Wed, 08 Feb 2017 08:15:57 -0500

DRAFT NIST Special Publication 800-63B

Digital Identity Guidelines

Authentication and Lifecycle Management

Paul A. Grassi
James L. Fenton
Elaine M. Newton
Ray A. Perlner
Andrew R. Regenscheid
William E. Burr
Justin P. Richer

Privacy Authors:

Naomi B. Lefkovitz
Jamie M. Danker

Usability Authors:

Yee-Yin Choong
Kristen K. Greene
Mary F. Theofanos

COMPUTER SECURITY

National Institute of
Standards and Technology
U.S. Department of Commerce

DRAFT NIST Special Publication 800-63B

Digital Identity Guidelines

Authentication and Lifecycle Management

Paul A. Grassi
Applied Cybersecurity Division
Information Technology Laboratory

James L. Fenton
Altmode Networks
Los Altos, CA

Elaine M. Newton
Office of the Director
Information Technology Laboratory

Ray A. Perlner
Computer Security Division
Information Technology Laboratory

Andrew R. Regenscheid
Computer Security Division
Information Technology Laboratory

William E. Burr
Dakota Consulting, Inc.
Silver Spring, MD

Justin P. Richer
Bespoke Engineering
Billerica, MA

Privacy Authors:

Naomi B. Lefkovitz
Applied Cybersecurity Division
Information Technology Laboratory

Jamie M. Danker
National Protection and Programs Directorate
Department of Homeland Security

Usability Authors:

Yee-Yin Choong
Kristen K. Greene
Mary F. Theofanos
Information Access Division
Information Technology Laboratory

Month TBD 2017

National Institute of Standards and Technology

Kent Rochford, Acting Under Secretary of Commerce for Standards and Technology and Director

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. § 3541 et seq., Public Law (P.L.) 113-283. NIST is responsible for developing information security standards and guidelines, including minimum requirements for federal 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.

National Institute of Standards and Technology Special Publication 800-63B
Natl. Inst. Stand. Technol. Spec. Publ. 800-63B, xxx pages (MonthTBD 2017)
CODEN: NSPUE2

> 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 NIST, 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 http://csrc.nist.gov/publications (http://csrc.nist.gov/publications).

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.

Abstract

These guidelines provide technical requirements for Federal agencies implementing digital identity services are not intended to constrain the development or use of standards outside of this purpose. These guidelines focus on the remote authentication of subjects interacting with government systems over open networks, establishing that a given claimant is a subscriber that has 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. This publication supersedes corresponding sections of NIST SP 800-63-1 and SP 800-63-2.

Keywords

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

Acknowledgements

The authors would like to acknowledge the thought leadership and innovation of the original authors: Donna F. Dodson, W. Timothy Polk, Sarbari Gupta, and Emad A. Nabbus. Without their tireless efforts, we would not have had the incredible baseline from which to evolve 800-63 to the document it is today. In addition, special thanks to the Federal Privacy Council's Digital Authentication Task Force for the contributions to the development of privacy requirements and considerations.

Audience

Compliance with NIST Standards and Guidelines

Conformance Testing

Trademark Information

Requirements Notation and Conventions

The terms “SHALL” and “SHALL NOT” indicate requirements to be followed strictly in order to conform to the publication and from which no deviation is permitted.

The terms “SHOULD” and “SHOULD NOT” indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.

The terms “MAY” and “NEED NOT” indicate a course of action permissible within the limits of the publication.

The terms “CAN” and “CANNOT” indicate a possibility and capability, whether material, physical or causal or, in the negative, the absence of that possibility or capability.

Table of Contents

1. Purpose

This section is informative.

This document and its companion documents, [Special Publication (SP) 800-63-3] (sp800-63-3.html), [SP 800-63A] (sp800-63a.html), and [SP 800-63C] (sp800-63c.html), provide technical guidelines to agencies for the implementation of digital authentication.

2. Introduction

This section is informative.

Digital identity is the unique representation of a subject engaged in an online transaction. A digital identity is always unique in the context of a digital service, but does not necessarily need to uniquely identify the subject. In other words, accessing a digital service may not mean that the physical representation of the underlying subject is known. Identity proofing establishes that a subject is actually who they claim to be. Digital authentication is the process of determining the validity of one or more authenticators used to claim a digital identity. Authentication establishes a subject attempting to access a digital service is in control of the technologies used to authenticate. For services in which return visits are applicable, successfully authenticating provides reasonable risk-based assurances that the subject that is accessing the service today is the same as that which accessed the service yesterday. Digital identity presents a technical challenge because this process often involves the proofing of individuals over an open network, and always involves the authentication of individual subjects over an open network to access digital government services. Of which exists multiple opportunities for impersonation and other attacks to fraudulently claim another subject's digital identity.

The ongoing authentication of subscribers is central to this process. Subscriber authentication is performed by verifying that the claimant controls one or more *authenticators* (called *tokens* in earlier editions of SP 800-63) associated with a given subscriber. A successful authentication results in the assertion of an identifier, either pseudonymous or non-pseudonymous, and optionally other identity information, to the relying party (RP).

This document provides recommendations on types of authentication processes, including choices of authenticators, that may be used at various *Authenticator Assurance Levels* (AALs). It also provides recommendations on the lifecycle of authenticators, including revocation in the event of loss or theft.

This technical guideline applies to digital authentication of subjects to systems over a network. It does not primarily address the authentication of a person who is physically present, for example, for access to buildings, although some credentials that are used remotely may also be used in local authentication. This technical guideline also establishes requirements that Federal systems and service providers participating in authentication protocols be authenticated to subscribers.

The strength of an authentication transaction is characterized by an ordinal measurement known as the AAL. Stronger authentication (a higher AAL) requires a higher level of capabilities or resources on the part of an attacker in order to successfully authenticate, effectively reducing the risk of an authentication error. A high-level summary of the technical requirements for each of the AALs is provided below; see Section 4 and 5 of this document for specific normative requirements.

Authenticator Assurance Level 1 - AAL1 provides some assurance that the claimant controls an authenticator registered to the subscriber. AAL1 requires single-factor authentication using a wide range of available authentication technologies. Successful authentication requires that the claimant prove possession and control of the authenticator through a secure authentication protocol.

Authenticator Assurance Level 2 - AAL2 provides high confidence that the claimant controls authenticator(s) registered to the subscriber. Proof of possession and control of two different authentication factors is required through a secure authentication protocol. Approved cryptographic techniques are required at AAL2 and above.

Authenticator Assurance Level 3 - AAL3 provides very high confidence that the claimant controls authenticator(s) registered to the subscriber. Authentication at AAL3 is based on proof of possession of a key through a cryptographic protocol. AAL3 is like AAL2 but also requires a “hard” cryptographic authenticator that provides verifier impersonation resistance.

The following table states which sections of the document are normative and which are informative:

Section Name	Normative/Informative
1. Purpose	Informative
2. Introduction	Informative
3. Definitions and Abbreviations	Informative
4. Authenticator Assurance Levels	Normative
5. Authenticator and Verifier Requirements	Normative
6. Authenticator Lifecycle Management	Normative
7. Session Management	Normative
8. Threat and Security Considerations	Informative
9. Privacy Considerations	Informative
10. Usability Considerations	Informative
11. References	Informative
Appendix A: Strength of Memorized Secrets	Informative

3. Definitions and Abbreviations

This section is informative.

There is a variety of terms used in the area of authentication. While the definitions of many terms are consistent with the original version of SP 800-63, some have changed in this revision. Since there is no single, consistent definition of many of these terms, careful attention to how the terms are defined here is warranted.

The definitions in this section are primarily those that are referenced in this document. Refer to the other documents in the SP 800-63 document family for additional definitions and abbreviations specific to their content.

Active Attack

An attack on the authentication protocol where the attacker transmits data to the claimant, Credential Service Provider (CSP), verifier, or Relying Party (RP). Examples of active attacks include man-in-the-middle (MitM), impersonation, and session hijacking.

Approved

Federal Information Processing Standard (FIPS) approved or NIST recommended. An algorithm or technique that is either 1) specified in a FIPS or NIST Recommendation, or 2) adopted in a FIPS or NIST Recommendation.

Assurance

In the context of [OMB M-04-04] and this document, assurance is defined as 1) the degree of confidence in the vetting process used to establish the identity of an individual to whom the credential was issued, and 2) the degree of confidence that the individual who uses the credential is the individual to whom the credential was issued.

Asymmetric Keys

Two related keys, consisting 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.

Attack

An attempt by an unauthorized individual to defeat security controls. For example, to fool a verifier or an RP into believing that the unauthorized individual in question is the subscriber.

Attacker

A party who acts with malicious intent to compromise an information system.

Attribute

A quality or characteristic ascribed to someone or something.

Authenticated Protected Channel

A communication channel that uses approved encryption where the initiator of the connection (client) has authenticated the recipient (server). Authenticated protected channels provide confidentiality and MitM protection and are frequently used in the user authentication process. Transport Layer Security (TLS) [BCP 195] is an example of an authenticated protected channel when the certificate presented by the recipient is verified by the initiator.

Authentication

The process of establishing confidence in the identity of users or information systems. Authentication of users (subscribers) implies confirmation of the subscriber’s presence and intent to authenticate.

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 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 his/her identity. Secure authentication protocols also demonstrate to the claimant that he or she is communicating with the intended verifier.

Authentication Protocol Run

An exchange of messages between a claimant and a verifier that results in authentication (or authentication failure) between the two parties.

Authentication Secret

A generic term for any secret value that could be used by an attacker 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 claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the claimant’s identity. In previous editions of SP 800-63, this was referred to as a *token*.

Authenticator Assurance Level (AAL)

A metric describing robustness of the authentication process proving that the claimant is in control of a given subscriber’s authenticator(s).

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 are dependent upon the authenticator output, but they may or may not explicitly contain it.

Authenticator Secret

The secret value contained within an authenticator.

Biometrics

Automated recognition of individuals based on their behavioral and biological characteristics.

In this document, biometrics may be used to unlock multi-factor authenticators and prevent repudiation of registration.

Challenge-Response Protocol

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

Claimant

A party whose identity is to be verified using one or more authentication protocols.

Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA)

An interactive feature added to web-forms to distinguish use of the form by humans as opposed to automated agents. Typically, it requires entering text corresponding to a distorted image or from a sound stream.

Credential

An object or data structure that authoritatively binds an identity (and optionally, additional attributes) to an authenticator possessed and controlled by a subscriber.

While common usage often assumes that the credential is maintained by the subscriber, this document also uses the term to refer to electronic records maintained by the CSP which establish a binding between the subscriber’s authenticator(s) and their identity.

Credential Service Provider (CSP)

A trusted entity that issues or registers subscriber authenticators and issues electronic credentials to subscribers. The CSP may encompass verifiers that it operates. A CSP may be an independent third party, or may issue credentials for its own use.

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 to an attacker’s website which causes 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, merely by viewing a malicious link in a webmail message 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 and can therefore compromise the confidentiality and integrity of data transfers between the website and client. Websites are vulnerable if they display user supplied data from requests or forms without sanitizing the data so that it is not executable.

Cryptographic Key

A value used to control cryptographic operations, such as decryption, encryption, signature generation or signature verification. For the purposes of this document, key requirements shall meet the minimum requirements stated in Table 2 of [SP 800-57 Part 1].

