor authenticator from a social network provider, considered AAL2 without identity proofing, and would like to use that authenticator at an RP that requires IAL2. This would necessitate identity proofing at IAL2, perhaps by a different CSP, and binding authenticators at enrollment with that CSP.

CSPs **SHOULD**, where practical, accommodate subscriber-provided authenticators to relieve the burden on the subscriber of managing many authenticators. The binding of these authenticators **SHALL** be done as described in Sec. 4.1.2. If the authenticator strength is not self-evident (e.g., between single-factor and multi-factor authenticators of a given type), the CSP **SHALL** assume that the weaker authenticator has been used unless it can establish that the stronger authenticator is being used (e.g., by verification with the issuer or manufacturer of the authenticator).

4.1.4. Renewal

The subscriber **SHOULD** bind a new or updated authenticator before an existing authenticator's expiration. The process for this **SHOULD** conform closely to the binding process for an additional authenticator described in Sec. 4.1.2. The CSP **MAY** periodically take other actions (e.g., confirming contact addresses), either as a part of the renewal process or separately. Following the successful use of the replacement authenticator, the CSP **SHOULD** invalidate the expiring authenticator.

4.2. Account Recovery

Account recovery is when a subscriber recovers from losing control of the authenticators necessary to authenticate at a desired AAL. This may be accomplished by repeating portions of the identity proofing process or by presenting one or more recovery codes, perhaps in conjunction with using an authenticator that is still available to the subscriber bound to their subscriber account. Once this is completed, the subscriber can bind one or more new authenticators to their subscriber account. An account recovery event always causes one or more notifications to be sent to the subscriber to aid in detecting the fraudulent use of account recovery.

Account recovery differs from authentication in several ways. Since account recovery is rarely expected to be invoked, it is generally less convenient than authentication and — depending on the situation and recovery methods offered by the CSP — may involve extended waiting times.

4.2.1. Account Recovery Methods

Four general classes of account recovery methods are recognized. CSPs **SHALL** support one or more of the following:

- Saved recovery codes
- Issued recovery codes
- Use of recovery contacts
- Repeated identity proofing

In addition to these methods, the CSP **MAY** support an application-specific method (e.g., interaction with a CSP agent) to recover a subscriber account. The use of alternative methods **SHALL** be based on a risk analysis and documented by the CSP.

4.2.1.1. Saved Recovery Codes

At enrollment, a CSP that supports this recovery option **SHOULD** issue a recovery code to the subscriber. The recovery code **SHALL** include at least 64 bits from an approved random bit generator. The saved recovery code may be presented as numeric or alphanumeric (e.g., Base64) for manual entry or as a machine-readable optical label (e.g., QR code) that contains the recovery code. At any point following enrollment, the subscriber **MAY** request a replacement recovery code. The issuance of a replacement recovery code **SHALL** result in an account recovery notification, as described in Sec. 4.6.

Saved recovery codes are intended to be maintained offline (e.g., printed or written down) and stored securely by the subscriber for future use. The verification of saved recovery codes **SHALL** be subject to the throttling requirements in Sec. 3.2.2. Saved recovery codes **SHALL** be stored in the subscriber account in hashed form using an approved one-way function, as described in Sec. 3.1.1.2. Following the use of a saved recovery code, the CSP **SHALL** invalidate that recovery code and **SHALL** issue a new saved recovery code to the subscriber.

4.2.1.2. Issued Recovery Codes

CSPs that support this option allow the subscriber to maintain one or more recovery addresses (e.g., postal, email, text message, or voice). When recovery is required, a recovery code will be sent to a claimant-chosen address. The issued recovery code **SHALL** include at least six decimal digits (or equivalent) from an approved random bit generator, as described in Sec. 3.2.12). The issued recovery code may be presented as a numeric or alphanumeric (e.g., Base64) for manual entry, a secure (e.g., https) link with a representation of the confirmation code, or a machine-readable optical label (e.g., QR code) that contains the recovery code.

Issued recovery codes **SHALL** be valid for at most:

- 21 days when sent to a postal address within the contiguous United States,
- 30 days when sent to a postal address outside the contiguous United States,
- 10 minutes when sent via text messaging or voice, or
- 24 hours when sent to an email address.

The verification of issued recovery codes **SHALL** be subject to the throttling requirements in Sec. 3.2.2.

When establishing recovery addresses, the CSP **SHALL** send a confirmation code with the same characteristics as a recovery code to the newly established recovery address. The recovery address **SHALL** be established only after the subscriber successfully confirms it. CSPs **SHALL** allow the subscriber to establish at least two recovery addresses.

4.2.1.3. Recovery Contacts

CSPs that support the use of recovery contacts **SHALL** allow the subscriber to specify one or more addresses of trusted associates to receive issued recovery codes. The requirements for recovery contacts are very similar to those for issued recovery codes with the following exceptions:

- The validity time for recovery codes sent to recovery contacts **MAY** be extended by 24 hours (i.e., valid for no more than 24 hours and 10 minutes if sent via text messaging) to provide additional time for the recovery contact to communicate the recovery code to the subscriber.
- Confirmation of the recovery code address **MAY** also be extended by 24 hours to allow the recovery contact to send the confirmation code to the subscriber for entry.

4.2.1.4. Repeated Identity Proofing

When the subscriber account has been identity proofed at a minimum of IAL1, CSPs **SHOULD** support account recovery by repeating a portion of the identity proofing process. The CSP **SHALL** repeat the necessary steps of identity proofing consistent with the level of initial identity proofing and **SHALL** confirm that the claimant's identity is consistent with the previously established account. If the CSP has retained a biometric sample from the user or a copy of the evidence used during the initial proofing and it is of sufficient quality and resolution, the CSP **MAY** repeat only the verification portion of the identity proofing process, as described in [SP800-63A].

4.2.2. Recovery Requirements by IAL/AAL

Different recovery methods apply depending on the IAL and the maximum AAL associated with the subscriber account.

4.2.2.1. Recovery at AAL1

Since identity proofing requires issuing authenticators that are sufficient for multi-factor authentication to allow the subscriber to access personal information about themselves, subscriber accounts at AAL1 are without identity proofing, and therefore, repeated identity proofing is not possible. The CSP **SHALL** require the successful use of a saved recovery code, issued recovery code, or recovery contact.

4.2.2.2. Recovery at AAL2

To recover an account that can authenticate at a maximum of AAL2, the CSP **SHALL** require the subscriber to complete one of the following:

- Two recovery codes obtained using different methods from the set (saved, issued, and recovery contacts)
- One recovery code from the set (saved, issued, and recovery contacts) plus authentication with a single-factor authenticator bound to the subscriber account
- Repeated identity proofing (provided that the subscriber account has been identity proofed)

4.2.2.3. Recovery at AAL3

If an account that can authenticate at AAL3 has been identity proofed at IAL1 or IAL2, the requirements are the same as those for recovery at AAL2.

If an account that can authenticate at AAL3 has been identity proofed at IAL3, the CSP **SHALL** successfully perform a successful biometric comparison against the biometric characteristic collected during the initial identity proofing session, in an onsite attended identity proofing session, as described in [SP800-63A]. The CSP **MAY** also require the presentation of evidence used in the initial identity proofing process.

4.2.3. Account Recovery Notification

In all cases, account recovery **SHALL** cause a notification to be sent to the subscriber, as described in Sec. 4.6.

4.3. Loss, Theft, Damage, and Compromise

Compromised authenticators include those that have been lost, stolen, or subject to unauthorized duplication or that have activation factors that are no longer in the subscriber's control. Generally, one must assume that a lost authenticator has been stolen or compromised by someone other than the legitimate holder of the authenticator. Damaged or malfunctioning authenticators are also considered compromised to guard against any possibility of the extraction of the authenticator's secret. One notable exception is a password that has been forgotten without other indications of having been compromised, such as having been obtained by an attacker.

The CSP **SHALL** suspend, invalidate, or destroy compromised authenticators from the subscriber's account promptly following compromise detection. Organizations **SHOULD** establish time limits for this process.

To facilitate the secure reporting of an authenticator's loss, theft, damage, or compromise, the CSP **SHOULD** provide the subscriber with a method of authenticating using a backup or alternate authenticator. This backup authenticator **SHALL** be a password or a physical authenticator. Either could be used, but only one authentication factor is required to make this report. Alternatively, the subscriber **MAY** establish an authenticated protected channel for the CSP to verify the information collected during identity proofing. The CSP **MAY** choose to verify a contact address (i.e., the email address, telephone number, or postal address) and suspend or invalidate authenticators that are reported to have been compromised.

CSPs **MAY** support the temporary suspension of authenticators that are suspected of possible compromise. If suspension is supported, it **SHOULD** be reversed if the subscriber successfully authenticates to the CSP using a valid (i.e., not suspended) authenticator and requests reactivation of the suspended authenticator. The CSP **MAY** set a time limit after which a suspended authenticator can no longer be reactivated.

4.4. Expiration

CSPs **MAY** issue authenticators that expire. If and when an authenticator expires, it **SHALL NOT** be usable for authentication. When an authentication is attempted using an expired authenticator, the CSP **SHOULD** indicate to the subscriber that the authentication failure is due to expiration rather than some other cause.

The CSP **SHOULD** retrieve any authenticator that contains personal information or provide for its zeroization (erasure) or destruction promptly following expiration.

The replacement of expired authenticators **SHALL** conform to the binding process for an additional authenticator, as described in Sec. 4.1.2.

4.5. Invalidation

The invalidation of an authenticator (sometimes referred to as revocation or termination) is the removal of the binding between the authenticator and a subscriber account.

CSPs **SHALL** promptly invalidate authenticators when a subscriber account ceases to exist (e.g., subscriber's death, the discovery of a fraudulent subscriber) when requested by the subscriber, when the authenticator is compromised, or when the CSP determines that the subscriber no longer meets its eligibility requirements. The CSP **SHALL** make a risk-based determination of the authenticity of invalidation requests from the subscriber, noting that the consequences of not invalidating a compromised authenticator are usually more significant than the denial-of-service potential of invalidating one in error.

The CSP **SHOULD** retrieve any authenticator that contains personal information or provide for its zeroization (erasure) or destruction promptly following invalidation.

Further requirements on the invalidation of PIV authenticators are found in [FIPS201].

4.6. Account Notifications

Certain subscriber account events, such as the binding of an authenticator and account recovery, require the subscriber to be independently notified. These notifications help the subscriber detect possible fraud associated with their subscriber account.

Events that require notification **SHALL** cause a notification to be sent to the notification addresses stored in the subscriber account. Notification addresses may be a:

- Postal address
- Email address
- Address (e.g., telephone number) to which a text message or voice message is to be sent

CSPs **SHALL** support at least two notification addresses per subscriber account, and at least one **SHALL** be validated during the identity proofing process. The CSP **SHOULD** allow subscribers with authentication at AAL2 or higher (or at AAL1 if that is the highest AAL available for the subscriber account) to update their notification addresses. The CSP **SHOULD** encourage the subscriber to maintain multiple notification addresses.

Notifications **SHALL** be sent to all notification addresses except postal addresses. However, notifications **SHALL** be sent to postal addresses if no other form of notification address is stored in the subscriber account or if the notification is for account recovery at AAL3.

The notification **SHALL** provide clear instructions, including contact information, in case the recipient repudiates the event associated with the notification.

5. Session Management

This section is normative.

Once an authentication event has occurred, it is often desirable to allow the subscriber to continue using the application across multiple subsequent interactions without requiring them to repeat the authentication event. This is particularly the case with federation scenarios (described in [SP800-63C]) in which the authentication event necessarily involves the coordination of several components and parties across a network.

To facilitate this behavior, a *session* **MAY** be started in response to an authentication event and continue until it is terminated. The session **MAY** be terminated for any number of reasons, including but not limited to an inactivity timeout or an explicit logout event. The session **MAY** be extended through a reauthentication event (described in Sec. 5.2) in which the subscriber repeats some of the initial authentication process or performs a full authentication, thereby reestablishing the authenticated session.

Session management is preferable to the continual presentation of credentials, as the poor usability of continual presentation often creates incentives for workarounds (e.g., caching activation factors), thereby negating authentication intent and obscuring the freshness of the authentication event.