See also Asymmetric Keys, Symmetric Key.

Cryptographic Authenticator

An authenticator where the secret is a cryptographic key. A hardware cryptographic authenticator is a cryptographic module containing one or more cryptographic keys.

Cryptographic Module

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

Data Integrity

The property that data has not been altered by an unauthorized entity.

Derived Credential

A credential issued based on proof of possession and control of one or more authenticators associated with a previously issued credential, so as not to duplicate the identity proofing process.

Digital Authentication

The process of establishing confidence in user identities electronically presented to an information system. In previous editions of SP 800-63, this was referred to as *Electronic Authentication*.

Digital Signature

An asymmetric key operation where 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 but not confidentiality protection.

Eavesdropping Attack

An attack in which an attacker listens passively to the authentication protocol to capture information which can be used in a subsequent active attack to masquerade as the claimant.

Electronic Authentication (E-Authentication)

See *Digital Authentication*.

Entropy

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

Equal Error Rate (EER)

The value where the false match rate (FMR) and false non-match rate (FNMR) of a sensor are equal. EER is a figure of merit for the sensor; the lower the EER is, the more certain the sensor's decision is likely to be.

Federal Information Security Management Act (FISMA)

Title III of the E-Government Act requiring each federal agency to develop, document, and implement an agency-wide program to provide information security for the information and information systems that support the operations and assets of the agency, including those provided or managed by another agency, contractor, or other source.

Federal Information Processing Standard (FIPS)

Under the Information Technology Management Reform Act (Public Law 104-106), the Secretary of Commerce approves standards and guidelines that are developed by the National Institute of Standards and Technology (NIST) for Federal computer systems. These standards and guidelines are issued by NIST as FIPS for use government-wide. 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 through the FIPS home page: http://www.nist.gov/itl/fips.cfm (http://www.nist.gov/itl/fips.cfm)

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; and
2. Collision resistant - It is computationally infeasible to find any two distinct inputs that map to the same output.

Identity

A set of attributes that uniquely describe a person within a given context.

Identity Assurance Level (IAL)

A metric describing degree of confidence that the Applicant’s Claimed Identity is their real identity.

Identity Provider (IdP)

The party that manages the subscriber’s primary authentication credentials and issues assertions derived from those credentials. This is commonly the CSP as discussed within this document suite.

Kerberos

A widely used authentication protocol developed at MIT. In “classic” Kerberos, users share a secret password with a Key Distribution Center (KDC). The user, Alice, who wishes to communicate with another user, Bob, authenticates to the KDC and is furnished a “ticket” by the KDC to use to authenticate with Bob.

When Kerberos authentication is based on passwords, the protocol is known to be vulnerable to offline dictionary attacks by eavesdroppers who capture the initial user-to-KDC exchange. Longer password length and complexity provide some mitigation to this vulnerability, although sufficiently long passwords tend to be cumbersome for users.

Man-in-the-Middle Attack (MitM)

An attack on the authentication protocol in which the attacker positions himself or herself in between the claimant and verifier so that he or she can intercept and/or alter data traveling between them.

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 or an authenticator that requires more than one authentication factor for successful authentication. Multi-factor authentication can be performed using a single authenticator that provides more than one factor or by a combination of authenticators that provide different factors.

The three authentication factors are something you know, something you have, and something you are.

Network

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

Offline Attack

An attack where the attacker obtains some data (typically by eavesdropping on an authentication protocol run or by penetrating a system and stealing security files) that he/she is able to analyze in a system of his/her own choosing.

Online Attack

An attack against an authentication protocol where 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.

Passive Attack

An attack against an authentication protocol where the attacker intercepts data traveling along the network between the claimant and verifier, but does not alter the data (i.e., eavesdropping).

Password

A secret that a claimant memorizes and uses to authenticate his or her identity. Passwords are typically character strings.

Personal Identification Number (PIN)

A password consisting only of decimal digits.

Pharming

An attack in which an attacker corrupts an infrastructure service such as DNS (Domain Name Service) causing 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.

Possession and control of an authenticator

The ability to activate and use the authenticator in an authentication protocol.

Practice Statement

A formal statement of the practices followed by the parties to an authentication process (i.e., RA, CSP, or verifier). It usually describes the policies and practices of the parties and can become legally binding.

Private Credentials

Credentials that cannot be disclosed by the CSP because the contents can be used to compromise the authenticator.

Private Key

The secret part of an asymmetric key pair that is used to digitally sign or decrypt data.

Protected Session

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

A participant is said to be *authenticated* if, during the session, he, she or it proves possession of an authenticator in addition to the session keys, and if the other party can verify the identity associated with that authenticator. If both participants are authenticated, the protected session is said to be *mutually authenticated*.

Public Credentials

Credentials that describe the binding in a way that does not compromise the authenticator.

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 the name of a subscriber to a public key. The certificate indicates that the subscriber identified in the certificate has sole control and access to the private key. See also [RFC 5280].

Public Key Infrastructure (PKI)

A set of policies, processes, server platforms, software and workstations used for the purpose of administering 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.

Registration

The process through which an applicant applies to become a subscriber of a CSP and the CSP validates the identity of the applicant.

Relying Party (RP)

An entity that relies upon the subscriber’s authenticator and credentials, either directly or via a verifier, to establish a claimant’s identity, typically to process a transaction or grant access to information or a system.

Remote

(*As in remote authentication or remote transaction*) An information exchange between network-connected devices where the information cannot be reliably protected end-to-end by a single organization’s security controls.

Note: Any information exchange across the Internet is considered remote.

Replay Attack

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

Risk Assessment

The process of identifying the risks to system security and determining the probability of occurrence, the resulting impact, and additional safeguards that would mitigate this impact. Part of Risk Management and synonymous with Risk Analysis.

Salt

A non-secret value that is 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)*.

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 use of a session secret that the subscriber's software (a browser, application, or OS) can present to the RP or CSP in lieu of the subscriber's authentication credentials.

Session Hijack Attack

An attack in which the attacker is able to insert himself or herself 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 also be similarly compromised.

Shared Secret

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

Side Channel Attack

An attack enabled by leakage of information from a physical cryptosystem. Timing, power consumption, electromagnetic and acoustic emissions are examples of characteristics that could be exploited in a side-channel attack.

Social Engineering

The act of deceiving an individual into revealing sensitive information by associating with the individual to gain confidence and trust.

Special Publication (SP)

A type of publication issued by NIST. Specifically, the SP 800-series reports on the Information Technology Laboratory's research, guidelines, and outreach efforts in computer security, and its collaborative activities with industry, government, and academic organizations.

Subscriber

A party who has received a credential bound to an authenticator from a CSP.

Symmetric Key

A cryptographic key that is used to perform both the cryptographic operation and its inverse, for example to encrypt and decrypt, or create a MAC and to verify the code.

Token

See *Authenticator*.

Token Authenticator

See *Authenticator Output*.

Token Secret

See *Authenticator Secret*.

Transport Layer Security (TLS)

An authentication and security protocol widely implemented in browsers and web servers. TLS is defined by [RFC 5246]. TLS is similar to the older SSL protocol, and TLS 1.0 is effectively SSL version 3.1. NIST [SP 800-52], *Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations* specifies how TLS is to be used in government applications.

Trust Anchor

A public or symmetric key that is trusted because it is directly built into hardware or software, or securely provisioned via out-of-band means, rather than because it is vouched for by another trusted entity (e.g. in a public key certificate).

Usability

Per ISO/IEC 9241-11: 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

Verifier

An entity that verifies the claimant's identity by verifying the claimant's possession and control of one or two authenticators using an authentication protocol. To do this, the verifier may also need to validate credentials that link the authenticator(s) and identity and check their status.

Verifier Impersonation

A scenario where the attacker impersonates the verifier in an authentication protocol, usually to capture information that can be used to masquerade as a subscriber to the real verifier. In previous editions of SP 800-63, authentication protocols that are resistant to verifier impersonation have been described as "strongly MitM resistant".

Weakly Bound Credentials

Credentials that are bound to a subscriber in a manner than can be modified without invalidating the credential.

Zeroize

Overwrite a memory location with data consisting entirely of bits with the value zero so that the data is destroyed and not recoverable. This is often contrasted with deletion methods that merely destroy reference to data within a file system rather than the data itself.

Zero-knowledge Password Protocol

A password based authentication protocol that allows a claimant to authenticate to a verifier without revealing the password to the verifier. Examples of such protocols are EKE, SPEKE and SRP.

4. Authenticator Assurance Levels

This section is normative.

In order to satisfy the requirements of a given AAL, a claimant SHALL be authenticated with at least a given level of strength to be recognized as a subscriber. The result of an authentication process is an identifier that MAY be pseudonymous and that SHALL be used each time that subscriber authenticates to that RP. Optionally, 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 Section 5.

FIPS 140 requirements are satisfied by [FIPS 140-2] or newer revisions.

Table 4-1 lists strict adherence to M-04-04 Level of Assurance, mapping the corresponding AALs.

Table 4-1. Legacy M-04-04 AAL Requirements

M-04-04 Level of Assurance (LOA)	Authenticator Assurance Level (AAL)
1	1
2	2 or 3
3	2 or 3
4	3

However, Table 4-2 shows the expanded set of AALs that are allowable to meet M-04-04 Levels of Assurance. Agencies SHALL select the corresponding AAL based on the impact of an authentication failure.

Table 4-2. Recommended M-04-04 AAL Requirements

M-04-04 Level of Assurance	Authenticator Assurance Level
1, without making personal data available	1, 2 or 3
1, making personal data available	2 or 3
2	2 or 3
3	2 or 3
4	3

At IAL1, it is possible that attributes are collected and made available by the digital service. Any personal data, whether self-asserted or validated, requires multi-factor authentication; therefore agencies SHALL select a minimum of AAL2 when self-asserted personal data, collected at IAL1, is made available online.

4.1. Authenticator Assurance Level 1

AAL1 provides some assurance that the claimant controls an authenticator registered to the subscriber. AAL1 requires single-factor authentication using a wide range of available authentication technologies. Successful authentication requires that the claimant prove possession and control of the authenticator through a secure authentication protocol.

4.1.1. Permitted Authenticator Types

AAL1 permits the use of any of the following authenticator types, which are defined in Section 5:

- Memorized Secret
- Look-up Secret
- Out of Band
- Single-factor One-Time Password (OTP) Device
- Multi-factor OTP Device
- Single-factor Cryptographic Software
- Single-factor Cryptographic Device
- Multi-factor Cryptographic Software
- Multi-factor Cryptographic Device

4.1.2. Authenticator and Verifier Requirements

Cryptographic authenticators used at AAL1 SHALL use approved cryptography. Software-based authenticators that operate within the context of a general purpose operating system MAY, where practical, attempt to detect compromise of the platform in which they are running (e.g., by malware) and SHOULD decline to operate when such a compromise is detected.

Communication between the claimant and channel (the primary channel in the case of an Out of Band authenticator) SHALL be via an authenticated protected channel to provide confidentiality of the authenticator output and resistance to MitM attacks.

Verifiers operated by government agencies at AAL1 SHALL be validated to meet the requirements of [FIPS 140] Level 1.