5.1. Session Bindings

A session occurs between the software (i.e., the session subject) that a subscriber is running (e.g., browser, application, or operating system) and the RP or CSP that the subscriber is accessing (i.e., the session host). A session secret **SHALL** be shared between the subscriber's software and the accessed service. This secret binds the two ends of the session and allows the subscriber to continue using the service over time. The secret **SHALL** be presented directly by the subscriber's software, or possession of the secret **SHALL** be proven using a cryptographic mechanism.

The continuity of authenticated sessions **SHALL** be based upon the possession of a session secret that is issued by the verifier at the time of authentication and optionally refreshed during the session. The nature of a session depends on the application, such as:

- A web browser session with a "session" cookie or
- An instance of a mobile application that retains a session secret.

Session secrets **SHOULD NOT** be persistent (i.e., retained across a restart of the associated application or a reboot of the host device) because they are tied to specific sessions that a restart or reboot would end. Cookies and similar "remember my browser" features **SHALL NOT** be used instead of authentication except as provided for reauthentication at AAL2 in Sec. 2.2.3 when the inactivity limit has been exceeded but the time limit has not.

The secret used for session binding **SHALL** be generated by the session host in direct response to an authentication event. A session **SHOULD** inherit the AAL properties of the authentication event that triggered its creation. A session **MAY** be considered at a lower AAL than the authentication event but **SHALL NOT** be considered at a higher AAL than the authentication event.

The secrets used for session binding **SHALL** meet all of the following requirements:

1. Secrets are established during or immediately following authentication.
2. Secrets are established using input from an approved random bit generator as described in Sec. 3.2.12, and are at least 64 bits in length.
3. Secrets are erased or invalidated by the session subject when the subscriber logs out.
4. Secrets are either transferred from the session host to the RP or CSP via an authenticated protected channel or derived from keys that are established as part of establishing a valid, mutually authenticated protected channel.
5. Secrets will time out and are not accepted after the times specified in Sec. 2.1.3, Sec. 2.2.3, and Sec. 2.3.3, as appropriate for the AAL.
6. Secrets are unavailable to intermediaries between the host and the subscriber's endpoint.

In addition, secrets used for session binding **SHOULD** be erased on the subscriber endpoint when they log out or when the secret is deemed to have expired. They **SHOULD NOT** be placed in insecure locations (e.g., HTML5 Local Storage) due to the potential exposure of local storage to cross-site scripting (XSS) attacks.

Following authentication, authenticated sessions **SHALL NOT** fall back to an insecure transport (e.g., from https to http).

POST/PUT content **SHALL** contain a session identifier that the RP **SHALL** verify to protect against cross-site request forgery.

Several mechanisms exist for managing a session over time. The following sections give different examples, additional requirements, and considerations for each example technology. Additional informative guidance is available in the OWASP *Session Management Cheat Sheet* [OWASP-session].

Sessions **SHOULD** provide a readily accessible mechanism for subscribers to terminate (i.e., log off) their session when their interaction is complete. Session logoff gives the subscriber additional confidence and control over the security of their session, particularly in situations where the endpoint might be accessible to others.

5.1.1. Browser Cookies

Browser cookies are the predominant mechanism by which a session is created and tracked when a subscriber accesses a service. Cookies are not authenticators but are suitable as short-term secrets for the duration of a session.

Cookies used for session maintenance:

1. **SHALL** be tagged to be accessible only on secure (HTTPS) sessions.
2. **SHALL** be accessible to the minimum practical hostnames and paths.
3. **SHOULD** be tagged as inaccessible via JavaScript (HttpOnly).
4. **SHOULD** be tagged to expire at or soon after the session's validity period. This requirement is intended to limit the accumulation of cookies but **SHALL NOT** be relied upon to enforce session timeouts.
5. **SHOULD** have the "__Host-" prefix and set "Path=/".
6. **SHOULD** set "SameSite=Lax" or "SameSite=Strict".
7. **SHOULD** contain only an opaque string (e.g., a session identifier) and **SHALL NOT** contain cleartext personal information.

5.1.2. Access Tokens

An access token (e.g., OAuth [RFC6749]) is used to allow an application to access a set of services on a subscriber's behalf following an authentication event. The RP **SHALL NOT** interpret the presence of an OAuth access token as an indicator of the subscriber's presence in the absence of other signals. The OAuth access token and any associated refresh tokens could be valid long after the authentication session has ended and the subscriber has left the application.

5.2. Reauthentication

Periodic reauthentication of sessions **SHALL** be performed to confirm the subscriber's continued presence at an authenticated session (i.e., that the subscriber has not walked away without logging out).

Session management uses two types of timeouts. An *overall timeout* limits the duration of an authenticated session to a specific period following authentication or a previous reauthentication. An *inactivity timeout* terminates a session without activity from the subscriber for a specific period. For both types of timeouts, the RP **MAY** alert the subscriber that the session is about to be terminated and allow the subscriber to make the session active or reauthenticate as appropriate before the session expires. When either timeout expires, the session **SHALL** be terminated. Session activity **SHALL** reset the inactivity timeout, and successful reauthentication during a session **SHALL** reset both timeouts.

The overall and inactivity timeout expiration limits depend on several factors, including the AAL of the session, the environment in which the session is conducted (e.g., whether the subscriber is in a restricted area), the type of endpoint being used (e.g., mobile application or web-based), whether the endpoint is a managed device[5], and the nature of the application itself. Agencies **SHALL** establish and document the inactivity and overall time limits being enforced in a system security plan such as that described in [SP800-39].

Detailed requirements for each AAL are given in Sec. 2.1.3, Sec. 2.2.3, and Sec. 2.3.3.

Special considerations apply to session management and reauthentication when using a federation protocol and IdP to authenticate at the RP, as described in [SP800-63C], special considerations apply to session management and reauthentication. The federation protocol communicates an authentication event at the IdP to the RP using an assertion, and the RP then begins an authenticated session based on the successful validation of this assertion. Since the IdP and RP manage sessions separately from each other and the federation protocol does not connect the session management between the IdP and RP, the termination of the subscriber's sessions at an IdP and an RP are independent of each other. Likewise, the subscriber's sessions at multiple different RPs are established and terminated independently of each other.

Consequently, when an RP session expires and the RP requires reauthentication, it is possible that the session at the IdP has not expired and that a new assertion could be generated from this session at the IdP without explicitly reauthenticating the subscriber. The IdP can communicate the time and details of the authentication event to the RP, but it is up to the RP to determine whether the reauthentication requirements have been met. Section 4.7 of [SP800-63C] provides additional details and requirements for session management within a federation context.

[5]Managed devices include personal computers, laptops, mobile devices, virtual machines, or infrastructure components that are equipped with a management agent that allows information technology staff to discover, maintain, and control them.

5.3. Session Monitoring

Session monitoring (sometimes called *continuous authentication*) is the ongoing evaluation of session characteristics to detect possible fraud during a session.

Session monitoring **MAY** be performed by the RP, in coordination with the CSP/verifier, as a risk reduction measure. When potential fraud is detected during a session, the RP **SHOULD** take action in conjunction with the CSP/verifier, such as to reauthenticate, terminate the session, or notify appropriate support personnel. Session characteristics that **MAY** be evaluated include:

- Usage patterns, velocity, and timing
- Behavioral biometric characteristics (e.g., typing cadence)
- Device and browser characteristics
- Geolocation
- IP address characteristics (e.g., whether the IP address is in a block known for abuse)

Most of these characteristics have privacy implications. Collection, storage of expected subscriber characteristics, and processing of session characteristics **SHALL** be included in the privacy risk assessment described in Sec. 7.

6. Threats and Security Considerations

This section is informative.

6.1. Authenticator Threats

An attacker who can gain control of an authenticator will often be able to masquerade as the authenticator's owner. Threats to authenticators can be categorized based on attacks on the types of authentication factors that comprise the authenticator:

- "Something you know" may be disclosed to an attacker. For example, the attacker may guess a password. If the authenticator is a shared secret, the attacker could access the CSP or verifier and obtain the secret value or perform a dictionary attack on a hash of that value. An attacker may observe the entry of a PIN or passcode, find a written record or journal entry of a PIN or passcode, or install malicious software (e.g., a keyboard logger) to capture the secret. Additionally, an attacker may determine the secret through offline attacks on a password database maintained by the verifier.
- "Something you have" may be lost, damaged, stolen from the owner, or cloned by an attacker. For example, an attacker who gains access to the owner's computer may copy a software authenticator. A hardware authenticator may be stolen, tampered with, or duplicated. Out-of-band secrets may be intercepted by an attacker and used to authenticate their own session. A subscriber may be socially engineered to provide access to secrets without intentional collusion.
- "Something you are" may be replicated. For example, an attacker may obtain a copy of the subscriber's fingerprint and construct a replica.

Subscribers sometimes collude with attackers, and virtually nothing can be done from an authentication perspective to prevent these attacks. With this caveat in mind, threats to the authenticators used for digital authentication are listed in Table 2 along with some examples.

Table 2. Authenticator Threats

Authenticator Threat/Attack	Description	Examples
Theft	An attacker steals a physical authenticator.	A hardware cryptographic authenticator is stolen.
		An OTP authenticator is stolen.
		A look-up secret authenticator is stolen.
		A cell phone is stolen.

Duplication	The subscriber's authenticator has been copied with or without their knowledge.	Passwords written on paper are disclosed.
		Passwords stored in an electronic file are copied.
		A vulnerability in an insufficiently secure password manager is exploited.
		A software PKI authenticator (i.e., private key) is copied.
		A Look-up secret authenticator is copied.
		A counterfeit biometric authenticator is manufactured.
		Exportable cryptographic keys are obtained from a device or cloud-based sync fabric.
Eavesdropping	The attacker observes the authenticator secret or authenticator output as the subscriber is authenticating.	Passwords are obtained by watching keyboard entries.
		Passwords or authenticator outputs are intercepted by keystroke logging software.
		A PIN is captured from a PIN pad device.
		A hashed password is obtained and used by an attacker for another authentication (i.e., *pass-the-hash attack*).
	The attacker intercepts an out-of-band secret by compromising the communication channel.	An out-of-band secret is transmitted via unencrypted Wi-Fi and received by the attacker.
Offline Cracking	The authenticator is exposed using analytical methods outside of the authentication mechanism.	A software PKI authenticator is subjected to a dictionary attack to identify the correct password to decrypt the private key.

Side-Channel Attack	The authenticator's secret is exposed using the physical characteristics of the authenticator.	A key is extracted by differential power analysis on a hardware cryptographic authenticator.
		A cryptographic authenticator secret is extracted by analysis of the authenticator's response time over several attempts.
Phishing or Pharming	The authenticator output is captured by fooling the claimant into thinking that the attacker is a verifier or RP.	A claimant reveals a password to a website impersonating the verifier.
		A password is revealed by a bank subscriber in response to an email inquiry from a phisher pretending to represent the bank.
		A password is revealed by the claimant at a bogus verifier website reached through DNS spoofing.
Social Engineering	The attacker establishes a level of trust with a subscriber to convince them to reveal their authenticator secret or authenticator output.	A password is revealed by the subscriber to an officemate asking for the password on behalf of the subscriber's boss.
		A password is revealed by a subscriber in a telephone inquiry from an attacker masquerading as a system administrator.
		An attacker who has convinced the mobile operator to redirect the victim's mobile phone to them receives an out-of-band secret sent via SMS.

		A subscriber erroneously approves a push-based authentication request coming from a repeated “fatigue” attack.
Online Guessing	The attacker connects to the verifier online and attempts to guess a valid authenticator output in the context of that verifier.	Online dictionary attacks are used to guess passwords.
		Online guessing is used to guess authenticator outputs for an OTP authenticator that is registered to a legitimate subscriber.
Endpoint Compromise	Malicious code on the endpoint proxies allow remote access to a connected authenticator without the subscriber’s consent.	A cryptographic authenticator connected to the endpoint is used to authenticate remote attackers.
	Malicious code on the endpoint causes authentication to other than the intended verifier.	Authentication is performed on behalf of an attacker rather than the subscriber.
		A malicious app on the endpoint reads an out-of-band secret sent via SMS, and the attacker uses the secret to authenticate.
	Malicious code on the endpoint compromises a multi-factor software cryptographic authenticator.	Malicious code proxies authenticate or export authenticator keys from the endpoint.
Unauthorized Binding	An attacker causes an authenticator under their control to be bound to a subscriber account.	An attacker intercepts an authenticator or provisioning key en route to the subscriber.