4.1.3. Reauthentication

At AAL1, reauthentication of the subscriber SHOULD be repeated at least once per 30 days, regardless of user activity.

4.1.4. Security Controls

The CSP SHOULD employ appropriately tailored security controls from the low baseline of security controls defined in [SP 800-53] or equivalent industry standard and SHOULD ensure that the minimum assurance requirements associated with the *low* baseline are satisfied.

4.1.5. Records Retention

The CSP shall comply with their respective records retention policies in accordance with applicable laws and regulations. If the CSP opts to retain records in the absence of any legal requirements, the CSP SHALL conduct a privacy risk assessment to determine how long records should be retained.

4.2. Authenticator Assurance Level 2

AAL2 provides high confidence that the claimant controls authenticator(s) registered to the subscriber. Proof of possession and control of two different authentication factors is required through a secure authentication protocol. Approved cryptographic techniques are required at AAL2 and above.

4.2.1. Permitted Authenticator Types

At AAL2, it is required to have either a multi-factor authenticator or a combination of two single-factor authenticators. Authenticator requirements are specified in Section 5.

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

- Multi-factor OTP Device
- Multi-factor Cryptographic Software
- Multi-factor Cryptographic Device

When a combination of two single-factor authenticators is used, it SHALL include a Memorized Secret authenticator and one possession-based (“something you have”) authenticator from the following list:

- Look-up Secret
- Out of Band
- Single-factor OTP Device
- Single-factor Cryptographic Software
- Single-factor Cryptographic Device

> Note: When biometric authentication implements the requirements in Section 5.2.3 the device has to be authenticated. Therefore, it is unnecessary to implement another factor with biometrics as the device is “something you have”, which serves as a valid second factor of the authenticator.

4.2.2. Authenticator and Verifier Requirements

Cryptographic authenticators used at AAL2 SHALL use approved cryptography. Authenticators procured by government agencies SHALL be validated to meet the requirements of [FIPS 140] Level 1. Software-based authenticators that operate within the context of a general purpose operating system MAY, where practical, attempt to detect compromise of the platform in which they are running (e.g., by malware) and SHOULD decline to operate when such a compromise is detected. At least one authenticator used at AAL2 SHALL be replay resistant as described in Section 5.2.8. Authentication at AAL2 SHOULD demonstrate authentication intent from at least one authenticator as discussed in Section 5.2.9.

Communication between the claimant and verifier (the primary channel in the case of an Out of Band authenticator) SHALL be via an authenticated protected channel to provide confidentiality of the authenticator output and resistance to MitM attacks.

Verifiers operated by government agencies at AAL2 SHALL be validated to meet the requirements of [FIPS 140] Level 1.

When a biometric factor is used in authentication at AAL2, the verifier SHOULD make a determination that the biometric sensor and subsequent processing meet the performance requirements stated in Section 5.2.3.

4.2.3. Reauthentication

At AAL2, authentication of the subscriber SHALL be repeated at least once per 12 hours, regardless of user activity. Reauthentication of the subscriber SHALL be repeated following no more than 30 minutes of user inactivity. The CSP MAY prompt the user to cause activity just before the inactivity timeout, if desired. Reauthentication MAY use a single authentication factor.

4.2.4. Security Controls

The CSP SHOULD employ appropriately tailored security controls from the moderate baseline of security controls defined in [SP 800-53] or equivalent industry standard and SHOULD ensure that the minimum assurance requirements associated with the *moderate* baseline are satisfied.

4.2.5. Records Retention

CSPs shall comply with their respective records retention policies in accordance with whatever laws and regulations apply to those entities. If the CSP opts to retain records in the absence of any legal requirements, the CSP SHALL conduct a privacy risk assessment to determine how long records should be retained.

4.3. Authenticator Assurance Level 3

AAL3 provides very high confidence that the claimant controls authenticator(s) registered to the subscriber. Authentication at AAL3 is based on proof of possession of a key through a cryptographic protocol. AAL3 is like AAL2 but also requires a “hard” cryptographic authenticator that provides verifier impersonation resistance.

4.3.1. Permitted Authenticator Types

Authentication Assurance Level 3 requires the use of one of two kinds of hardware devices:

- Multi-factor Cryptographic Device
- Single-factor Cryptographic Device used in conjunction with Memorized Secret

4.3.2. Authenticator and Verifier Requirements

Communication between the claimant and channel SHALL be via an authenticated protected channel to provide confidentiality of the authenticator output and resistance to MitM attacks. All cryptographic device authenticators used at AAL3 SHALL be verifier impersonation resistant as described in Section 5.2.5 and SHALL be replay resistant as described in Section 5.2.8. All authentication and reauthentication processes at AAL3 SHALL demonstrate authentication intent from at least one authenticator as described in Section 5.2.9.

Multi-factor authenticators used at AAL3 SHALL be hardware cryptographic modules validated at [FIPS 140] Level 2 or higher overall with at least [FIPS 140] Level 3 physical security. Single-factor cryptographic devices used at AAL3 SHALL be validated at [FIPS 140] Level 1 or higher overall with at least [FIPS 140] Level 3 physical security.

Verifiers at AAL3 SHALL be validated at [FIPS 140] Level 1 or higher.

When a biometric factor is used in authentication at AAL3, the verifier SHALL make a determination that the biometric sensor and subsequent processing meet the performance requirements stated in Section 5.2.3.

4.3.3. Reauthentication

At AAL3, authentication of the subscriber SHALL be repeated at least once per 12 hours, regardless of user activity. Reauthentication of the subscriber SHALL be repeated following a period of no more than 15 minutes of user inactivity. Reauthentication SHALL use both authentication factors. The verifier MAY prompt the user to cause activity just before the inactivity timeout.

4.3.4. Security Controls

The CSP SHOULD employ appropriately tailored security controls from the high baseline of security controls defined in [SP 800-53] or an equivalent industry standard and SHOULD ensure that the minimum assurance requirements associated with the *high* baseline are satisfied.

4.3.5. Records Retention

The CSP SHALL comply with their respective records retention policies in accordance with whatever laws and regulations apply to those entities. If the CSP opts to retain records in the absence of any legal requirements, the CSP SHALL conduct a privacy risk assessment to determine how long records should be retained.

4.4. Privacy Requirements

The CSP SHOULD employ appropriately tailored privacy controls defined in [SP 800-53] or equivalent industry standard.

CSPs SHALL NOT use or disclose information about authenticators for any purpose other than conducting authentication or to comply with law or legal process, unless the CSP provides clear notice and obtains consent from the subscriber for additional uses. CSPs MAY NOT make consent a condition of the service. Care SHALL be taken to ensure that use of such information is limited to its original purpose for collection. If the use of such information does not fall within uses related to authentication or to comply with law or legal process, the CSP SHALL provide notice and obtain consent from the subscriber. This notice SHOULD follow the same principles as described in *Notice and Consent* in [SP 800-63A Section 8.2] (sp800-63a.html#consent) and SHOULD not be rolled up into a legalistic privacy policy or general terms and conditions. Rather, if there are uses outside the bounds of these explicit purposes, the subscriber SHOULD be provided with a meaningful way to understand the purpose for additional uses, and the opportunity to accept or decline.

Regardless of whether the CSP is an agency or private sector provider, the following requirements apply to the agency offering or using the authentication service:

1. The agency SHALL consult with their Senior Agency Official for Privacy (SAOP) to conduct an analysis to determine whether the collection of Personally Identifiable Information (PII) to issue or maintain authenticators triggers the requirements of the *Privacy Act of 1974* [Privacy Act].
   - The agency SHALL publish a System of Records Notice (SORN) to cover such collections, as applicable.
   - The agency SHALL consult with their SAOP to 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].
   - The agency SHALL publish a Privacy Impact Assessment (PIA) to cover such collection, as applicable.

4.5. Summary of Requirements

(Informative; refer to preceding sections for normative requirements)

Table 4-3 summarizes the requirements for each of the AALs:

Table 4-3. AAL Summary of Requirements

Requirement	AAL1	AAL2	AAL3

Permitted authenticator types	Memorized Secret;
	Look-up Secret;
	Out of Band;
	SF OTP Device;
	MF OTP Device;
	SF Crypto Software;
	SF Crypto Device;
	MF Crypto Software;
	MF Crypto Device
		MF OTP Device;
		MF Crypto Software;
		MF Crypto Device;
		or memorized secret plus:
		• Look-up Secret
		• Out of Band
		• SF OTP Device
		• SF Crypto Software
		• SF Crypto Device
			MF Crypto Device
			SF Crypto Device plus Memorized Secret

FIPS 140 verification	Level 1 (Government agency verifiers)	Level 1 (Government agency authenticators and verifiers)	Level 2 overall (MF authenticators) Level 1 overall (verifiers and SF Crypto Devices) Level 3 physical security (all authenticators)
Reauthentication	30 days	12 hours or 30 minutes inactivity; may use one authentication factor	12 hours or 15 minutes inactivity; shall use both authentication factors
Security controls	[SP 800-53] Low Baseline (or equivalent)	[SP 800-53] Moderate Baseline (or equivalent)	[SP 800-53] High Baseline (or equivalent)
MitM resistance	Required	Required	Required
Verifier impersonation resistance	Not required	Not required	Required
Verifier compromise resistance	Not required	Not required	Required
Replay resistance	Not required	Required	Required
Authentication intent	Not required	Recommended	Required

5. Authenticator and Verifier Requirements

This section is normative.

This section provides the detailed requirements specific for each type of authenticator. With the exception of reauthentication requirements specified in Section 4 and the requirement for verifier impersonation resistance at AAL3 described in Section 5.2.5, the technical requirements for each of the authenticator types are the same regardless of the AAL at which the authenticator is used.

5.1. Requirements by Authenticator Type

5.1.1. Memorized Secrets

	A Memorized Secret authenticator (commonly referred to as a *password* or, if numeric, a *PIN*) is a secret value that is intended to be chosen and memorable by the user. Memorized secrets need to be of sufficient complexity and secrecy that it would be impractical for an attacker to guess or otherwise discover the correct secret value.

5.1.1.1. Memorized Secret Authenticators

Memorized secrets SHALL be at least 8 characters in length if chosen by the subscriber; memorized secrets chosen randomly by the CSP or verifier SHALL be at least 6 characters in length and MAY be entirely numeric. Since the CSP or verifier may disallow some choices of memorized secrets based on their appearance on a blacklist of compromised values, the subscriber SHALL choose a different memorized secret if a choice is rejected. No other complexity requirements for memorized secrets SHOULD be imposed; a rationale for this is presented in Appendix A.

5.1.1.2. Memorized Secret Verifiers

Verifiers SHALL require subscriber-chosen memorized secrets to be at least 8 characters in length. Verifiers SHOULD permit user-chosen memorized secrets to be up to 64 characters or more in length. All printing ASCII [RFC 20] characters as well as the space character SHOULD be acceptable in memorized secrets; Unicode [ISO/ISC 10646:2014] characters SHOULD be accepted as well. Verifiers MAY remove multiple consecutive space characters, or all space characters, prior to verification provided that the result is at least 8 characters in length. Truncation of the secret SHALL NOT be performed. For purposes of the above length requirements, each Unicode code point SHALL be counted as a single character.