Latent Keys	A decommissioned device retains authentication keys	A device (e.g., laptop computer) is sold without recognition that device-based authentication keys are present and could be used by a new owner.
Proliferation of Keys	Transferring device-based authentication keys between devices increases the attack surface	A subscriber copies authentication keys to many devices, possibly some that are not under their direct control, and loses track of where the keys are stored
Key Transfer Security	Authentication keys are transferred between devices through an insufficiently secure cloud service	Access to a cloud service that stores authentication keys requires only single-factor authentication
		Keys are made available to others through a URL sent via email
Insider Threats	An insider with access to the CSP (e.g., customer support representative) colludes with an attacker to give access to subscriber accounts.	

6.2. Threat Mitigation Strategies

Table 3 summarizes related mechanisms that assist in mitigating the threats described in Table 2.

Table 3. Mitigating Authenticator Threats

Authenticator Threat/Attack	Threat Mitigation Mechanisms	Normative Reference Sections
Theft	Use multi-factor authenticators that must be activated through a password or biometric.	2.2.1, 2.3.1
	Use a combination of authenticators that includes a password or biometric.	2.2.1, 2.3.1

Duplication	Use authenticators from which it is difficult to extract and duplicate long-term authentication secrets.	2.2.2, 2.3.2, 3.1.6.1
	Enforce AAL2 requirements for access to sync fabrics that contain exported authentication keys, and only allow them to be imported into trusted devices.	3.1.7.1
Eavesdropping	Ensure the endpoint's security before use, especially with respect to freedom from malware (e.g., such as key loggers).	2.2.2
	Avoid using unauthenticated and unencrypted communication channels to send out-of-band authenticator secrets.	3.1.3.1
	Authenticate over authenticated protected channels (e.g., observe the lock icon in the browser window).	2.1.2, 2.2.2, 2.3.2
	Use authentication protocols that are resistant to replay attacks (e.g., *pass-the-hash*).	3.2.7
	Use authentication endpoints that employ trusted input and display capabilities.	3.1.6.1, 3.1.7.1
Offline Cracking	Use an authenticator with a high entropy authenticator secret.	3.1.2.1, 3.1.4.1, 3.1.5.1, 3.1.6.1, 3.1.7.1
	Store centrally verified passwords in a salted, hashed form, including a keyed hash.	3.1.1.1.2
Side-Channel Attack	Use authenticator algorithms that maintain constant power consumption and timing regardless of secret values.	2.3.2

Phishing or Pharming	Use authenticators that provide phishing resistance.	3.2.5
Social Engineering	Avoid using authenticators that present a social engineering risk to third parties (e.g., customer service agents).	4.1.2.1, 4.2
Online Guessing	Use authenticators that generate high entropy output.	3.1.2.1, 3.1.6.1, 3.1.7.1
	Use an authenticator that locks after repeated failed activation attempts.	3.2.2
Endpoint Compromise	Use hardware authenticators that require physical action by the claimant.	3.2.8
	Maintain software-based keys in restricted-access storage.	3.1.3.1, 3.1.6.1, 3.1.7.1, 3.2.13
Unauthorized Binding	Provision authenticators and associated keys using authenticated protected channels or in person.	4.1
Latent Keys	Ensure the secure disposal of equipment that contains device-based authentication keys	4.4, 4.5
	In enterprise applications, limit the transfer of keys to organizationally managed or trusted devices	B.2
Key Transfer Security	Encourage or require subscribers to use cloud services that have been approved for key storage and transfer	B.2

Several other strategies may be applied to mitigate the threats described in Table 3:

- *Multiple factors* make successful attacks more difficult to accomplish. If an attacker must steal a cryptographic authenticator and guess a password, then the work to discover both factors may be too high.

- *Physical security mechanisms* may be employed to protect a stolen authenticator from duplication. Physical security mechanisms can provide tamper evidence, detection, and response.
- *Requiring long passwords* that do not appear in common dictionaries may force attackers to try every possible value.
- *System and network security controls* may be employed to prevent an attacker from gaining access to a system or installing malicious software.
- *Periodic training* may be performed to ensure that subscribers understand when and how to report a compromise or a suspicion of compromise and to recognize patterns of behavior that may signify that an attacker is attempting to compromise the authentication process.
- *Out-of-band techniques* may be employed to verify the proof of possession of registered devices (e.g., cell phones).

6.3. Authenticator Recovery

The weak point in many authentication mechanisms is the process followed when a subscriber loses control of one or more authenticators and needs to replace them. In many cases, the options for authenticating the subscriber are limited, and economic concerns (e.g., the cost of maintaining call centers) motivate the use of inexpensive and often less secure backup authentication methods. To the extent that authenticator recovery is human-assisted, social engineering attacks are also risky.

To maintain the integrity of the authentication factors, it is essential that one authentication factor cannot be leveraged to obtain an authenticator of a different factor. For example, a password must not be usable to obtain a new list of look-up secrets.

6.4. Session Attacks

Hijacking attacks on the session following an authentication event can have similar security impacts. The session management guidelines in Sec. 5 are essential to maintaining session integrity against attacks (e.g., XSS). It is also important to sanitize all information to be displayed [OWASP-XSS-prevention] to ensure that it does not contain executable content. These guidelines recommend that session secrets be made inaccessible to mobile code to provide extra protection against the exfiltration of session secrets.

Another post-authentication threat is cross-site request forgery (CSRF), which takes advantage of users' tendency to have multiple sessions active simultaneously. It is essential to embed and verify a session identifier for web requests to prevent a valid URL or request from being unintentionally or maliciously activated.

7. Privacy Considerations

These privacy considerations supplement the guidance in Sec. 4. This section is informative.

7.1. Privacy Risk Assessment

The authentication requirements in Sec. 2 and the optional session monitoring guidelines in Sec. 5.3 require the CSP to conduct a privacy risk assessment for records retention. Such a privacy risk assessment would include:

1. The likelihood that the records retention could create a problem for the subscriber, such as invasiveness or unauthorized access to the information.
2. The impact if such a problem did occur.

CSPs should be able to reasonably justify any response to identified privacy risks, including accepting, mitigating, and sharing the risk. Subscriber consent is a form of sharing the risk. It is therefore only appropriate for use when a subscriber could reasonably be expected to have the capacity to assess and accept the shared risk.

7.2. Privacy Controls

Section 2.4.3 requires CSPs to employ appropriately tailored privacy controls. [SP800-53] provides a set of privacy controls for CSPs to consider when deploying authentication mechanisms, including notices, redress, and other important considerations for successful and trustworthy deployments.

7.3. Use Limitation

Section 2.4.3 requires CSPs to maintain the objectives of predictability (enabling reliable assumptions by individuals, owners, and operators about PII and its processing by an information system) and manageability (i.e., providing the capability for the granular administration of PII, including alteration, deletion, and selective disclosure) commensurate with privacy risks that can arise from the processing of attributes for purposes other than identity proofing, authentication, authorization, or attribute assertion; related fraud mitigation; or to comply with law or legal process [NISTIR8062].

CSPs may have various business purposes for processing attributes, including providing non-identity services to subscribers. However, processing attributes for purposes other than those specified at collection can create privacy risks. CSPs can identify appropriate measures that are commensurate with the privacy risks that arise from additional processing. For example, absent applicable laws, regulations, or policies, obtaining consent may not be necessary when processing attributes to provide non-identity services requested by subscribers. However, notices may help subscribers maintain reliable assumptions about the processing (i.e., predictability). Other processing of attributes may carry different privacy risks that call for obtaining consent or allowing subscribers more control over the use or disclosure of specific attributes (i.e., manageability). Subscriber consent must be meaningful. Therefore, as stated in Sec. 2.4.3, when CSPs use consent measures, the subscriber's acceptance of additional uses shall not be a condition of providing authentication services.

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

7.4. Agency-Specific Privacy Compliance

Section 2.4.3 describes specific compliance obligations for federal CSPs. It is critical to involve the agency SAOP in the earliest stages of digital authentication system development to assess and mitigate privacy risks and advise the agency on compliance requirements, such as whether or not the collection of PII to issue or maintain authenticators triggers the *Privacy Act of 1974* [PrivacyAct] or the *E-Government Act of 2002* [E-Gov] requirement to conduct a PIA. For example, concerning the centralized maintenance of biometrics, Privacy Act requirements will likely be triggered and require coverage by a new or existing Privacy Act system of records notice due to the collection and maintenance of PII and any other attributes that are necessary for authentication. The SAOP can similarly assist the agency in determining whether a PIA is required.

These considerations should not be read as a requirement to develop a Privacy Act SORN or PIA for authentication alone. In many instances, a PIA and SORN can encompass the entire digital identity process or include the digital authentication process as part of a larger programmatic PIA that discusses the online services or benefits that the agency is establishing.

Due to the many components of digital authentication, the SAOP needs to be aware of and understand each component. For example, other privacy artifacts may apply to an agency that offers or uses federated CSP or RP services (e.g., Data Use Agreements, Computer Matching Agreements). The SAOP can assist the agency in determining what additional requirements apply. Moreover, a thorough understanding of the individual components of digital authentication will enable the SAOP to assess and mitigate privacy risks through compliance processes or other means.

8. Usability Considerations

This section is informative.

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

[ISO/IEC9241-11] defines usability as the “extent to which a system, product, or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” This definition focuses on users, their goals, and the contexts of use as the key elements necessary for achieving effectiveness, efficiency, satisfaction, and usability.

A user’s goal when accessing an information system is to perform an intended task. Authentication is the function that enables this goal. However, from the user’s perspective, authentication stands between them and their intended task. Effective design and implementation of the authentication process makes it easy to do the right thing, hard to do the wrong thing, and easy to recover if the wrong thing happens.

Organizations need to be cognizant of the overall implications of their stakeholders’ entire digital authentication ecosystem. Users often employ multiple authenticators, each for a different RP. They then struggle to remember passwords, recall which authenticator goes with which RP, and carry multiple physical authentication devices. Evaluating the usability of authentication is critical, as poor usability often results in coping mechanisms and unintended workarounds that can ultimately degrade the effectiveness of security controls.

Integrating usability into the development process can lead to authentication solutions that are secure and usable while still addressing users’ authentication needs and organizations’ business goals. The impacts of usability across digital systems needs to be considered as part of the risk assessment when deciding on the appropriate AAL. Authenticators with a higher AAL sometimes offer better usability and should be allowed for use with lower AAL applications.

Leveraging federation for authentication can alleviate many usability issues, though such an approach has its tradeoffs, as discussed in [SP800-63C].

This section provides general usability considerations and possible implementations but does not recommend specific solutions. The implementations mentioned are examples that encourage innovative technological approaches to address specific usability needs. Furthermore, usability considerations and their implementations are sensitive to many factors that prevent a one-size-fits-all solution. For example, a font size that works in a desktop computing environment may force text to scroll off of a small OTP authenticator screen. Performing a usability evaluation on the selected authenticator is a critical component of implementation. It is important to conduct evaluations with representative users, set realistic goals and tasks, and identify appropriate contexts of use.

Guidelines and considerations are described from the users' perspective.

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

8.1. Common Usability Considerations for Authenticators

When selecting and implementing an authentication system, consider usability across the entire lifetime of the selected authenticators (e.g., their typical use and intermittent events) while being mindful of users, their goals, and their contexts of use.

A single authenticator type does not usually suffice for the entire user population. Therefore, whenever possible and based on AAL requirements, CSPs should support alternative authenticator types and allow users to choose the type that best meets their needs. Task immediacy, perceived cost-benefit trade-offs, and unfamiliarity with certain authenticators often impact choices. Users tend to choose options that incur the least burden or cost at that moment. For example, if a task requires immediate access to an information system, a user may prefer to create a new subscriber account and password rather than select an authenticator that requires more steps. Alternatively, users may choose a federated identity option that is approved at the appropriate IAL, AAL, and FAL if they already have a subscriber account with an identity provider. Users may understand some authenticators better than others and have different levels of trust based on their understanding and experience.

Positive user authentication experiences are integral to achieving desired business outcomes. Therefore, organizations should strive to consider authenticators from the users' perspective. The overarching authentication usability goal is to minimize user burden and authentication friction (e.g., the number of times a user has to authenticate, the steps involved, and the amount of information they have to track). Single sign-on exemplifies one such minimization strategy.

Usability considerations applicable to most authenticators are described below. Subsequent sections describe usability considerations specific to a particular authenticator.

Usability considerations that are applicable to most authenticators include:

- Provide information on the use and maintenance of the authenticator (e.g., what to do if the authenticator is lost or stolen), and instructions for use, especially if there are different requirements for first-time use or initialization.
- Authenticator availability, as users will need to remember to have their authenticator readily available. Consider the need for alternative authentication options to protect against loss, damage, or other negative impacts on the original authenticator and the potential loss of battery power, if applicable.
- Alternative authentication options whenever possible and based on AAL requirements. This allows users to choose an authenticator based on their context, goals, and tasks (e.g., the frequency and immediacy of the task). Alternative authentication options also help address availability issues that may occur with a particular authenticator.
- Characteristics of user-facing text:
  - Write user-facing text (e.g., instructions, prompts, notifications, error messages) in plain language for the intended audience. Avoid technical jargon, and write for the audience's expected literacy level.
  - Consider the legibility of user-facing and user-entered text, including font style, size, color, and contrast with the surrounding background. Illegible text contributes to user entry errors. To enhance legibility, consider the use of:
    - High contrast (i.e., black on white)
    - Sans serif fonts for electronic displays and serif fonts for printed materials.
    - Fonts that clearly distinguish between characters that are easily confused (e.g., the capital letter "O" and the number zero "0")
    - A minimum font size of 12 points as long as the text fits for display on the device
  - Avoid using icons (e.g., padlocks or shields) that might be confused with security indicators in browsers.
- User experience during authenticator entry:
  - Offer the option to display text during entry, as masked text entry is error-prone. Once a given character is displayed long enough for the user to see, it can be hidden. Consider the device when determining masking delay time, as it takes longer to enter passwords on mobile devices (e.g., tablets and smartphones) than on traditional desktop computers. Ensure that masking delay durations are consistent with user needs.

- Ensure that the time allowed for text entry is adequate (i.e., the entry screen does not time out prematurely). Ensure that the allowed text entry times are consistent with user needs.
- Provide clear, meaningful, and actionable feedback on entry errors to reduce user confusion and frustration. Significant usability implications arise when users do not know that they have entered text incorrectly.
- Allow at least 10 entry attempts for authenticators that require the entry of the authenticator output by the user. The longer and more complex the entry text, the greater the likelihood of user entry errors.
- Provide clear, meaningful feedback on the number of remaining allowed attempts. For rate limiting (i.e., throttling), inform users how long they have to wait until the next attempt.

- Minimize the impact of form-factor constraints, such as limited touch and display areas on mobile devices:
  - Larger touch areas improve usability for text entry since typing on small devices is significantly more error-prone and time-consuming than typing on a full-size keyboard due to the size of the input mechanism (e.g., a finger) relative to the size of the on-screen target.
  - Follow good user interface and information design for small displays.

Usability considerations for intermittent events (e.g., reauthentication, subscriber account lock-out, expiration, revocation, damage, loss, theft, and non-functional software) across authenticator types include:

- Prompt users to perform some activity just before (e.g., two minutes before) an inactivity timeout would otherwise occur.
- Prompt users to save their work before a fixed reauthentication timeout occurs regardless of user activity.
- Clearly communicate how and where to acquire technical assistance (e.g., provide users with a link to an online self-service feature, chat sessions, or a phone number for help desk support). Ideally, sufficient information can be provided to enable users to recover from intermittent events on their own without outside intervention.
- Provide an accessible means for the subscriber to end their session (i.e., logoff).

8.2. Usability Considerations by Authenticator Type

The following sections describe other usability considerations that are specific to particular authenticator types.

8.2.1. Passwords

Typical Usage

Users often manually input the password (sometimes referred to as a passphrase or PIN). Alternatively, they may use a password manager to assist in the selection of a secure password and in maintaining distinct passwords for each authenticated service. The use of distinct passwords is important to avoid “password stuffing” attacks in which an attacker uses a compromised password from one site on other sites where the user might also have an account. Agencies should carefully evaluate password managers before making recommendations or mandates to confirm that they meet expectations for secure implementation.

Usability considerations for typical usage without a password manager include:

- Memorability of the password
  - The likelihood of a recall failure increases as there are more items for users to remember. With fewer passwords, users can more easily recall the specific password needed for a particular RP.
  - The memory burden is greater for a less frequently used password.
- User experience during entry of the password
  - Support copy and paste functionality in fields for entering passwords, including passphrases.

Intermittent Events

Usability considerations for intermittent events include:

- When users create and change passwords
  - Clearly communicate information on how to create and change passwords.
  - Clearly communicate password requirements, as specified in Sec. 3.1.1.
  - Allow at least 64 characters in length to support the use of passphrases. Encourage users to make passwords as lengthy as they want and use any characters that they like (including spaces) to aid memorization. Ensure that user interfaces support sufficient password lengths.
  - Do not impose other composition rules (e.g., mixtures of different character types) on passwords.
  - Do not require that passwords be changed arbitrarily (e.g., periodically) unless there is a user request or evidence of authenticator compromise (see Sec. 3.1.1 for additional information).
- Provide clear, meaningful, and actionable feedback when chosen passwords are rejected (e.g., when it appears on a “blocklist” of unacceptable passwords or has been used previously).

8.2.2. Look-Up Secrets

Typical Usage

Subscribers use a printed or electronic authenticator to look up the appropriate secrets needed to respond to a verifier's prompt. For example, a user may be asked to provide a specific subset of the numeric or character strings printed on a card in table format.

Usability considerations for typical usage include:

- User experience during entry of look-up secrets.
  - Consider the complexity and size of the prompts. There are greater usability implications with larger subsets of secrets that a user is prompted to look up. Both the cognitive workload and physical difficulty for entry should be taken into account.

8.2.3. Out-of-Band

Typical Usage

Out-of-band authentication requires that users have access to a primary and secondary communication channel.

Usability considerations for typical usage include:

- Notify users of the receipt of a secret on a lockable device. If the out-of-band device is locked, authentication to the device should be required to access the secret.
- Depending on the implementation, consider form-factor constraints, which are particularly problematic when users must enter text on mobile devices. Providing larger touch areas will improve usability for entering secrets on mobile devices.
- Consider offering features that do not require text entry on mobile devices (e.g., a copy-paste feature), which are particularly helpful when the primary and secondary channels are on the same device. For example, it is difficult for users to transfer the authentication secret manually using a smartphone because they must switch back and forth — potentially multiple times — between the out-of-band application and the primary channel.
- Messages and notifications to out-of-band devices should contain contextual information for the user, such as the name of the service being accessed.
- Out-of-band messages should be delivered in a consistent manner and style to aid the subscriber in identifying potentially suspicious authentication requests.

8.2.4. Single-Factor OTP

Typical Usage

Users access the OTP generated by the single-factor OTP authenticator. The authenticator output is typically displayed on the authenticator, and the user enters it during the session being authenticated.

Usability considerations for typical usage include:

- Authenticator output allows at least one minute between changes but ideally allows users two full minutes, as specified in Sec. 3.1.4.1. Users need adequate time to enter the authenticator output, including looking back and forth between the single-factor OTP authenticator and the entry screen.
- Depending on the implementation, the following are additional usability considerations for implementers:
  - It is preferable for the single-factor OTP authenticator to supply its output via an electronic interface (e.g., USB port) so that users do not have to manually enter the authenticator output. However, if a physical input (e.g., pressing a button) is required to operate, the location of the USB ports could pose usability difficulties. For example, the USB ports of some computers are located on the back of the computer and may be difficult for users to reach.
  - Limited availability of a direct computer interface (e.g., USB port) could pose usability difficulties. For example, the number of USB ports on laptop computers is often very limited. This may force users to unplug other USB peripherals to use the single-factor OTP authenticator.

8.2.5. Multi-Factor OTP

Typical Usage

Users access the OTP generated by the multi-factor OTP authenticator through a second authentication factor. The OTP is typically displayed on the device, and the user manually enters it during the session being authenticated. The second authentication factor may be achieved through some kind of integral entry pad to enter a password, an integral biometric (e.g., fingerprint) reader, or a direct computer interface (e.g., USB port). Usability considerations for the additional factor also apply (see Sec. 8.2.1 for passwords and Sec. 8.4 for biometrics used in multi-factor authenticators).

Usability considerations for typical usage include:

- User experience during manual entry of the authenticator output
  - For time-based OTP, provide a grace period in addition to the time during which the OTP is displayed. Users need adequate time to enter the authenticator output, including looking back and forth between the multi-factor OTP authenticator and the entry screen.
  - Consider form-factor constraints if users must unlock the multi-factor OTP authenticator via an integral entry pad or enter the authenticator output on mobile devices. Typing on small devices is significantly more error-prone and time-consuming than typing on a traditional keyboard. Providing larger touch areas improves usability for unlocking the multi-factor OTP authenticator or entering the authenticator output on mobile devices.
  - Limited availability of a direct computer interface (e.g., USB port) could pose usability difficulties. For example, laptop computers often have a limited number of USB ports, which may force users to unplug other USB peripherals to use the multi-factor OTP authenticator.

8.2.6. Single-Factor Cryptographic Authenticator

Typical Usage

Users authenticate by proving possession and control of the cryptographic key.

Usability considerations for typical usage include:

- Give cryptographic keys appropriately descriptive names that are meaningful to users so that they can recognize and recall which cryptographic key to use for which authentication task. This prevents users from having to deal with multiple similarly and ambiguously named cryptographic keys. Selecting from multiple cryptographic keys on smaller mobile devices may be particularly problematic if the names of the cryptographic keys are shortened due to reduced screen sizes.
- Requiring a physical input (e.g., pressing a button) to operate a single-factor cryptographic authenticator could pose usability difficulties. For example, some USB ports are located on the back of computers, making it difficult for users to reach the port.
- For connected authenticators, the limited availability of a direct computer interface (e.g., USB port) could pose usability difficulties. For example, laptop computers often have a limited number of USB ports, which may force users to unplug other USB peripherals to use the authenticator.

8.2.7. Multi-Factor Cryptographic Authenticator

Typical Usage

To authenticate, users prove possession and control of the cryptographic key and control of the activation factor. Usability considerations for the additional factor also apply (see Sec. 8.2.1 for passwords and Sec. 8.4 for biometrics used as activation factors).

Usability considerations for typical usage include:

- Give cryptographic keys appropriately descriptive names that are meaningful to users so that they can recognize and recall which cryptographic key to use for which authentication task. This prevents users from having to deal with multiple similarly and ambiguously named cryptographic keys. Selecting from multiple cryptographic keys on smaller mobile devices may be particularly problematic if the names of the cryptographic keys are shortened due to reduced screen sizes.
- Do not require users to keep external multi-factor cryptographic authenticators connected following authentication. Users may forget to disconnect the authenticator when they are done with it (e.g., forgetting a smartcard in the smartcard reader and walking away from the computer).
  - Users need to be informed about whether the authenticator is required to stay connected or not.
- For connected authenticators, the limited availability of a direct computer interface (e.g., USB port) could pose usability difficulties. For example, laptop computers often have a limited number of USB ports, which may force users to unplug other USB peripherals to use the authenticator.

8.3. Summary of Usability Considerations

Figure 4 summarizes the usability considerations for typical usage and intermittent events for each authenticator type. Many of the usability considerations for typical usage apply to most of the authenticator types, as demonstrated in the rows. The table highlights common and divergent usability characteristics across the authenticator types. Each column allows readers to easily identify the usability attributes to address for each authenticator. Depending on the users' goals and context of use, certain attributes may be valued over others. Whenever possible, provide alternative authenticator types, and allow users to choose between them.