If Unicode characters are accepted in memorized secrets, the verifier SHOULD apply the Normalization Process for Stabilized Strings defined in Section 12.1 of Unicode Standard Annex 15 [UAX 15] using either the NFKC or NFKD normalization. Subscribers choosing memorized secrets containing Unicode characters SHOULD be advised that some characters may be represented differently by some endpoints, which can affect their ability to authenticate successfully. This process is applied prior to hashing of the byte string representing the memorized secret.

Memorized secrets that are randomly chosen by the CSP (e.g., at enrollment) or by the verifier (e.g., when a user requests a new PIN) SHALL be at least 6 characters in length and SHALL be generated using an approved random bit generator.

Memorized secret verifiers SHALL NOT permit the subscriber to store a “hint” that is accessible to an unauthenticated claimant. Verifiers also SHALL NOT prompt subscribers to use specific types of information (e.g., “What was the name of your first pet?”) when choosing memorized secrets.

When processing requests to establish and change memorized secrets, verifiers SHALL compare the prospective secrets against a list that contains values known to be commonly-used, expected, or compromised. For example, the list MAY include (but is not limited to):

- Passwords obtained from previous breach corpuses.
- Dictionary words.
- Repetitive or sequential characters (e.g. ‘aaaaaa’, ‘1234abcd’).
- Context specific words, such as the name of the service, the username, and derivatives thereof.

If the chosen secret is found in the list, the CSP or verifier SHALL advise the subscriber that they need to select a different secret, SHALL provide the reason for rejection, and SHALL require the subscriber to choose a different value.

Verifiers SHALL implement a throttling mechanism that effectively limits the number of failed authentication attempts an attacker can make on the subscriber’s account as described in Section 5.2.2.

Verifiers SHOULD NOT impose other composition rules (e.g., mixtures of different character types) on memorized secrets. Verifiers SHOULD NOT require memorized secrets to be changed arbitrarily (e.g., periodically) and SHOULD only require a change if the subscriber requests a change or there is evidence of compromise of the authenticator.

In order to assist the claimant in entering a memorized secret successfully, the verifier SHOULD offer an option to display the secret (rather than a series of dots or asterisks, typically) until it is entered. This allows the claimant to verify their entry if they are in a location where their screen is unlikely to be observed. The verifier MAY also permit the user’s device to display individual entered characters for a short time after each character is typed to verify correct entry, particularly on mobile devices.

The verifier SHALL use approved encryption and SHALL utilize an authenticated protected channel when requesting memorized secrets in order to provide resistance to eavesdropping and MitM attacks.

Verifiers SHALL store memorized secrets in a form that is resistant to offline attacks. Secrets SHALL be hashed with a *salt* value using an approved hash function such as PBKDF2 as described in [SP 800-132]. The salt value SHALL be a 32-bit or longer random value generated by an approved random bit generator and stored along with the hash result. At least 10,000 iterations of the hash function SHOULD be performed. A keyed hash function (e.g., HMAC [FIPS198-1]), with the key stored separately from the hashed authenticators (e.g., in a hardware security module) SHOULD be used to further resist dictionary attacks against the stored hashed authenticators.

5.1.2. 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 secret(s) needed to respond to a prompt from the verifier. For example, a claimant may be asked by the verifier to provide a specific subset of the numeric or character strings printed on a card in table format. A common application of look-up secrets is the use of "recovery keys" stored by the subscriber for use in the event another authenticator is lost or malfunctions.

5.1.2.1. Look-up Secret Authenticators

CSPs creating look-up secret authenticators SHALL use an approved random bit generator to generate the list of secrets, and SHALL deliver the authenticator securely to the subscriber. Look-up secrets SHALL have at least 64 bits of entropy, or SHALL have at least 20 bits of entropy if the number of failed authentication attempts is limited as described in Section 5.2.2.

If the authenticator uses look-up secrets sequentially from a list, the subscriber MAY dispose of used secrets, but only after a successful authentication.

5.1.2.2. Look-up Secret Verifiers

Verifiers of look-up secrets SHALL prompt the claimant for the next secret from their authenticator or for a specific (e.g., numbered) secret. A given secret from an authenticator SHALL be used successfully only once; therefore, a given authenticator can only be used for a finite number of successful authentications. If the look-up secret is derived from a grid card, each cell of the grid SHALL be used only once.

Verifiers SHALL store look-up secrets in a form that is resistant to offline attacks. Secrets SHALL be hashed with a *salt* value using an approved hash function as described in [SP 800-132]. The salt value SHALL be a 128-bit or longer random value generated by an approved random bit generator that is stored along with the hash result. A keyed hash function (e.g., HMAC [FIPS198-1]), with the key stored separately from the hashed authenticators (e.g., in a hardware security module) SHOULD be used to further resist dictionary attacks against the stored hashed authenticators.

Look-up secrets SHALL be generated using an approved random bit generator and SHALL have at least 20 bits of entropy. When look-up secrets have less than 64 bits of entropy, the verifier SHALL implement a throttling mechanism that effectively limits the number of failed authentication attempts an attacker can make on the subscriber's account as described in Section 5.2.2.

The verifier SHALL use approved encryption and SHALL utilize an authenticated protected channel when requesting look-up secrets in order to provide resistance to eavesdropping and MitM attacks.

5.1.3. 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 that is separate from the primary channel for e-authentication. 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 on their mobile device and type it (typically a 6-digit code) into their authentication session.

- 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.

- The claimant compares secrets received from the primary channel and the secondary channel and confirms the authentication via the secondary channel.

The purpose of the secret is to securely bind the authentication operation on the primary and secondary channel. When the response is via the primary communication channel, the secret also establishes the claimant's control of the out-of-band device.

5.1.3.1. Out-of-Band Authenticators

The out-of-band authenticator SHALL establish a separate channel with the verifier in order 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 the device does not leak information from one to the other without the authorization of the claimant.

The out-of-band device SHOULD be uniquely addressable and communication over the secondary channel SHALL be private. Methods that do not prove possession of a specific device, such as voice-over-IP (VOIP) or email, SHALL NOT be used for out-of-band authentication.

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

- Establish an authenticated protected channel to the verifier using approved cryptography. The key used SHALL be stored in the most secure storage available on the device (e.g., keychain storage, trusted platform module, trusted execution environment).
- Authenticate to a public mobile telephone network using a SIM card or equivalent that uniquely identifies the device. This method SHALL only be used if a secret is being sent from the verifier to the out-of-band device via the telephone network (SMS or voice).

If a secret is sent by the verifier to the out-of-band device, the device SHOULD NOT display the authentication secret on a device while it is locked by the owner (i.e., requires an entry of a PIN, passcode, or biometric). However, authenticators SHOULD indicate the receipt of an authentication secret on a locked device.

If the out-of-band authenticator sends an approval message over the secondary communication channel (rather than by the claimant transferring a received secret to the primary communication channel), it SHALL do one of the following:

- The authenticator SHALL accept transfer of the secret from the primary channel which it SHALL send to the verifier over the secondary channel to associate the approval with the authentication transaction. The claimant MAY perform the transfer manually or use a technology such as a barcode or QR code to effect the transfer.
- The authenticator SHALL present a secret received via the secondary channel from the verifier and prompt the claimant to verify the consistency of that secret with the primary channel, prior to accepting a yes/no response from the claimant. It SHALL then send that response to the verifier.

5.1.3.2. Out-of-Band Verifiers

If out-of-band verification is to be made using the public switched telephone network (PSTN), the verifier SHALL verify that the pre-registered telephone number being used is associated with a physical device. Changing the pre-registered telephone number SHALL NOT be possible without two-factor authentication at the time of the change. Verifiers SHALL use known and verifiable routes to deliver the secret, for example, by using Class 2 SMS. Verifiers SHOULD be aware of indicators such as device swap, SIM change, number porting, or other abnormal behavior *before* using the PSTN to deliver an out-of-band authentication secret.

> Note: Out-of-band authentication using the PSTN (SMS or voice) is discouraged and is being considered for removal in future editions of this guideline.

If out-of-band verification is to be made using a secure application, such as on a smart phone, the verifier MAY send a push notification to that device. The verifier then waits for the establishment of an authenticated protected channel and verifies the authenticator’s identifying key. The verifier SHALL NOT store the identifying key itself, but SHALL use a verification method such as use of 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.

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

- Transfer of secret to primary channel - The verifier MAY signal the device containing the subscriber’s authenticator to indicate readiness to authenticate. It SHALL then transmit a random secret to the out-of-band authenticator. The verifier SHALL then wait for the secret to be returned on the primary communication channel.
- Transfer of secret to secondary channel - 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 on the secondary channel from the claimant’s out-of-band authenticator.
- Verification of secrets by claimant - The verifier SHALL display a random authentication secret to the claimant via the primary channel, and SHALL send the same secret to the out-of-band authenticator via the secondary channel for presentation to the claimant. It SHALL then wait for an approval (or disapproval) message via the secondary channel.

In all cases, the authentication SHALL be considered invalid if not completed within 5 minutes. In order to provide replay resistance as described in Section 5.2.7, verifiers SHALL accept a given authentication secret only once during the validity period.

The verifier SHALL generate random authentication secrets with at least 20 bits of entropy using an approved random bit generator. If the authentication secret has less than 64 bits of entropy, the verifier SHALL implement a throttling mechanism that effectively limits the number of failed authentication attempts an attacker can make on the subscriber's account as described in Section 5.2.2.

5.1.4. Single-factor OTP Device

A single-factor OTP device generates OTPs. This includes hardware devices as well as software-based OTP generators installed on devices such as mobile phones. This device has an embedded secret that is used as the seed for generation of OTPs and does not require activation through a second factor. The OTP is displayed on the device and manually input to the verifier, thereby proving possession and control of the device. An OTP device may, for example, display 6 characters at a time. A single-factor OTP device is *something you have*.

Single-factor OTP devices are similar to look-up secret authenticators with the exception that the secrets are cryptographically and independently generated by the authenticator and 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.

5.1.4.1. Single-factor OTP Authenticators

Single-factor OTP authenticators contain two persistent values. The first is a symmetric key that persists for the lifetime of the device. The second is a nonce that is 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 [SP 800-131A] (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 device over its lifetime.

The authenticator output is obtained by using an approved block cipher or hash function to combine the key and nonce in a secure manner. The authenticator output MAY be truncated to as few as 6 decimal digits (approximately 20 bits of entropy).

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 2 minutes. The OTP value associated with a given nonce SHALL be accepted only once.

5.1.4.2. Single-factor OTP Verifiers

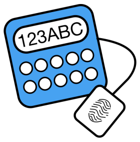
Single-factor OTP verifiers effectively duplicate the process of generating the OTP used by the authenticator. As such, the symmetric keys used by authenticators are also present in the verifier, and SHALL be strongly protected against compromise.

When a multi-factor OTP authenticator is being associated with a subscriber account, the verifier (or associated CSP) SHALL obtain secrets required to duplicate the authenticator output from the authenticator source (typically its manufacturer) using approved cryptography.

The verifier SHALL use approved encryption and an authenticated protected channel when collecting the OTP in order to provide resistance to eavesdropping and MitM attacks. Time-based OTPs SHALL have a lifetime of less than 2 minutes. In order to provide replay resistance as described in Section 5.2.7, verifiers SHALL accept a given time-based OTP only once during the validity period.