Multi-factor authenticators (e.g., multi-factor OTPs and multi-factor cryptographic) also inherit their activation factor's usability considerations. As biometrics are only allowed as an activation factor in multi-factor authentication solutions, usability considerations for biometrics are not included in Fig. 4 and are discussed in Sec. 8.4.

Usability Considerations	Passwords	Look-Up Secrets	Out-of-Band Devices	Single-Factor OTP	Multi-Factor OTP	Single-Factor Cryptographic	Multi-Factor Cryptographic
Typical Usage							
Authenticator availability - authenticators readily in user's possession	•	•	•	•	•	•	•
Plain language for user-facing text (e.g., instructions, prompts, notifications, error messages)	•	•	•	•	•	•	•
Legibility of user-facing text or text entered by users	•	•	•	•	•	•	•
Unmasked text entry		•	•	•	•		
Support text entry - length of 64 characters, copy and paste	•						
Delayed masking during text entry	•						
Adequate time allowed for text entry	•	•	•	•	•		
Entry errors - Need clear and meaningful feedback	•	•	•	•	•		
Minimum of 10 attempts allowed	•	•	•	•	•		
Remaining allowed attempts - need clear and meaningful feedback	•	•	•	•	•		
Form factor constraints	•	•	•	•	•	•	•
Location and availability of a direct computer interface such as a USB port						•	•
Physical input required (such as pressing a button)						•	
Cryptographic keys need for descriptive and meaningful names						•	•
Complexity and size of the prompts		•					
Authentication to secondary device to access the authentication secret			•				
Continuous hardware connection not required							•
Intermittent Events							
Reauthentication due to user inactivity	•	•	•	•	•	•	•
Fixed periodic reauthentication	•	•	•	•	•	•	•
Provisions for technical assistance	•	•	•	•	•	•	•
Provisions to create and change passwords	•						

Fig. 4. Usability considerations by authenticator type

8.4. Usability Considerations for Biometrics

This section provides a high-level overview of general usability considerations for biometrics. A more detailed discussion of biometric usability can be found in *Usability & Biometrics*, *Ensuring Successful Biometric Systems* [UsabilityBiometrics].

User familiarity and practice with the device improve performance for all modalities. Device affordances (i.e., properties of a device that allow a user to perform an action), feedback, and clear instructions are critical to a user's success with the biometric device. For example, provide clear instructions on the required actions for liveness detection. Ideally, users can select the modality that they are most comfortable with for their second authentication factor. Various user populations may be more comfortable, familiar with, and accepting of some biometric modalities than others. Additionally, user experience with biometrics is an activation factor. Provide clear, meaningful feedback on the number of remaining allowed attempts. For example, for rate limiting (i.e., throttling), inform users of the time period they have to wait until their next attempt.

Typical Usage

The three biometric modalities that are most commonly used for authentication are fingerprint, face, and iris.

- Fingerprint usability considerations:
  - Users have to remember which fingers they used for initial enrollment.
  - The amount of moisture on the finger affects the sensor's ability for successful capture.
  - Additional factors that influence fingerprint capture quality include age, gender, and occupation (e.g., users who handle chemicals or work extensively with their hands may have degraded friction ridges).
- Face usability considerations:
  - Users have to remember whether they wore any artifacts (e.g., glasses) during enrollment, which affects facial recognition accuracy.
  - Differences in environmental lighting conditions may affect facial recognition accuracy.
  - Facial expressions affect facial recognition accuracy (e.g., smiling versus a neutral expression).
  - Facial poses affect facial recognition accuracy (e.g., looking down or away from the camera).
- Iris usability considerations:
  - Wearing colored contacts may affect iris recognition accuracy.

- Users who have had eye surgery may need to re-enroll after surgery.
- Differences in environmental lighting conditions may affect iris recognition accuracy, especially for certain iris colors.

Intermittent Events

Since biometrics are only permitted as a second factor for multi-factor authentication, usability considerations for intermittent events with the primary factor still apply. Intermittent events that may affect recognition accuracy using biometrics include:

- Degraded fingerprints or finger injuries
- Dirty, dry, or wet hands; wearing gloves; or wearing a mask
- Natural facial or weight changes over time
- Eye surgery

Across all biometric modalities, usability considerations for intermittent events include:

- An alternative authentication method must be readily available and clearly communicated. Users should never be required to attempt biometric authentication and should be permitted to use a password as an alternative second factor.
- There should be provisions for technical assistance:
    - Clearly communicate information on how and where to acquire technical assistance. For example, provide users with a link to an online self-service feature or a phone number for help desk support. Ideally, provide sufficient information to enable users to recover from intermittent events on their own without outside intervention.
    - Inform users of factors that may affect the sensitivity of the biometric sensor (e.g., cleanliness of the sensor).

9. Equity Considerations

This section is informative.

Accurate and equitable authentication service is an essential element of a digital identity system. While the accuracy aspects of authentication are primarily the subject of the security requirements found elsewhere in this document, the ability for all subscribers to reliably authenticate is required to provide equitable access to government services, as specified in Executive Order 13985, *Advancing Racial Equity and Support for Underserved Communities Through the Federal Government* [EO13985]. When assessing equity risks, a CSP should consider the overall user population for its authentication service. Additionally, the CSP further identifies groups of users within the population whose shared characteristics may cause them to be subject to inequitable access, treatment, or outcomes when using that service. Section 8 describes considerations to help ensure the overall usability and equity for all persons who use authentication services.

A primary aspect of equity is that the CSP needs to anticipate the needs of its subscriber population and offer authenticator options that are suitable for that population. Some examples of authenticator suitability problems are:

- SMS-based out-of-band authentication may not be usable for subscribers in rural areas without mobile phone service.
- OTP authenticators may be difficult for subscribers with vision issues to read.
- Out-of-band authentication secrets sent via a voice telephone call may be difficult for subscribers with hearing difficulties to understand.
- Facial matching algorithms may not match the facial characteristics of all ethnicities or those wearing glasses equally well.
- Some subscribers may be missing fingers, have degraded fingerprints (e.g., from working with chemicals or extensively using their hands), or have dexterity problems that interfere with fingerprint collection.
- The cost of hardware-based authenticators may be beyond the means of some subscribers.
- Accurate manual entry of passwords may be difficult for subscribers with mobility and dexterity-related physical disabilities.
- Certain authenticator types may be challenging for subscribers with intellectual, developmental, learning, or neurocognitive difficulties.
- Lower-income subscribers are less likely to have up-to-date devices that are required by some authentication modes.
- Lower-income subscribers may be limited to the use of a smartphone and, therefore, may be unable to use USB-connected authenticators.

- Subscribers with less technological skill may need help to enter OTP codes from one device to another.
- Older subscribers may need help with the small form factor of some authenticators.
- Subscribers experiencing addiction, sexual exploitation, or other trauma may struggle to remember passwords or activation secrets.

While CSPs are required to mitigate the common and expected problems in this area, it is not feasible to anticipate all potential equity problems, which will vary for different applications. Accordingly, CSPs need to provide mechanisms for subscribers to report inequitable authentication requirements and advise them on potential alternative authentication strategies.

This guideline recommends the binding of additional authenticators to minimize the need for account recovery (see Sec. 4.2). However, a subscriber may need help to purchase a second hardware-based authenticator as a backup. This inequity can be addressed by making inexpensive authenticators such as look-up secrets (see Sec. 3.1.2) available for use in the event of an authenticator failure or loss.

CSPs need to be responsive to subscribers who experience authentication challenges that cannot be solved using the authenticators that they currently support. This might involve supporting a new authenticator type or allowing federated authentication through a trusted service that meets the subscriber's needs.

References

This section is informative.

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

[Blocklists] Habib H, Colnago J, Melicher W, Ur B, Segreti S, Bauer L, Christin N, Cranor L (2017) Password Creation in the Presence of Blacklists. *Proceedings 2017 Workshop on Usable Security* (Internet Society, San Diego, CA). https://doi.org/10.14722/usec.2017.23043

[Composition] Komanduri S, Shay R, Kelley PG, Mazurek ML, Bauer L, Christin N, Cranor LF, Egelman S (2011) Of Passwords and People: Measuring the Effect of Password-Composition Policies. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, NY), pp 2595–2604. Available at https://www.ece.cmu.edu/~lbauer/papers/2011/chi2011-passwords.pdf

[CTAP2.2] Bradley J, Hodges J, Jones MB, Kumar A, Lindemann R, Verrept J (2023) Client to Authenticator Protocol (CTAP), version 2.2. (FIDO Alliance, Beaverton, OR) Available at https://fidoalliance.org/specs/fido-v2.2-rd-20230321/fido-client-to-authenticator-protocol-v2.2-rd-20230321.html

[E-Gov] E-Government Act of 2002, P.L. 107-347, 116 Stat. 2899, 44 U.S.C. § 101 (2002). Available at https://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf

[EO13681] Obama B (2014) Improving the Security of Consumer Financial Transactions. (The White House, Washington, DC), Executive Order 13681, October 17, 2014. Available at https://www.federalregister.gov/d/2014-25439

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

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

[FIDO2] Bradley J, Hodges J, Jones MB, Kumar A, Lindemann R, Verrept J (2022) Client to Authenticator Protocol (CTAP). (FIDO Alliance, Beaverton, OR). Available at https://fidoalliance.org/specs/fido-v2.1-ps-20210615/fido-client-to-authenticator-protocol-v2.1-ps-errata-20220621.html

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

[FIPS201] National Institute of Standards and Technology (2022) Personal Identity Verification (PIV) of Federal Employees and Contractors. (U.S. Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) 201-3. https://doi.org/10.6028/NIST.FIPS.201-3

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

[ISO/IEC2382-37] International Standards Organization (2022) *Information technology — Vocabulary — Part 37: Biometrics* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/73514.html

[ISO/IEC10646] International Standards Organization (2020) *Information technology — Universal coded character set (UCS)* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/76835.html

[ISO/IEC19795-1] International Standards Organization (2021) *Information technology - Biometric performance testing and reporting Part 1: Principles and framework* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/73515.html

[ISO/IEC30107-1] International Standards Organization (2023) *Information technology — Biometric presentation attack detection — Part 1: Framework* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/83828.html

[ISO/IEC30107-3] International Standards Organization (2023) *Information technology — Biometric presentation attack detection — Part 3: Testing and reporting* (ISO, Geneva, Switzerland). Available at https://www.iso.org/standard/79520.html

[Managers] Lyastani SG, Schilling M, Fahl S, Backes M, Bugiel S (2018) Better managed than memorized? Studying the Impact of Managers on Password Strength and Reuse. *27th USENIX Security Symposium (USENIX Security 18)* (USENIX Association, Baltimore, MD), pp 203–220. Available at https://www.usenix.org/conference/usenixsecurity18/presentation/lyastani

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

[OWASP-session] Open Web Application Security Project (2021) *Session Management Cheat Sheet*. Available at https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html

[OWASP-XSS-prevention] Open Web Application Security Project (2021) *XSS (Cross Site Scripting) Prevention Cheat Sheet*. Available at https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html

[Persistence] Herley C, Van Oorschot P (2012) A Research Agenda Acknowledging the Persistence of Passwords. *IEEE Security & Privacy Magazine*, (IEEE, Garden Grove, CA) 10(1):28–36. https://doi.org/10.1109/MSP.2011.150

[Policies] Weir M, Aggarwal S, Collins M, Stern H (2010) Testing Metrics for Password Creation Policies by Attacking Large Sets of Revealed Passwords. *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS '10*, (ACM, New York, NY, USA), pp 162–175. https://doi.org/10.1145/1866307.1866327

[PrivacyAct] Privacy Act of 1974, Pub. L. 93-579, 5 U.S.C. § 552a, 88 Stat. 1896 (1974). Available at https://www.govinfo.gov/content/pkg/USCODE-2020-title5/pdf/USCODE-2020-title5-partI-chap5-subchapII-sec552a.pdf

[PSL] Mozilla Foundation (2022) *Public Suffix List*. Available at https://publicsuffix.org/list/

[RBG] National Institute of Standards and Technology (2023) *Random Bit Generation*. Available at https://csrc.nist.gov/projects/random-bit-generation

[RFC20] Cerf V (1969) ASCII format for network interchange. (Internet Engineering Task Force (IETF)), IETF Request for Comments (RFC) 20. https://doi.org/10.17487/RFC0020

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

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

[RFC6749] Hardt D (2012) The OAuth 2.0 Authorization Framework. (Internet Engineering Task Force (IETF)), IETF Request for Comments (RFC) 6749. https://doi.org/10.17487/RFC6749

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

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

[Shannon] Shannon CE (1948) A Mathematical Theory of Communication. *Bell System Technical Journal* 27(3):379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

[SP800-39] Joint Task Force (2011) Managing Information Security Risk. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-39. https://doi.org/10.6028/NIST.SP.800-39

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

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

[SP800-57Part1] Barker EB (2020) Recommendation for Key Management: Part 1 – General. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-57 Part 1, Rev. 5. https://doi.org/10.6028/NIST.SP.800-57pt1r5

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

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

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

[SP800-73] Cooper C, Ferraiolo H, Mehta K, Francomacaro S, Chandramouli R, Mohler J (2015) Interfaces for Personal Identity Verification. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP)800-73-4, Includes updates as of February 8, 2016. https://doi.org/10.6028/NIST.SP.800-73-4

[SP800-90A] Barker E, Kelsey J (2015) Recommendation for Random Number Generation Using Deterministic Random Bit Generators. (National Institute of Standards and

Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-90A, Rev. 1. https://doi.org/10.6028/NIST.SP.800-90Ar1

[SP800-90B] Turan MS, Barker E, Kelsey J, McKay K, Baish M, Boyle M (2018) Recommendation for the Entropy Sources Used for Random Bit Generation. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-90B. https://doi.org/10.6028/NIST.SP.800-90B

[SP800-90C] Barker E, Kelsey J, McKay K, Roginsky A, Turan MS (2022) Recommendation for Random Bit Generator (RBG) Constructions. (National Institute of Standards and Technology, Gaithersburg, MD), Draft NIST Special Publication (SP) 800-90C. https://doi.org/10.6028/NIST.SP.800-90C.3pd

[SP800-116] Ferraiolo H, Mehta KL, Ghadiali N, Mohler J, Johnson V, Brady S (2018) A Recommendation for the Use of PIV Credentials in Physical Access Control Systems (PACS). (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-116, Rev. 1 [or as amended]. https://doi.org/10.6028/NIST.SP.800-116r1

[SP800-131A] Barker E, Roginsky A (2019) Transitioning the Use of Cryptographic Algorithms and Key Lengths. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-131A Rev. 2. https://doi.org/10.6028/NIST.SP.800-131Ar2

[SP800-132] Turan M, Barker E, Burr W, Chen L (2010) Recommendation for Password-Based Key Derivation. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) 800-132. https://doi.org/10.6028/NIST.SP.800-132

[SP800-157] Ferraiolo H, Regenscheid AR, Fenton J (2023) Guidelines for Derived Personal Identity Verification (PIV) Credentials. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Special Publication (SP) NIST SP 800-157r1 ipd (initial public draft). https://doi.org/10.6028/NIST.SP.800-157r1.ipd

[Strength] Kelley PG, Komanduri S, Mazurek ML, Shay R, Vidas T, Bauer L, Christin N, Cranor LF, Lopez J (2012) Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. *2012 IEEE Symposium On Security and Privacy (SP)*, pp 523–537. Available at http://ieeexplore.ieee.org/iel5/6233637/6234400/06234434.pdf

[TLS] Rescorla E (2018) The Transport Layer Security (TLS) Protocol Version 1.3. (Internet Engineering Task Force, Reston, VA), RFC 8446. https://doi.org/10.17487/RFC8446

[TOTP] M'Raihi D, Machani S, Pei M, Rydell J (2011) TOTP: Time-Based One-Time Password Algorithm. (Internet Engineering Task Force, Reston, VA), RFC 6238. https://doi.org/10.17487/RFC6238

[UsabilityBiometrics] National Institute of Standards and Technology (2008) Usability & Biometrics: Ensuring Successful Biometric Systems. (National Institute of Standards and Technology, Gaithersburg, MD). Available at https://www.nist.gov/system/files/usability_and_biometrics_final2.pdf

[UAX15] Whistler K (2022) Unicode Normalization Forms. (The Unicode Consortium, South San Francisco, CA), Unicode Standard Annex 15, Version 15.0.0, Rev. 53. Available at https://www.unicode.org/reports/tr15/

[WebAuthn] Hodges J, Jones JC, Jones MB, Kumar A, Lundberg E (2021) Web Authentication: An API for accessing Public Key Credentials - Level 2. (World Wide Web Consortium, Cambridge, MA). Available at https://www.w3.org/TR/2021/REC-webauthn-2-20210408/

Appendix A. Strength of Passwords

This appendix is informative.

This appendix uses the word “password” for ease of discussion. Where used, it should be interpreted to include passphrases, PINs, and passwords.

A.1. Introduction

Passwords are a widely used form of authentication despite concerns about their use from both a usability and security standpoint [Persistence]. Humans have a limited ability to memorize complex, arbitrary secrets, so they often choose passwords that can be easily guessed. To address the resultant security concerns, online services have introduced rules to increase the complexity of these passwords. The most notable form is composition rules, which require users to choose passwords that are constructed using a mix of character types (e.g., at least one digit, uppercase letter, and symbol). However, analyses of breached password databases reveal that the benefit of such rules is less significant than initially thought [Policies], and the impacts on usability and memorability are severe.

The complexity of user-chosen passwords has often been characterized using the information theory concept of entropy [Shannon]. While entropy can be readily calculated for data with deterministic distribution functions, estimating the entropy for user-chosen passwords is challenging, and past efforts to do so have not been particularly accurate. For this reason, a different and somewhat more straightforward approach based primarily on password length is presented herein.

Many attacks associated with password use are not affected by password complexity and length. Keystroke logging, phishing, and social engineering attacks are equally effective on lengthy and complex passwords as they are on simple ones. These attacks are outside of the scope of this Appendix.

A.2. Length

Password length is a primary factor in characterizing password strength [Strength] [Composition]. Passwords that are too short yield to brute-force attacks and dictionary attacks. The minimum password length required depends on the threat model being addressed. Online attacks in which the attacker attempts to log in by guessing the password can be mitigated by limiting the permitted login attempt rate. To prevent an attacker (or a persistent claimant with poor typing skills) from quickly inflicting a denial-of-service attack on the subscriber by making many incorrect guesses, passwords need to be complex enough that a reasonable number of attempts can be permitted with a low probability of a successful guess, and rate limiting can be applied before there is a significant chance of a successful guess.

Offline attacks are sometimes possible when the attacker obtains one or more hashed passwords through a database breach. The ability of the attacker to determine one or more users' passwords depends on how the password is stored. Commonly, passwords are salted with a random value and hashed, preferably using a computationally expensive algorithm. Even with such measures, the current ability of attackers to compute many billions of hashes per second in an offline environment that is not subject to rate limiting requires passwords to be orders of magnitude more complex than those expected to resist only online attacks.

Users should be encouraged to make their passwords as lengthy as they want, within reason. Since the size of a hashed password is independent of its length, there is no reason to prohibit the use of lengthy passwords (or passphrases) if the user wishes. Extremely long passwords (perhaps megabytes long) could require excessive processing time to hash, so it is reasonable to have some limit.

A.3. Complexity

Composition rules are commonly used in an attempt to increase the difficulty of guessing user-chosen passwords. However, research has shown that users respond in very predictable ways to the requirements imposed by composition rules [Policies]. For example, a user who might have chosen "password" as their password would be relatively likely to choose "Password1" if required to include an uppercase letter and a number or "Password1!" if a symbol is also required.

Users also express frustration when online services reject their attempts to create complex passwords. Many services reject passwords with spaces and various special characters. Characters that are not accepted are sometimes the result of an effort to avoid attacks that depend on those characters (e.g., SQL injection). However, an unhashed password would not be sent intact to a database, so such precautions are unnecessary. Users should also be able to include space characters to allow the use of phrases. Space characters add little to the complexity of passwords and may introduce usability issues (e.g., the undetected use of two spaces rather than one), so removing repeated spaces in typed passwords may be beneficial if initial verification fails.

Since users' password choices are often predictable, so attackers are likely to guess passwords that have previously proven successful. These include dictionary words and passwords from previous breaches, such as the "Password1!" example above. For this reason, passwords chosen by users should be compared against a blocklist of unacceptable passwords. This list should include passwords from previous breach corpuses, dictionary words used as passwords, and specific words (e.g., the name of the service itself) that users are likely to choose. Since a minimum length requirement will also govern the user's choice of passwords, this dictionary only needs to include entries that meet that requirement. As noted in Sec. 3.1.1.2, it is not beneficial for the blocklist to be excessively large or comprehensive, since its primary purpose is to prevent the use of very common passwords that might be guessed in an online attack before throttling restrictions take effect. An excessively large blocklist will likely frustrate users who attempt to choose a memorable password.

Highly complex passwords introduce a new potential vulnerability: they are less likely to be memorable and more likely to be written down or stored electronically in an unsafe manner. While these practices are not necessarily vulnerable, some methods of recording such secrets will be. This is an additional motivation for not requiring excessively long or complex passwords.

A.4. Central vs. Local Verification

While passwords that are used as a separate authentication factor are often centrally verified by the CSP's verifier, those that are used as an activation factor for a multi-factor authenticator are either verified locally by the authenticator or used to derive the authenticator output, which will be incorrect if the wrong activation factor is used. Both of these situations are referred to as "local verification."

The attack surfaces and vulnerabilities for central and local verification are very different. Accordingly, the requirements for centrally verified passwords differ from those verified locally. Centrally verified passwords require the verifier (i.e., an online resource) to store salted and iteratively hashed verification secrets for all of the subscribers' passwords. Although the salting and hashing process increases the computational effort to determine the passwords from the hashes, the verifier is an attractive target for attackers, particularly those interested in compromising an arbitrary subscriber rather than a specific one.

Local verifiers do not have the same concerns with large-scale attacks on a central online verifier but depend to a greater extent on the physical security of the authenticator and the integrity of its associated endpoint. To the extent that the authenticator stores the activation factor, that factor must be protected against physical and side-channel (e.g., power and timing analysis) attacks on the authenticator. When the activation factor is entered through the associated endpoint, the endpoint needs to be free of malware (e.g., key-logging software). Since such threats are less dependent on the length and complexity of the password, these requirements are relaxed for local verification.

Online password-guessing attacks are a similar threat to centrally and locally verified passwords. Throttling, which is the primary defense against online attacks, can be particularly challenging for local verifiers because of the limited ability of some authenticators to securely store information about unsuccessful attempts. Throttling can be performed by either keeping a count of invalid attempts in the authenticator or generating an authenticator output rejected by the CSP verifier, which does the throttling. In this case, the invalid outputs must not be evident to the attacker, who could otherwise make offline attempts until a valid-looking authenticator output appears.

A.5. Summary

Length and complexity requirements beyond those recommended here significantly increase the difficulty of using passwords and increase user frustration. As a result, users often work around these restrictions counterproductively. Other mitigations (e.g., blocklists, secure hashed storage, machine-generated random passwords, and rate limiting) are more effective at preventing modern brute-force attacks, so no additional complexity requirements are imposed.

Appendix B. Syncable Authenticators

This appendix is normative.

B.1. Introduction

The ability to “sync” authenticators — specifically to copy (i.e., clone) their authentication secrets to the cloud and thence to additional authenticators — is a relatively new development in authentication. This appendix provides additional guidelines on the use of syncable authenticators.