If the authenticator output has less than 64 bits of entropy, the verifier SHALL implement a throttling mechanism that effectively limits the number of failed authentication attempts an attacker can make on the subscriber's account as described in Section 5.2.2.

5.1.5. Multi-factor OTP Devices



A multi-factor OTP hardware device generates OTPs for use in authentication after activation through an additional authentication factor. The second factor of authentication may be achieved through some kind of 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 device and manually input to the verifier. For example, an OTP device may display 6 characters at a time, thereby proving possession and control of the device. The multi-factor OTP device is *something you have*, and it SHALL be activated by either *something you know* or *something you are*.

5.1.5.1. Multi-factor OTP Authenticators

Multi-factor OTP authenticators operate in a similar manner to single-factor OTP authenticators (see Section 5.1.4.1), except that they require the entry of either a memorized secret or use of a biometric to obtain a password from the authenticator. Each use of the authenticator SHALL require the input of the additional factor.

The authenticator output SHALL have at least 6 decimal digits of entropy. The output SHALL be generated by using an approved block cipher or hash function to combine a symmetric key stored on a personal hardware device with a nonce to generate an OTP. The nonce MAY be based on the date and time or on a counter generated on the device.

Any memorized secret used by the authenticator for activation SHALL be at least 6 decimal digits in length or of equivalent complexity and SHALL be rate limited as specified in Section 5.2.2. A biometric activation factor SHALL meet the requirements of Section 5.2.3, 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 such as a probe produced through signal processing) SHALL be erased from memory immediately after a password has been generated.

5.1.5.2. Multi-factor OTP Verifiers

Multi-factor OTP verifiers effectively duplicate the process of generating the OTP used by the authenticator, but without the requirement that a second factor be provided. As such, the symmetric keys used by authenticators SHALL be strongly protected against compromise.

When a multi-factor OTP authenticator is being associated with a subscriber account, the verifier (or associated CSP) SHALL obtain secrets required to duplicate the authenticator output from the authenticator source (typically its manufacturer) using approved cryptography. The verifier or CSP SHALL also establish, via the authenticator source, that the authenticator is a multi-factor device. In the absence of a trusted statement that it is a multi-factor device, the verifier SHALL treat it the authenticator as single-factor, in accordance with Section 5.1.4.

The verifier SHALL use approved encryption and SHALL utilize an authenticated protected channel when collecting the OTP in order to provide resistance to eavesdropping and MitM attacks. Time-based OTPs SHALL have a lifetime of less than 2 minutes. In order to provide replay resistance as described in Section 5.2.7, verifiers SHALL accept a given time-based OTP only once during the validity period.

If the authenticator output or activation secret has less than 64 bits of entropy, the verifier SHALL implement a throttling mechanism that effectively limits the number of failed authentication attempts an attacker can make on the subscriber's account as described in Section 5.2.2. A biometric activation factor SHALL meet the requirements of Section 5.2.3, including limits on the number of consecutive authentication failures.

5.1.6. Single-factor Cryptographic Software

A single-factor software cryptographic authenticator is a cryptographic key stored on disk or some other "soft" media. Authentication is accomplished by proving possession and control of the key. The authenticator output is highly dependent on the specific cryptographic protocol, but it is generally some type of signed message. The single-factor software cryptographic authenticator is *something you have*.

5.1.6.1. Single-factor Cryptographic Software Authenticators

Single-factor software cryptographic authenticators encapsulate a secret key that is unique to the authenticator. The key SHALL be stored in the most secure storage available on the device (e.g., keychain storage, trusted platform module, or trusted execution environment if available). The key SHALL be strongly protected against unauthorized disclosure by the use of access controls that limit access to the key to only those software components on the device requiring access.

5.1.6.2. Single-factor Cryptographic Software Verifiers

The requirements for a single-factor cryptographic software verifier are identical to those for a single-factor cryptographic device verifier, described in Section 5.1.7.2.

5.1.7. Single-factor Cryptographic Devices

A single-factor cryptographic device is a hardware device that performs cryptographic operations using protected cryptographic key(s) and provides the authenticator output via direct connection to the user endpoint. The device uses embedded symmetric or asymmetric cryptographic keys, and does not require activation through a second factor of authentication. Authentication is accomplished by proving possession of the device via the authentication protocol. The authenticator output is provided by direct connection to the user endpoint and is highly dependent on the specific cryptographic device and protocol, but it is typically some type of signed message. A single-factor cryptographic device is *something you have*.

5.1.7.1. Single-factor Cryptographic Device Authenticators

Single-factor cryptographic device authenticators encapsulate a secret key that is unique to the device and SHALL NOT be exportable (i.e., it cannot be removed from the device). The authenticator operates by signing a challenge nonce presented through a direct computer interface such as a USB port. Although cryptographic devices contain software, they differ from cryptographic software authenticators by the fact that all embedded software is under control of the CSP (or other issuer), and that the entire authenticator is subject to any applicable FIPS 140 requirements at the AAL being authenticated.

The secret key and its algorithm SHALL provide at least the minimum security length specified in the latest revision of [SP 800-131A] (112 bits as of the date of this publication). The challenge nonce SHALL be at least 64 bits in length. Approved cryptography SHALL be used.

Single-factor cryptographic device authenticators SHOULD require a physical input such as the pressing of a button in order to operate. This provides defense against unintended operation of the device, which might occur if the device to which it is connected is compromised.

5.1.7.2. Single-factor Cryptographic Device Verifiers

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

The verifier has either symmetric or asymmetric cryptographic keys corresponding to each authenticator. While both types of keys SHALL be protected against modification, symmetric keys SHALL additionally be strongly protected against unauthorized disclosure.

The challenge nonce SHALL be at least 64 bits in length, and SHALL either be unique over the lifetime of the authenticator or statistically unique (generated using an approved random bit generator).

5.1.8. Multi-factor Cryptographic Software

A multi-factor software cryptographic authenticator is a cryptographic key is stored on disk or some other "soft" media that requires activation through a second factor of authentication. Authentication is accomplished by proving possession and control of the key. The authenticator output is highly dependent on the specific cryptographic protocol, but it is generally some type of signed message. The multi-factor software cryptographic authenticator is *something you have*, and it SHALL be activated by either *something you know* or *something you are*.

5.1.8.1. Multi-factor Cryptographic Software Authenticators

Multi-factor software cryptographic authenticators encapsulate a secret key that is unique to the authenticator and is accessible only through the input of an additional factor, either a memorized secret or a biometric. The key SHOULD be stored in the most secure storage available on the device (e.g., keychain storage, trusted platform module, trusted execution environment).

Each authentication operation using the authenticator SHALL require the input of both factors.

Any memorized secret used by the authenticator for activation SHALL be at least 6 decimal digits in length or of equivalent complexity and SHALL be rate limited as specified in Section 5.2.2. A biometric activation factor SHALL meet the requirements of Section 5.2.3, and SHALL include limits on the allowable number of consecutive authentication failures.

The unencrypted key and activation secret or biometric sample (and any biometric data derived from the biometric sample such as a probe produced through signal processing) SHALL be erased from memory immediately after an authentication transaction has taken place.

5.1.8.2. Multi-factor Cryptographic Software Verifiers

The requirements for a multi-factor cryptographic software verifier are identical to those for a multi-factor cryptographic device verifier, described in Section 5.1.9.2.

5.1.9. Multi-factor Cryptographic Devices

A multi-factor cryptographic device is a hardware device that performs cryptographic operations using a protected cryptographic key(s) that require activation through a second authentication factor. Authentication is accomplished by proving possession of the device and control of the key. The authenticator output is provided by direct connection to the user endpoint and is highly dependent on the specific cryptographic device and protocol, but it is typically some type of signed message. The multi-factor cryptographic device is *something you have*, and it SHALL be activated by either *something you know* or *something you are*.

5.1.9.1. Multi-factor Cryptographic Device Authenticators

Multi-factor cryptographic device authenticators use tamper-resistant hardware to encapsulate a secret key that is unique to the authenticator and is accessible only through the input of an additional factor, either a memorized secret or a biometric. The authenticator operates by signing a challenge nonce presented through a direct computer interface such as a USB port. Although cryptographic devices contain software, they differ from cryptographic software authenticators by the fact that all embedded software is under control of the CSP (or manufacturer), and that the entire authenticator is subject to any applicable FIPS 140 requirements at the AAL being authenticated.

The secret key and its algorithm SHALL provide at least the minimum security length specified in the latest revision of [SP 800-131A] (112 bits as of the date of this publication). The challenge nonce SHALL be at least 64 bits in length. Approved cryptography SHALL be used.

Each authentication operation using the authenticator SHOULD require the input of the additional factor. Input of the additional factor MAY be accomplished via either direct input on the device or via a hardware connection (e.g., USB, smartcard).

Any memorized secret used by the authenticator for activation SHALL be at least 6 decimal digits in length or of equivalent complexity and SHALL be rate limited as specified in Section 5.2.2. A biometric activation factor SHALL meet the requirements of Section 5.2.3, and SHALL include 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 such as a probe produced through signal processing) SHALL be overwritten in memory immediately after an authentication transaction has taken place.

5.1.9.2. Multi-factor Cryptographic Device Verifiers

Multi-factor cryptographic device verifiers generate a challenge nonce, send it to the corresponding authenticator, and use the authenticator output to verify possession of the device and activation factor.

The verifier has either symmetric or asymmetric cryptographic keys corresponding to each authenticator. While both types of keys SHALL be protected against modification, symmetric keys SHALL additionally be strongly protected against unauthorized disclosure.

The challenge nonce SHALL be at least 64 bits in length, and SHALL either be unique over the lifetime of the authenticator or statistically unique (generated using an approved random bit generator). The verification operation SHALL use approved cryptography.

5.2. General Authenticator Requirements

5.2.1. Physical Authenticators

CSPs SHALL provide subscriber instructions on how to appropriately protect the authenticator against theft or loss. The CSP SHALL provide a mechanism to revoke or suspend the authenticator immediately upon notification from subscriber that loss or theft of the authenticator is suspected.

5.2.2. Rate Limiting (Throttling)

When required in the authenticator type descriptions under Section 5.1, the verifier SHALL implement controls to protect against online guessing attacks. Unless otherwise specified in the description of a given authenticator, the verifier SHALL effectively limit online attackers to no more than 100 consecutive failed attempts on a single account.

Additional techniques MAY be used to prioritize authentication attempts that are likely to come from the subscriber over those that are more likely to come from an attacker:

- Requiring the claimant to complete a CAPTCHA before attempting authentication.
- Requiring the claimant to wait following a failed attempt for a period of time that is increasing in intervals from, say, 30 seconds to an hour, as the account approaches its maximum allowance for consecutive failed attempts.
- Only accepting authentication requests from a white list of IP addresses at which the subscriber has been successfully authenticated before.
- Leveraging other risk-based or adaptive authentication techniques to identify user behavior that falls within, or out of, typical norms.

When the subscriber successfully authenticates, the verifier SHOULD disregard any previous failed attempts from the same IP address.

5.2.3. Use of Biometrics

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

- Biometric False Match Rates (FMR) and False Non-Match Rates (FNMR) do not provide confidence in the authentication of the subscriber by themselves. In addition, FMR and FNMR do not account for spoofing attacks.