B.2. Cloning of Authentication Keys

In some cases, the secret keys associated with multi-factor cryptographic authenticators (e.g., those based on the WebAuthn standard [WebAuthn]) may be stored in a sync fabric. This allows the keys to be backed up and transferred to other devices. The following requirements apply to keys managed in this manner:

- All keys **SHALL** be generated using approved cryptography.
- Private keys that are cloned or exported from a device **SHALL** only be stored in an encrypted form.
- All authentication transactions **SHALL** perform private-key operations on the local device using cryptographic keys that are generated on-device or recovered from the sync fabric (e.g., in cloud storage).
- Private keys stored in cloud-based accounts **SHALL** be protected by access control mechanisms such that only the authenticated user can access their private keys in the sync fabric.
- User access to private keys in the sync fabric **SHALL** be protected by AAL2-equivalent MFA to preserve the integrity of the authentication protocols using the synced keys.
- These general requirements and any other agency-specific requirements for using syncable authenticators **SHALL** be documented and communicated, including on public-facing websites and digital service policies, where applicable.

Additional requirements for federal enterprise[6] use of syncable authenticators:

- Federal enterprise private keys (i.e., federal keys) **SHALL** be stored in sync fabrics that have achieved FISMA Moderate protections or equivalent.

[6] With respect to these requirements, federal enterprise systems and keys include what would be considered in scope for PIV guidance, such as government contractors, government employees, and mission partners. It does not include government-to-consumer or public-facing use cases.

- Devices (e.g., mobile phones, laptops, tablets) that generate, store, and sync authenticators containing federal enterprise private keys **SHALL** be protected by mobile device management software or other device configuration controls that prevent the syncing or sharing of keys to unauthorized devices or sync fabrics.
- Access to the sync fabric **SHALL** be controlled by agency-managed accounts (e.g., a central identity and access management solution or platform-based managed account) to maintain enterprise control over the private key's life cycle.
- Authenticators that generate private keys **SHOULD** support attestation features that can be used to verify the capabilities and sources of the authenticator (e.g., enterprise attestation).

These controls specifically support syncing and should be considered additive to the existing multi-factor cryptographic authenticator requirements and AAL2 requirements, including [FIPS140] validation.

> Syncing authentication keys inherently means that the key can be exported. Authentication at AAL2 may be supported subject to the above requirements. However, syncing violates the non-exportability requirements of AAL3. Similar protocols using keys not stored in an exportable manner that meet the other requirements of AAL3 may be used.

B.3. Implementation Requirements

Many syncable authenticators are built upon W3C's [WebAuthn] specification, which provides a common data structure, a challenge-response cryptographic protocol, and an API for leveraging public-key credentials. The specification is flexible and adaptive, meaning that not all deployments of WebAuthn credentials will meet the requirements of federal agencies for implementation.

The specification has a series of flags that the RP application can request from the authenticator to provide additional context for the authentication event and determine whether it meets the RP's access policies. This section describes certain flags in the WebAuthn specification that federal agencies acting as RPs should understand and interrogate when building their syncable authenticator implementations to align with NIST AAL2 guidelines.

The following requirements apply to WebAuthn Level 3 flags:

User Present (UP)

The User Present flag indicates that a "presence" test was used to confirm that the user has interacted with the authenticator (e.g., tapping a hardware token inserted into a USB port). This supports authentication intent, as described in Sec. 3.2.8. Verifiers **SHOULD** confirm that the User Present flag has been set.

User Verified (UV)

The User Verified flag indicates that the authenticator has locally authenticated the user using one of the available "user verification" methods. Verifiers **SHALL** indicate that UV is preferred and **SHALL** inspect responses to confirm the value of the UV flag. This indicates whether the authenticator can be treated as a multi-factor cryptographic authenticator. If the user is not verified, agencies **SHALL** treat the authenticator as a single-factor cryptographic authenticator. A further extension to the WebAuthn Level 3 specification (see Sec. 10.3 of [WebAuthn]) provides additional data on verification methods if agencies seek to gain context on the local authentication event.

Backup Eligible

The Backup Eligible flag indicates whether the authenticator can be synced to a different device (i.e., whether the key can be stored elsewhere). It is important to note that just because an authenticator *can* be synced does not mean that it *has* been synced. Verifiers **MAY** use this flag to establish policies that restrict the use of syncable authenticators. This flag is necessary to distinguish authenticators that are device-bound from those that may be cloned to more than one device.

Backup State

The Backup State flag indicates whether an authenticator *has* been synced to a different device. Verifiers **MAY** use this flag to establish restrictions on authenticators that are synced to other devices. Agencies **SHOULD NOT** condition acceptance based on this flag for public-facing applications due to user experience concerns. This flag **MAY** be used for enterprise applications to support the restriction of syncable authenticators for specific applications.

In addition to the flags specified above, agencies may wish to gain additional information on the origins and capabilities of the syncable authenticators that they choose to implement and accept. Within the context of FIDO2 WebAuthn, some authenticators support attestation features that can be used to determine the capabilities and manufacturers of specific authenticators. For enterprise use cases, agencies **SHOULD** implement attestation capabilities based on the functionality offered by their platform providers. This would take the form of an enterprise attestation in which the RP requests identifying information about the authenticator.

Attestations **SHOULD NOT** block the use of syncable authenticators for broad public-facing applications. Due to their limited availability in consumer products, requiring their use is likely to divert users to less secure authentication options that are vulnerable to phishing (e.g., PSTN-based out-of-band authentication). While authentication transaction metadata, such as the User Verified flag indicating the use of a local activation factor, is available in WebAuthn responses, attestation can provide stronger assurance of the characteristics of the authenticator used in a transaction. RPs **MAY** use attestation to determine the level of confidence they have in a syncable authenticator.

Even if the RP requests flag and attestation data, the authenticator may not return all of the requested information, or it may return information that is inconsistent with the expected response mandated for access to a resource. Agencies **SHALL** evaluate the use cases for syncable authenticators and determine the appropriate access policy decisions that they intend to make based on the returned information.

B.4. Sharing

Cybersecurity guidelines have historically cautioned against sharing authenticators between users, expecting different users to maintain their own unique authenticators. Despite this guidance, authenticator and password sharing occurs within some user groups and applications to allow individuals to share access to a digital account.

As indicated in Table 5, some syncable authenticator implementations have embraced this user behavior and established methods for sharing authentication keys between different users. Further, some implementations actively encourage sharing syncable authenticators as a convenient and more secure alternative to sharing passwords for common services.

For enterprise use cases, concerns over sharing keys can be effectively mitigated using device management techniques that limit the ability for keys to be moved off of approved devices or sync fabrics. However, similar mitigations are not currently available for public-facing use cases, leaving RPs dependent on the sharing models adopted by syncable authenticator providers. Owners of public-facing applications should be aware of the risks associated with shared authenticators. When interacting with the public, agencies have limited visibility into which specific authenticators are being employed by their users and should assume that all syncable authenticators may be subject to sharing. While many sharing models have substantial controls that minimize risks (e.g., requiring close proximity between devices to allow sharing), other implementations are less restrictive.

The risk of sharing posed by this new class of authenticators is not unique. It applies to all authenticator types, some of which are weaker than syncable authenticators. Any authenticator can be shared by a user who is determined to share it. Users can actively share passwords, OTPs, out-of-band authenticators, and even push authentication events that allow a designee (whether formal or not) to authenticate on behalf of an end user.

Agencies determine which authenticators they will accept for their applications based on the specific risks, threats, and usability considerations they face. Syncable authenticators may be offered as a new option for applications that seek to implement up to AAL2. The trade-offs of this technology should be well-balanced based on their expected outcomes for security, privacy, equity, and usability.

B.5. Example

A common use of syncable authenticators is in an AAL2 authentication transaction. The following items summarize how WebAuthn syncable authenticators satisfy various aspects of AAL2 requirements:

Phishing resistance (recommended; required for federal enterprise)

Achieved: Properly configured syncable authenticators create a unique public or private key pair whose use is constrained to the domain in which it was created (i.e., the key can only be used with a specific website or RP). This prevents a falsified web page from being able to capture and reuse an authenticator output.

Replay resistance (required)

Achieved: Syncable authenticators prevent replay resistance (i.e., prevention of reuse in future transactions) through a random nonce that is incorporated into each authentication transaction.

Authentication intent (required)

Achieved: Syncable authenticators require users to input an activation secret to initiate the cryptographic authentication protocol. This serves as authentication intent, as the event cannot proceed without the user's active participation.

Multi-factor (required)

Achieved: The user verified (UV) flag value indicates whether a local authentication mechanism (i.e., an activation factor) was used to complete the transaction. Without user verification, the verifier prompts for an additional authentication factor as part of the transaction.

B.6. Security Considerations

Syncable authenticators present distinct threats and challenges that agencies should evaluate before implementation or deployment, as shown in Table 4.

Table 4. Syncable Authenticator Threats, Challenges, and Mitigations

Threat or Challenge	**Description**	**Mitigations**
Unauthorized key use or loss of control	Some syncable authenticator deployments support sharing private keys to devices that belong to other users who can then misuse the key	Enforce enterprise device management features or managed profiles that prevent synced keys from being shared.
		Notify users of key-sharing events through all available notification channels.
		Provide mechanisms for users to view keys, key statuses, and whether/where keys have been shared.
		Educate users about the risks of unauthorized key use through existing awareness and training mechanisms.
Sync fabric compromise	To support key syncing, most implementations clone keys to a sync fabric (i.e., a cloud-based service connected to multiple devices associated with an account).	Store only encrypted key material.
		Implement syncing fabric access controls that prevent anyone other than the authenticated user from accessing the private key.
		Evaluate cloud services for baseline security features (e.g., FISMA Moderate protections or comparable).
		Leverage hardware security modules to protect encrypted keys.

Unauthorized access to sync fabric and recovery	Synced keys are accessible via cloud-based account recovery processes, which represent a potential weakness to the authenticators.	Implement authentication recovery processes that are consistent with SP 800-63B.
		Restrict recovery capabilities for federal enterprise keys through device management or managed account capabilities.
		Bind multiple authenticators at AAL2 and above to support recovery.
		Require AAL2 authentication to add any new authenticators for user access to the sync fabric.
		Use only as a derived authenticator in federal enterprise scenarios [SP800-157].
		Notify the user of any recovery activities.
		Leverage a user-controlled secret (i.e., something not known to the sync fabric provider) to encrypt and recover keys.
Revocation	Since syncable authenticators use RP-specific keys, the ability to centrally revoke access based on those keys is challenging. For example, with traditional PKI, CRLs can be used centrally to revoke access. A similar process is not available for syncable authenticators (or any FIDO WebAuthn-based credentials).	Implement a central identity management (IDM) account for users to manage authenticators and remove them from the “home agency” account if they are compromised or expired.

		Leverage SSO and federation to limit the number of RP-specific keys that will need to be revoked in an incident.
		Establish policies and tools to request that users periodically review keys for validity and currency.

Appendix C. List of Symbols, Abbreviations, and Acronyms

AAL
Authentication Assurance Level

CSP
Credential Service Provider

CSRF
Cross-Site Request Forgery

XSS
Cross-Site Scripting

DNS
Domain Name System

FEDRAMP
Federal Risk and Authorization Management Program

FMR
False Match Rate

FNMR
False Non-Match Rate

IAL
Identity Assurance Level

IdP
Identity Provider

KBA
Knowledge-Based Authentication

MAC
Message Authentication Code

NARA
National Archives and Records Administration

OTP
One-Time Password

PAD
Presentation Attack Detection

PIA
Privacy Impact Assessment

PII
Personally Identifiable Information

PIN
Personal Identification Number

PKI
Public Key Infrastructure

PSTN
Public Switched Telephone Network

RP
Relying Party

SAOP
Senior Agency Official for Privacy

SSL
Secure Sockets Layer

SMS
Short Message Service

SORN
System of Records Notice

TEE
Trusted Execution Environment

TLS
Transport Layer Security

TPM
Trusted Platform Module

VOIP
Voice-Over-IP

XSS
Cross-Site Scripting

Appendix D. Glossary

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

account recovery
The ability to regain ownership of a *subscriber account* and its associated information and privileges.

activation
The process of inputting an *activation factor* into a *multi-factor authenticator* to enable its use for *authentication*.

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

activation secret
A *password* that is used locally as an *activation factor* for a *multi-factor authenticator*.

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

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

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

attestation
Information conveyed to the *CSP*, generally at the time that an *authenticator* is bound, describing the characteristics of a connected authenticator or the *endpoint* involved in an authentication operation.

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

authenticate
See *authentication*.

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

authenticated session
See *protected session*.