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

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

Biometrics SHALL be used with another authentication factor (something you have).

An authenticated protected channel between sensor (or endpoint containing a sensor that resists sensor replacement) and verifier SHALL be established and the sensor or endpoint authenticated **prior** to capturing the biometric sample from the claimant.

Empirical testing of the biometric system to be deployed SHALL demonstrate an EER of **1 in 1000** or better with respect to matching performance. The biometric system SHALL operate with an FMR of **1 in 1000** or better.

The biometric system SHOULD implement PAD. Testing of the biometric system to be deployed SHOULD demonstrate at least 90% resistance to presentation attacks for each relevant attack type (aka species), where resistance is defined as the number of thwarted presentation attacks divided by the number of trial presentation attacks.

> Note: PAD is being considered as a mandatory requirement in future editions of this guideline.

The biometric system SHALL allow no more than 5 consecutive failed authentication attempts or 10 consecutive failed attempts if PAD meeting the above requirements is implemented. Once that limit has been reached, the biometric authenticator SHALL either:

- Impose a delay of at least 30 seconds before the next attempt, increasing exponentially with each successive attempt, e.g., 1 minute before the following failed attempt, 2 minutes before the second following attempt, etc.

OR

- Disable the biometric user verification and offer another factor (a different biometric modality or a PIN/Passcode if it is not already a required factor) if such an alternative method is already implemented.

Determination of sensor/endpoint performance, integrity, and authenticity can be accomplished in several different ways, any of which are acceptable under this guideline. These include but are not limited to: authentication of the sensor or endpoint, certification by an approved accreditation authority, or runtime interrogation of signed metadata (e.g., attestation) as described in Section 5.2.4.

Biometric matching SHOULD be performed locally on claimant’s device or MAY be performed at a central verifier.

If matching is performed centrally:

- Use of the biometric SHALL be limited to one or more specific devices that are identified using approved cryptography.
- Biometric revocation, referred to as biometric template protection in ISO/IEC 24745, SHALL be implemented.
- All transmission of biometrics shall be over the authenticated protected channel.

Biometric samples collected in the authentication process MAY be used to train matching algorithms or, with user consent, for other research purposes. Biometric samples (and any biometric data derived from the biometric sample such as a probe produced through signal processing) SHALL be erased from memory immediately after any training or research data has been derived.

Biometrics are also used in some cases to prevent repudiation of registration and to verify that the same individual participates in all phases of the registration process as described in SP 800-63A.

5.2.4. Attestation

Attestation is information conveyed to the verifier regarding a directly connected authenticator or the endpoint involved in an authentication operation. Information conveyed by attestation MAY include, but is not limited to:

- The provenance (manufacturer or supplier certification), health, and integrity of the authenticator and/or endpoint.
- Security features of the authenticator.

- Security and performance characteristics of biometric sensor(s).
- Sensor modality.

If this attestation is signed, it SHALL be signed using a digital signature that provides at least the minimum security strength specified in the latest revision of [SP 800-131A] (112 bits as of the date of this publication).

Attestation information MAY be used as part of a risk-based authentication decision.

5.2.5. Verifier Impersonation Resistance

Verifier impersonation attacks, sometimes referred to as “phishing attacks”, refer to attempts by fraudulent verifiers and RPs to fool an unwary claimant into authenticating to an impostor website. In previous editions of SP 800-63, protocols that are resistant to verifier impersonation attacks were also referred to as “strongly MitM resistant”.

Authentication protocols that are verifier impersonation resistant SHALL authenticate the verifier and either:

1. Strongly and irreversibly bind the authenticator output to the public key of the certificate presented by the verifier to which it is sent, or to that verifier’s authenticated hostname or domain name; or
2. Determine whether the verifier’s authenticated hostname or domain name is on a list of trusted verifiers, and release the authenticator output only to a verifier on that list.

One example of the former class of verifier impersonation resistant authentication protocols is client-authenticated 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. Other protocols that MAY be used are techniques that irreversibly include the verifier’s hostname or domain in the generation of the authenticator output, making that authenticator output unusable by a fraudulent verifier (the attacker) if proxied to the intended verifier.

The latter class of verifier impersonation resistant protocols relies on access control to release the authenticator output only to trusted verifiers.

In contrast, authenticators that involve the manual entry of an authenticator output, such as out of band and OTP authenticators, SHALL NOT be considered verifier impersonation resistant because they assume the vigilance of the claimant to determine that they are communicating with the intended verifier.

5.2.6. Verifier-CSP Communications

In situations where the verifier and CSP are separate entities (as shown by the dotted line in [SP 800-63-3 Figure 4-1] (sp800-63-3.html#63Sec4-Figure1)), communications between the verifier and CSP SHALL occur through a mutually-authenticated secure channel (such as a client-authenticated TLS connection) using approved cryptography.

5.2.7. Verifier Compromise Resistance

Use of some types of authenticators requires that the verifier store a copy of the authenticator secret. For example, an OTP authenticator (described in Section 5.1.4) requires that the verifier independently generate the authenticator output for comparison against the value sent by the claimant. Because of the potential for the verifier to be compromised and stored secrets stolen, authentication protocols that do not require the verifier to persistently store secrets that could be used for authentication are considered stronger, and are described herein as being *verifier compromise resistant*. Note that such verifiers are not resistant to all attacks; a verifier could be compromised in a different way, such as to always accept a particular authenticator output.

Verifier compromise resistance can be achieved in different ways, for example:

1. Use a cryptographic authenticator that requires that the verifier store a public key corresponding to a private key held by the authenticator.
2. Store the expected authenticator output in hashed form. This method can be used with some look-up secret authenticators (described in Section 5.1.2), for example.

In order to be considered verifier compromise resistant, public keys stored by the verifier SHALL use approved cryptography and SHALL provide at least the minimum security strength specified in the latest revision of [SP 800-131A] (112 bits as of the date of this publication).

Other verifier compromise resistant secrets SHALL use approved hash algorithms and the underlying secrets SHALL have at least the minimum security strength specified in the latest revision of [SP 800-131A] (112 bits as of the date of this publication). Note that secrets (such as memorized secrets) having lower complexity SHALL NOT be considered verifier compromise resistant when hashed because of the potential to defeat the hashing process through dictionary lookup or exhaustive search.

5.2.8. 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 prior to 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 that the old protocol messages replayed do not contain the appropriate nonces or timeliness data related to the current authentication session.

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

In contrast, memorized secrets are not considered replay resistant because the authenticator output (the secret itself) is provided for each authentication.

5.2.9. Authentication Intent

An authentication process requires intent if it requires the subject to explicitly respond to each authentication or reauthentication request. The goal of authentication intent is to make it more difficult for directly connected physical authenticators (cryptographic devices) to be used without the subject's knowledge, such as by malware on the endpoint. Authentication intent SHALL be established by the authenticator itself, although multi-factor cryptographic devices MAY establish intent by reentry of the other authentication factor on the endpoint with which the authenticator is used.

Authentication intent MAY be established in a number of ways. Authentication processes that require intervention of the subject, e.g., to enter an authenticator output on their endpoint from an OTP device, establish intent by their very nature. Cryptographic devices that require user action (e.g., pushing a button or reinsertion) for each authentication or reauthentication operation are also considered to establish intent.

6. Authenticator Lifecycle Management

This section is normative.

During the lifecycle of an authenticator bound to a subscriber's identity, a number of events can occur that affect the use of that authenticator. These events include binding, loss, theft, unauthorized duplication, expiration, and revocation. This section describes the actions that SHALL be taken in response to those events.

6.1. Authenticator Binding

Authenticators MAY be issued (provided) by a CSP as part of a process such as enrollment; in other cases, the subscriber MAY provide their own, such as software or hardware cryptographic modules. For this reason, this guideline refers to the *binding* of an authenticator rather than the issuance, but this does not exclude the possibility that an authenticator is issued as well.

Throughout the online identity lifecycle, CSPs SHALL maintain a record of all authenticators that are or have been associated with the identity. The CSP or verifier SHALL also maintain the information required for throttling authentication attempts when required, as described in section 5.2.2.

The record created by the CSP SHALL contain the date and time the authenticator was bound to the account and SHOULD include information about the binding, such as the IP address and any device identifier associated with the enrollment. If available, the record SHOULD also contain information about unsuccessful authentications attempted with the authenticator.

6.1.1. Enrollment

The following requirements apply when an authenticator is bound to an identity as a result of a successful identity proofing transaction, as described in [SP 800-63A] (sp800-63a.html).

At IAL2, the CSP SHALL bind at least one, and SHOULD bind at least two, authenticators to the subscriber's online identity. Binding of multiple authenticators is preferred in order to recover from loss or theft of their primary authenticator. While at IAL1 all identifying information is self-asserted, creation of online material or an online reputation makes it undesirable to lose control of an account as result of the loss of an authenticator. The second authenticator makes it possible to securely recover from that situation and thus a CSP SHOULD bind at least two authenticators to the subscriber's credential at IAL1 as well.

At IAL2 and above, identifying information is associated with the online identity and the subscriber has undergone an identity proofing process as described in SP 800-63A. As a result, authenticators at the same AAL as the desired IAL SHALL be bound to the account. For example, if the subscriber has successfully completed proofing at IAL2, AAL2 or AAL3 authenticators are appropriate to bind to the IAL2 identity. While a CSP MAY bind an AAL1 authenticator to an IAL2 identity, if the subscriber is authenticated at AAL1, the CSP SHALL NOT expose personal information, even if self-asserted, to the subscriber. As above, the availability of additional authenticators provides backup methods of authentication if an authenticator is damaged, lost, or stolen.

Enrollment and binding MAY be broken up into a number of separate physical encounters or electronic transactions. Two electronic transactions are considered to be separate if they are not part of the same protected session. In these cases, the following methods SHALL be used to ensure that the same party acts as applicant throughout the processes:

1. For remote transactions -

a) The applicant SHALL identify themselves in each new transaction by presenting a temporary secret which was established during a prior transaction or encounter, or sent to the applicant's phone number, email address, or postal address of record.

b) Long-term authenticator secrets SHALL only be issued to the applicant within a protected session.

1. For physical transactions -

a) The applicant SHALL identify themselves in person by either using a secret as described above, or through the use of a biometric that was recorded during a prior encounter.

b) Temporary secrets SHALL not be reused.

c) If the CSP issues long-term authenticator secrets during a physical transaction, then they SHALL be loaded locally onto a physical device that is issued in person to the applicant or delivered in a manner that confirms the address of record.

6.1.2. Post-Enrollment Binding

6.1.2.1. Binding of Additional Authenticator at Existing AAL

CSPs and verifiers SHOULD encourage subscribers to maintain at least two valid authenticators of each factor they will be using, with the exception of memorized secrets. For example, a subscriber that usually uses an OTP device as a physical authenticator MAY also be issued a number of look-up secret authenticators, or register a device for out-of-band authentication, in case the physical authenticator is lost, stolen, or damaged.

Accordingly, CSPs SHOULD permit the binding of additional authenticators to a subscriber's account. Before adding the new authenticator, the CSP SHALL first require the subscriber to authenticate at the AAL (or a higher AAL) at which the new authenticator will be used. When an authenticator is added, the CSP SHOULD send a notification to the subscriber. The CSP MAY limit the number of authenticators that may be bound in this manner.

6.1.2.2. Adding an Additional Factor to a One-factor Account

If the subscriber's account has only one authentication factor bound to it (at IAL1/AAL1), and an additional authenticator of a different authentication factor is to be added, the subscriber MAY request that the account be upgraded to AAL2 (but still at IAL1). Once this has been done, the CSP SHALL no longer permit the subscriber to use single-factor authentication.

Prior to binding the new authenticator, the CSP SHALL first require the subscriber to authenticate at AAL1. The CSP SHOULD send a notification of the event to the subscriber.

6.1.2.3. Replacement of Lost Authentication Factor

If a subscriber loses all authenticators of a factor necessary to complete multi-factor authentication and has been identity proofed at IAL2 or IAL3, that subscriber SHALL repeat the identity proofing process. An abbreviated proofing process, confirming the binding of the claimant to previously-supplied evidence MAY be used if the CSP has retained the evidence from the original proofing process pursuant to a privacy risk assessment as described in [SP 800-63A, Section 4.2] (sp800-63a.html#genProofReqs). The CSP SHALL require the claimant to authenticate using an authenticator of the remaining factor, if any, to confirm binding to the existing identity. Reestablishment of authentication factors at IAL3 SHALL be done in person and SHALL verify the biometric collected during the proofing process.

The CSP SHOULD send a notification of the event to the subscriber; this MAY be the same notice as is required as part of the proofing process.

6.1.3. Binding to a Subscriber-provided Authenticator

A subscriber MAY already possess authenticators suitable for authentication at a particular AAL. For example, he or she MAY have a two-factor authenticator from a social network provider, considered AAL2 and IAL1, and would like to use those credentials at an RP that requires IAL2.

CSPs SHOULD, where practical, accommodate the use of subscriber-provided authenticators in order to relieve the burden to the subscriber of managing a large number of authenticators. Binding of these authenticators SHALL be done as described in Section 6.1.2.1.

6.1.4. Renewal

The CSP SHOULD bind an updated authenticator an appropriate amount of time in advance of an existing authenticator's expiration. The process for this SHOULD conform closely to the initial authenticator issuance process (e.g., confirming address of record). Following successful use of the new authenticator, the CSP MAY revoke the authenticator that it is replacing.

6.2. Loss, Theft, Damage, and Unauthorized Duplication

Loss, theft, damage to, and unauthorized duplication of an authenticator are handled similarly, because in most cases one must assume that a lost authenticator has potentially been stolen or compromised by someone that is not the legitimate claimant of the authenticator. Damaged or malfunctioning authenticators SHALL be treated in a similar manner to protect against any possibility of extraction of the authenticator secret. One notable exception is when a memorized secret is forgotten without other indication of having been compromised, such as being duplicated by an attacker.

To facilitate secure reporting of loss or theft of or damage to an authenticator, the CSP SHOULD provide the subscriber a method to authenticate to the CSP using a backup or alternate authenticator; either a memorized secret or a physical authenticator MAY be used for this purpose and only one authentication factor is required. Alternatively, the subscriber MAY establish an authenticated protected channel to the CSP and verify information collected during the proofing process. Alternatively, the CSP MAY verify an address of record (i.e., email, telephone, postal) and suspend authenticator(s) reported to have been compromised. The suspension SHALL be reversible if the subscriber successfully authenticates to the CSP and requests reactivation of an authenticator suspended in this manner.

6.3. 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 give an indication to the subscriber that the authentication failure is due to expiration rather than some other cause.

The CSP SHALL require subscribers to surrender or prove destruction of any physical authenticator containing attribute certificates signed by the CSP as soon as practical after expiration or receipt of a renewed authenticator.

6.4. Revocation and Termination

Revocation of an authenticator (sometimes referred to as termination, especially in the context of PIV credentials) refers to removal of the binding between an authenticator and a credential the CSP maintains. CSPs SHALL revoke the binding of authenticators promptly when an online identity ceases to exist (e.g., subscriber's death, discovery of a fraudulent subscriber), when requested by the subscriber, or when the CSP determines that the subscriber no longer meets its eligibility requirements.

The CSP SHALL require subscribers to surrender or prove destruction of any physical authenticator containing certified attributes signed by the CSP as soon as practical after revocation or termination takes place.

Further requirements on the termination of PIV credentials are found in [FIPS 201].

7. Session Management

Once an authentication event has taken place, it is often desirable to allow the user to continue using the application across multiple subsequent interactions without requiring the user to repeat the authentication event every time. This requirement is particularly true for federation scenarios (described in [SP 800-63C] (sp800-63c.html)), where the authentication event necessarily involves several components and parties coordinating across a network.

To facilitate this behavior, a *session* MAY be started in response to an authentication event, and continue the session until such time that it is terminated. The session MAY be terminated for any number of reasons, including but not limited to an inactivity timeout, an explicit logout event, or other means. The session MAY be continued through a reauthentication event (described in Section 7.2), wherein the user repeats some or all of the initial authentication event, thereby re-establishing the session.

Session management is preferable over continual presentation of credentials as the usability requirements of continual presentation often create incentives for workarounds such as cached unlocking credentials, negating the freshness of the authentication event.

7.1. Session Bindings

A session occurs between the software that a subscriber is running, such as a browser, application, or operating system (the session subject), and the RP or CSP that the subscriber is accessing (the session host). A session secret SHALL be shared between the subscriber's software and the service being accessed. This secret binds the two ends of the session, allowing the user to continue using the service over time. The secret MAY be presented directly by the user’s software (a bearer secret) or possession of the secret MAY be proven using a cryptographic mechanism (a proof of possession secret).

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 which 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.

Secrets used for session binding:

- SHALL be generated by the session host during an interaction, typically immediately following user authentication.
- SHALL be generated by an approved random bit generator and contain at least 64 bits of entropy.
- SHALL be erased or invalidated by the session subject when the user logs out.
- SHOULD be erased on the user endpoint when the user logs out or when the secret is deemed to have expired.
- SHOULD not be placed in insecure locations such as HTML5 Local Storage due to the potential exposure of local storage to XSS attacks.
- SHALL be sent to and received from the device using an authenticated protected channel.
- SHALL time out and not be accepted after the times specified in Sections 4.1.4, 4.2.4, and 4.3.4 (depending on AAL).
- SHALL not be available to insecure communications between the host and user endpoint; authenticated sessions SHALL not fall back to an insecure transport, such as from https to http, following authentication.

URLs or POST content SHALL contain a session identifier that SHALL be verified by the RP to ensure that actions taken outside the session do not affect the protected session.

There are several different mechanisms for managing a session over time. The following sections give three examples, along with additional requirements and considerations particular to each example technology.

7.1.1. Browser Cookies

Browser cookies are the predominant mechanism by which a session will be created and tracked for a user accessing a service.

Cookies:

- SHALL be tagged to be accessible only on secure (HTTPS) sessions.
- SHALL be accessible to the minimum practical set of hostnames and paths.
- SHOULD be tagged to be inaccessible via JavaScript (HttpOnly).
- SHOULD be tagged to expire at or soon after the validity period of the session. This requirement is intended to limit the accumulation of cookies, but SHALL NOT be depended upon to enforce session timeouts.

7.1.2. OAuth Tokens

An OAuth access token is be used to allow an application to access a set of services on behalf of a user following an authentication event. The presence of an OAuth access token SHALL NOT be interpreted by the RP to indicate the presence of the user, in the absence of other signals. The OAuth access token (and any associated refresh tokens) MAY be valid long after the authentication session has ended and the user has left the application.

7.1.3. Device Identification

Other methods of secure device identification, including but not limited to mutual TLS, token binding, or other mechanisms, MAY be used to enact a session between a user and a service.

7.2. Reauthentication

A session SHALL NOT be extended past the guidelines in Sections 4.1.3, 4.2.3, and 4.3.3 (depending on AAL) based on presentation of the session secret alone.

When a session is terminated due to a time-out or other action, the user MAY reauthenticate using their primary authentication mechanism or an appropriate subset thereof, depending on the AAL.

Table 7-1. AAL Reauthentication Requirements

AAL	Requirement
1	Presentation of any one factor

AAL	Requirement
2	Presentation of a memorized secret or biometric
3	Presentation of all factors

> Note: At AAL2, a memorized secret or biometric and not a physical authenticator is required because the session secret is *something you have*, and an additional authentication factor is required to continue the session.

7.2.1. Reauthentication from a federation or assertion

When using a federation protocol (sp800-63c.html#sec4) to connect the CSP and RP, special consideration needs to be made for session management and reauthentication. Both the CSP and RP employ separate session management technologies, and there SHALL NOT be any assumption of correlation between these sessions. Consequently, when a session expires at an RP and reauthentication is required by the RP, it is entirely possible that the session at the CSP is not expired and a new assertion could be generated from this session at the CSP without reauthenticating the user. Therefore, an RP requiring reauthentication through a federation protocol SHALL indicate a maximum acceptable authentication age to the CSP (if possible within the protocol), and the CSP SHALL honor this request (if possible). The CSP in all cases SHALL communicate the primary authentication event time to the RP to allow the RP to decide if the assertion is sufficient for reauthentication or not.

8. Threats and Security Considerations

This section is informative.

8.1. 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. The attacker might guess a memorized secret. Where the authenticator is a shared secret, the attacker could gain access to 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 may 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 might copy a software authenticator. A hardware authenticator might be stolen, tampered with, or duplicated.
- *Something you are* may be replicated. For example, an attacker may obtain a copy of the subscriber's fingerprint and construct a replica.
- *Out of band* secrets may be intercepted. An attacker may receive a challenge or response by eavesdropping on the primary or secondary communications channel. The attacker might then authenticate their own channel or save the message for later replay.

This document assumes that the subscriber is not colluding with the attacker who is attempting to falsely authenticate to the verifier. With this assumption in mind, the threats to the authenticator(s) used for e-authentication are listed in Table 8-1, along with some examples.

Table 8-1 Authenticator Threats

Authenticator Threats/Attacks	Description	Examples
Theft	A physical authenticator is stolen by an Attacker.	A hardware cryptographic device is stolen.
		An OTP device 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.