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

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

authentication factor
The three types of authentication factors are *something you know*, *something you have*, and *something you are*. Every *authenticator* has one or more authentication factors.

authentication intent
The process of confirming the *claimant's* intent to *authenticate* or reauthenticate by requiring user intervention in the authentication flow. Some *authenticators* (e.g., OTPs) establish authentication intent as part of their operation. Others require a specific step, such as pressing a button, to establish intent. Authentication intent is a countermeasure against use by malware at the *endpoint* as a proxy for authenticating an attacker without the subscriber's knowledge.

authentication protocol
A defined sequence of messages between a *claimant* and a *verifier* that demonstrates that the claimant has possession and control of one or more valid *authenticators* to establish their identity, and, optionally, demonstrates that the claimant is communicating with the intended verifier.

authentication secret
A generic term for any secret value that an attacker could use to impersonate the subscriber in an *authentication protocol*.

These are further divided into *short-term authentication secrets*, which are only useful to an attacker for a limited period of time, and *long-term authentication secrets*, which allow an attacker to impersonate the subscriber until they are manually reset. The *authenticator* secret is the canonical example of a long-term authentication secret, while the *authenticator output* — if it is different from the *authenticator secret* — is usually a short-term authentication secret.

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

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

authenticator output
The output value generated by an *authenticator*. The ability to generate valid authenticator outputs on demand proves that the *claimant* possesses and controls the authenticator. Protocol messages sent to the *verifier* depend on the authenticator output, but they may or may not explicitly contain it.

authenticator secret
The secret value contained within an *authenticator*.

authenticator type
A category of *authenticators* with common characteristics, such as the types of *authentication factors* they provide and the mechanisms by which they operate.

authenticity
The property that data originated from its purported source.

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

biometric sample
An analog or digital representation of biometric characteristics prior to biometric feature extraction, such as a record that contains a fingerprint image.

biometrics
Automated recognition of individuals based on their biological or behavioral characteristics. Biological characteristics include but are not limited to fingerprints, palm prints, facial features, iris and retina patterns, voiceprints, and vein patterns. Behavioral characteristics include but are not limited to keystrokes, angle of holding a smart phone, screen pressure, typing speed, mouse or mobile phone movements, and gyroscope position.

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

claimant
A *subject* whose identity is to be verified using one or more *authentication protocols*.

credential
An object or data structure that authoritatively binds an identity — via an *identifier* — and (optionally) additional *attributes*, to at least one *authenticator* possessed and controlled by a subscriber.

A credential is issued, stored, and maintained by the *CSP*. Copies of information from the credential can be possessed by the subscriber, typically in the form of one or more digital certificates that are often contained in an authenticator along with their associated *private keys*.

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

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

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

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

cryptographic authenticator
An *authenticator* that proves possession of an *authentication secret* through direct communication with a *verifier* through a cryptographic *authentication protocol*.

cryptographic key
A value used to control cryptographic operations, such as decryption, encryption, signature generation, or signature *verification*. For the purposes of these guidelines, key requirements shall meet the minimum requirements stated in Table 2 of [SP800-57Part1]. See *asymmetric keys* or *symmetric keys*.

cryptographic module
A set of hardware, software, or firmware that implements approved security functions including cryptographic algorithms and key generation.

digital authentication
The process of establishing confidence in user identities that are digitally presented to a system. In previous editions of SP 800-63, this was referred to as electronic authentication.

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

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

digital transaction
A discrete digital event between a user and a system that supports a business or programmatic purpose.

electronic authentication (e-authentication)
See *digital authentication*.

endpoint
Any device that is used to access a *digital identity* on a *network*, such as laptops, desktops, mobile phones, tablets, servers, Internet of Things devices, and virtual environments.

enrollment
The process through which a *CSP/IdP* provides a successfully identity-proofed *applicant* with a *subscriber account* and binds *authenticators* to grant persistent access.

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

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

factor
See *authentication factor*

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

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

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

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

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

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

identity
See *digital identity*

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

identity proofing
The processes used to collect, validate, and verify information about a *subject* in order to establish assurance in the subject's *claimed identity*.

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

identity resolution
The process of collecting information about an *applicant* to uniquely distinguish an individual within the context of the population that the *CSP* serves.

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

manageability
Providing the capability for the granular administration of *personally identifiable information*, including alteration, deletion, and selective disclosure. [NISTIR8062]

memorized secret
See *password*.

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

mobile code
Executable code that is normally transferred from its source to another computer system for execution. This transfer is often through the *network* (e.g., JavaScript embedded in a web page) but may transfer through physical media as well.

multi-factor authentication (MFA)
An authentication system that requires more than one distinct type of *authentication factor* for successful authentication. MFA can be performed using a *multi-factor authenticator* or by combining *single-factor* authenticators that provide different types of factors.

multi-factor authenticator
An *authenticator* that provides more than one distinct *authentication factor*, such as a cryptographic authentication device with an integrated biometric sensor that is required to activate the device.

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

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

non-repudiation
The capability to protect against an individual falsely denying having performed a particular transaction.

offline attack
An attack in which the attacker obtains some data (typically by eavesdropping on an authentication transaction or by penetrating a system and stealing security files) that the attacker is able to analyze in a system of their own choosing.

online attack
An attack against an *authentication protocol* in which the attacker either assumes the role of a *claimant* with a genuine *verifier* or actively alters the authentication channel.

online guessing attack
An attack in which an attacker performs repeated logon trials by guessing possible values of the *authenticator* output.

passphrase
A *password* that consists of a sequence of words or other text that a *claimant* uses to *authenticate* their identity. A passphrase is similar to a password in usage but is generally longer for added security.

password
A type of *authenticator* consisting of a character string that is intended to be memorized or memorable by the subscriber to permit the *claimant* to demonstrate *something they know* as part of an authentication process. Passwords are referred to as *memorized secrets* in the initial release of SP 800-63B.

personal identification number (PIN)
A *password* that typically consists of only decimal digits.

personal information
See *personally identifiable information*.

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

pharming
An attack in which an attacker corrupts an infrastructure service such as DNS (e.g., Domain Name System [DNS]) and causes the subscriber to be misdirected to a forged *verifier/RP*, which could cause the subscriber to reveal sensitive information, download harmful software, or contribute to a fraudulent act.

phishing
An attack in which the subscriber is lured (usually through an email) to interact with a counterfeit *verifier/RP* and tricked into revealing information that can be used to masquerade as that subscriber to the real verifier/RP.

phishing resistance
The ability of the *authentication protocol* to prevent the disclosure of *authentication secrets* and valid *authenticator* outputs to an impostor *verifier* without reliance on the vigilance of the *claimant*.

physical authenticator
An *authenticator* that the *claimant* proves possession of as part of an authentication process.

possession and control of an authenticator
The ability to activate and use the *authenticator* in an *authentication protocol*.

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

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

presentation attack
Presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system.

presentation attack detection (PAD)
Automated determination of a *presentation attack*. A subset of presentation attack determination methods, referred to as *liveness detection*, involves the measurement and analysis of anatomical characteristics or voluntary or involuntary reactions, to determine if a *biometric sample* is being captured from a living *subject* that is present at the point of capture.

Privacy Impact Assessment (PIA)
A method of analyzing how *personally identifiable information* (PII) is collected, used, shared, and maintained. PIAs are used to identify and mitigate privacy risks throughout the development lifecycle of a program or system. They also help ensure that handling information conforms to legal, regulatory, and policy requirements regarding privacy.

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

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

pseudonym
A name other than a legal name.

pseudonymity
The use of a *pseudonym* to identify a *subject*.

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

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

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

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

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

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

remote
A process or transaction that is conducted through connected devices over a *network*, rather than in person.

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

replay resistance
The property of an authentication process to resist *replay attacks*, typically by the use of an *authenticator* output that is valid only for a specific authentication.

restricted
An *authenticator* type, class, or instantiation that has additional risk of false acceptance associated with its use and is therefore subject to additional requirements.

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

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

salt
A non-secret value used in a cryptographic process, usually to ensure that the results of computations for one instance cannot be reused by an attacker.

Secure Sockets Layer (SSL)
See *Transport Layer Security (TLS)*.

Senior Agency Official for Privacy (SAOP)
Person responsible for ensuring that an agency complies with privacy requirements and manages privacy risks. The SAOP is also responsible for ensuring that the agency considers the privacy impacts of all agency actions and policies that involve PII.

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

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

shared secret
A secret used in authentication that is known to the subscriber and the verifier.

side-channel attack
An attack enabled by the leakage of information from a physical cryptosystem. Characteristics that could be exploited in a side-channel attack include timing, power consumption, and electromagnetic and acoustic emissions.

single-factor
A characteristic of an authentication system or an *authenticator* that requires only one *authentication factor* (i.e., something you know, something you have, or something you are) for successful authentication.

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

social engineering
The act of deceiving an individual into revealing sensitive information, obtaining unauthorized access, or committing fraud by associating with the individual to gain confidence and trust.

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

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

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

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

sync fabric
Any on-premises, cloud-based, or hybrid service used to store, transmit, or manage authentication keys generated by syncable *authenticators* that are not local to the user's device.

syncable authenticators
Software or hardware cryptographic *authenticators* that allow authentication keys to be cloned and exported to other storage to sync those keys to other authenticators (i.e., devices).

system of record (SOR)
An SOR is a collection of records that contain information about individuals and are under the control of an agency. The records can be retrieved by the individual's name or by an identifying number, symbol, or other *identifier*.

System of Record Notice (SORN)
A notice that federal agencies publish in the Federal Register to describe their systems of records.

token
See *authenticator*.

transaction
See *digital transaction*

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

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

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

verifier impersonation
See *phishing*.

zeroize
Overwrite a memory location with data that consists entirely of bits with the value zero so that the data is destroyed and unrecoverable. This is often contrasted with deletion methods that merely destroy references to data within a file system rather than the data itself.

Appendix E. Change Log

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

- Throughout: Removed Purpose and Definitions and Abbreviations numbered sections and renumbered sections accordingly. Section numbers referenced below are the new section numbers.
- Throughout: Changed the name of *memorized secrets* to *passwords*.
- Section 3.1.3: Disallowed the comparison of secrets from primary and secondary channel for out-of-band authentication.
- Section 3.1.3.1: Removed the prohibition on the use of VoIP phone numbers for out-of-band authentication.
- Section 3.1.3.4: Recognized multi-factor out-of-band authenticators that require an activation factor.
- Section 3.1.4 and Sec. 3.1.5: Removed "devices" from the authenticator name to recognize OTP applications.
- Section 3.1.6 and Sec. 3.1.7: Removed "software" and "device" distinction from the authenticator name; these are now authenticator characteristics.
- Section 3.1.7.4 and Appendix B : Added requirements for syncable authenticators.
- Section 3.2.3: Updated biometric performance requirements and metrics and included a discussion of equity impacts.
- Section 3.2.5: Added a definition and updated requirements for phishing-resistant authenticators.
- Section 3.2.10: Established separate requirements for locally verified memorized secrets known as *activation secrets*.
- Section 3.2.11: Added requirements for authenticators that are connected via wireless technologies such as NFC and Bluetooth.
- Section 3.2.12: Centralized the requirements for random values used throughout the document.
- Section 3.2.13: Added a new section on requirements for the non-exportability of authenticator secrets.
- Removed verifier compromise resistance as a distinct named requirement because it is generally a characteristic of the chosen authenticator type.
- Section 4: Section renamed "Authenticator Event Management."
- Section 4.1.1: Moved binding at enrollment to SP 800-63A.

- Section 4.1.2.1: Generalized binding an additional authenticator to all AALs.
- Section 4.1.2.2: Added requirements for binding authenticators that are not connected to an endpoint.
- Section 4.2: Revised the requirements and methods for account recovery.
- Section 4.6: Revised the requirements for notifications sent to subscribers.
- Section 5.1.1: Added requirements for browser cookies used for session maintenance.
- Section 5.2: Revised reauthentication requirements to define the overall structure of reauthentication here and specify timeout values in the AAL requirements.
- Section 5.3: Added guidelines for the use of session monitoring (continuous authentication).
- Section 9: Added a section on equity considerations.