Authenticator Threats/Attacks	Description	Examples
		Software PKI authenticator (private key) copied.
		Look-up secret authenticator copied.
Eavesdropping	The authenticator secret or authenticator output is revealed to the attacker as the subscriber is authenticating.	Memorized secrets are obtained by watching keyboard entry.
		Memorized secrets or authenticator outputs are intercepted by keystroke logging software.
		A PIN is captured from PIN pad device.
		A hashed password is obtained and used by an attacker for another authentication (*pass-the-hash attack*).
	An out of band secret is intercepted by the attacker by compromising the communication channel.	An out of band secret is transmitted via unencrypted wifi and received by the attacker.
Offline cracking	The authenticator is exposed using analytical methods outside the authentication mechanism.	A software PKI authenticator is subjected to dictionary attack to identify the correct password to use to decrypt the private key.
Side channel attack	The authenticator secret is exposed using 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 response time of the authenticator over a number of attempts.
Phishing or pharming	The authenticator output is captured by fooling the subscriber into thinking the attacker is a verifier or RP.	A password is revealed by subscriber to a website impersonating the verifier.
		A memorized secret is revealed by a bank subscriber in response to an email inquiry from a phisher pretending to represent the bank.
		A memorized secret is revealed by the subscriber at a bogus verifier website reached through DNS spoofing.
Social engineering	The attacker establishes a level of trust with a subscriber in order to convince the subscriber to reveal his or her authenticator secret or authenticator output.	A memorized secret is revealed by the subscriber to an officemate asking for the password on behalf of the subscriber’s boss.
		A memorized secret is revealed by a subscriber in a telephone inquiry from an attacker masquerading as a system administrator.
		An out of band secret sent via SMS is received by an attacker who has convinced the mobile operator to redirect the victim’s mobile phone to the attacker.
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 memorized secrets.
		Online guessing is used to guess authenticator outputs for an OTP device registered to a legitimate claimant.
Endpoint compromise	Malicious code on the endpoint proxies remote access to a connected authenticator without user 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; the attacker uses the secret to authenticate.
	Malicious code on the endpoint compromises a multi-factor software cryptographic authenticator.	Malicious code proxies authentication or exports authenticator keys from the endpoint.

8.2. Threat Mitigation Strategies

Related mechanisms that assist in mitigating the threats identified above are summarized in Table 8-2.

Table 8-2. Mitigating Authenticator Threats

Authenticator Threat/Attack	Threat Mitigation Mechanisms
Theft	Use multi-factor authenticators that need to be activated through a memorized secret or biometric.
Duplication	Use authenticators from which it is difficult to extract and duplicate long-term authentication secrets.
Eavesdropping	Ensure the security of the endpoint, especially with respect to freedom from malware such as key loggers, prior to use.
	Maintain situational awareness when entering memorized secrets and OTPs to ensure that they cannot be observed by others.
	Authenticate over authenticated protected channels (observe lock icon in browser window, for example).
	Use authentication protocols that are resistant to replay attacks such as *pass-the-hash*.
Offline cracking	Use an authenticator with a high entropy authenticator secret.
	Store memorized secrets in a salted, hashed form, including a keyed hash.
Side channel attack	Use authenticator algorithms that are designed to maintain constant power consumption and timing regardless of secret values.
Phishing or pharming	Use authenticators that provide verifier impersonation resistance.
	Be alert for unexpected hostnames in URLs.
	Do not click on links in email messages; instead, enter the URL manually or through a trusted bookmark.
Social engineering	Do not reveal authentication secrets to others, regardless of their story.
	Avoid use of authenticators that present a risk of social engineering of third parties such as customer service agents.
Online guessing	Use authenticators that generate high entropy output.
	Use an authenticator that locks up after a number of repeated failed activation attempts.
Endpoint compromise	Use hardware authenticators that require physical action by the subscriber.
	Provide secure display of identity of verifier and RP.
	Maintain software-based keys in restricted-access storage.

There are several other strategies that may be applied to mitigate the threats described in Table 5:

- *Multiple factors* make successful attacks more difficult to accomplish. If an attacker needs to both steal a cryptographic authenticator and guess a memorized secret, 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 the use of long memorized secrets* that don't 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 subscribers understand when and how to report compromise (or suspicion of compromise) or otherwise recognize patterns of behavior that may signify an attacker attempting to compromise the authentication process.
- *Out of band techniques* may be employed to verify proof of possession of registered devices (e.g., cell phones).

8.3. 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 remaining available to authenticate the subscriber are limited, and economic concerns (e.g., 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, there is also the risk of social engineering attacks.

In order to maintain the integrity of the authentication factors, it is essential that it not be possible to leverage an authentication involving one factor to obtain an authenticator of a different factor. For example, a memorized secret must not be usable to obtain a new list of look-up secrets.

8.4. Session Attacks

The above discussion focuses on threats to the authentication event itself, but hijacking attacks on the session following an authentication event can have similar security impacts. The session management guidelines in Section 7 are essential to maintain session integrity against attacks, such as XSS. In addition, it is important to sanitize all information to be displayed [OWASP-XSS-prevention] to ensure that it does not contain executable content. These guidelines also recommend that session secrets be made inaccessible to mobile code in order to provide extra protection against exfiltration of session secrets should it be possible to inject malicious mobile code.

Another post-authentication threat, CSRF, takes advantage of users' tendency to have multiple sessions active at the same time. It is important to embed and verify a session identifier into web requests to prevent the ability for a valid URL or request to be unintentionally or maliciously activated.

9. Privacy Considerations

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

9.1. Privacy Risk Assessment

Sections 4.1.5, 4.2.5, and 4.3.5 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 a problem did occur.

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

9.2. Privacy Controls

Section 4.4 encourages CSPs to employ appropriately tailored privacy controls. NIST [SP 800-53] provides a set of privacy controls for CSPs to consider when deploying authentication mechanisms. These controls cover notices, redress, and other important considerations for successful and trustworthy deployments.

9.3. Use Limitation

Section 4.4 does not permit the CSP to use information about authenticators that is collected and maintained in the authentication process for any purpose other than authentication or to comply with law or legal process, unless the CSP provides clear notice and obtains consent from the subscriber for additional uses. Care should be taken to ensure that use of such information is limited to its original purpose for collection. Consult your SAOP if there are questions about whether proposed agency uses fall within this scope. As stated in Section 4.4, acceptance by the subscriber of additional uses SHALL NOT be a condition of providing authentication services.

9.4. Agency-specific Privacy Compliance

Section 4.4 covers specific compliance obligations for federal CSPs. It is critical to involve your agency's 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* [Privacy Act] or the *E-Government Act of 2002* [E-Gov] requirement to conduct a PIA. For example, with respect to centralized maintenance of biometrics, it is likely that the Privacy Act requirements will be triggered and require coverage by either a new or existing Privacy Act system of records due to the collection and maintenance of PII and any other attributes 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 cases it will make the most sense to draft a PIA and SORN that encompasses the entire digital authentication process or include the digital authentication process as part of a larger programmatic PIA that discusses the program or benefit the agency is establishing online access to.

Due to the many components of digital authentication, it is important for the SAOP to have an awareness and understanding of each individual component so as to advise appropriately on what compliance requirements apply. Moreover, a thorough understanding of the individual components of digital authentication will enable the SAOP to thoroughly assess and mitigate privacy risks either through compliance processes or by other means.

10. Usability Considerations

This section is informative.

ISO/IEC 9241-11 defines usability as 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.” This definition focuses on users, their goals, and the context of use as key elements necessary for achieving effectiveness, efficiency, and satisfaction. A holistic approach considering these key elements is necessary to achieve usability.

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

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

Integrating usability into the development process can lead to authentication solutions that are secure and usable while addressing users’ authentication needs and organizations’ business goals.

The impact of usability across digital systems needs to be considered as part of the risk assessment when deciding on the AAL requirements. Authenticators with a higher AAL sometimes offer better usability, and should be allowed for use for lower AAL applications.

Leveraging a federation for authentication can alleviate many of the usability issues, though such an approach has its own tradeoffs, as discussed in NIST SP 800-63C, Federation and Assertions (sp800-63c.html).

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

ASSUMPTIONS

In this section, the term “users” means “claimants” or “subscribers.”

Guidelines and considerations are described from the users’ perspective.

Accessibility differs from usability and is out of scope for this document. Section 508 was enacted to eliminate barriers in information technology and require federal agencies to make their electronic and information technology public content accessible to people with disabilities. Refer to Section 508 law and standards for accessibility guidance.

10.1. Usability Considerations Common to Authenticators

When selecting and implementing an authentication system, consider usability across the entire lifecycle of the selected authenticators (i.e., enrollment and distribution, typical use, and intermittent events), while being mindful of the combination of users, their goals, and context of use.

A single authenticator type usually does not suffice for the entire user population. Therefore, whenever possible (based on AAL requirements), support alternative authenticator types and allow users to choose based on their needs. Task immediacy, perceived cost benefit tradeoffs, and unfamiliarity with certain authenticators often impact choice. 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 account and password rather than select an authenticator requiring more steps. Alternatively, users may choose a federated identity option (approved at the appropriate AAL) if they already have an 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 the success of an organization achieving the organization’s desired business outcomes. Therefore, 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 he or she has to track). Single sign-on exemplifies one such minimization strategy.

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

Usability considerations for typical usage of all authenticators include:

- Provide users information on the use and maintenance of the authenticator. For example, instructions for use (especially if there are different requirements for first-time use or initialization), information on authenticator expiration, and what to do if the authenticator is lost or stolen.
- Availability of the authenticator should also be considered; Users need to remember or have their authenticator readily available.
- Whenever possible (based on AAL requirements), users should be provided with alternative authentication options. 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, typically, write for a 6th to 8th grade literacy level.
    - Consider the legibility of user-facing and user-entered text, including font style, size, color, and contrast with surrounding background. Illegible text contributes to user entry errors. To enhance legibility, consider the use of:
        - High contrast, the highest contrast is black on white.
        - Sans serif fonts for electronic displays. Serif fonts for printed materials.
        - Fonts that clearly distinguish between easily confusable characters (such as the capital letter “O” and the number “0”).
        - A minimum font size of 12 points as long as the text fits for display on the device.
- 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 memorized secrets on mobile devices (such as tablets and smartphones) than on traditional desktop computers.
    - Ensure that the time allowed for text entry is adequate (i.e., the entry screen does not time out prematurely).
    - Provide clear, meaningful and actionable feedback on entry errors to reduce user confusion and frustration. Significant usability implications arise when users do not know they have entered text incorrectly.
    - Allow at least 10 entry attempts for authenticators requiring 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 number of remaining allowed attempts. For rate limiting (throttling), inform users how long they have to wait until the next attempt to reduce confusion and frustration.
- 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. The smaller the onscreen keyboard, the more difficult it is to type, 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.

Intermittent events include events such as reauthentication, account lock-out, expiration, revocation, damage, loss, theft, and non-functional software.

Usability considerations for intermittent events across authenticator types include:

- To prevent users from needing to reauthenticate due to user inactivity, prompt users in order to trigger activity just before (e.g., 2 minutes before) an inactivity timeout would otherwise occur.
- Prompt users with adequate time (e.g., 1 hour) to save their work before the fixed periodic reauthentication event required regardless of user activity.
- Clearly communicate how and where to acquire technical assistance. For example, provide users with information such as a link to an online self-service feature, chat sessions or a phone number for help desk support. Ideally, sufficient information can be provided to enable users to recover from intermittent events on their own without outside intervention.

10.2. Usability Considerations by Authenticator Type

In addition to the previously described general usability considerations applicable to most authenticators (Section 10.1), the following sections describe other usability considerations specific to specific authenticator types.

10.2.1. Memorized Secrets

Typical Usage

Users manually input the memorized secret (commonly referred to as a password or PIN).

Usability considerations for typical usage include:

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

Intermittent Events

Usability considerations for intermittent events include: * When users create and change memorized secrets: * Clearly communicate information on how to create and change memorized secrets. * Clearly communicate memorized secret requirements, as specified in Section 5.1.1. * Allow at least 64 characters in length to support the use of passphrases. Encourage users to make memorized secrets as lengthy as they want, using any characters they like (including spaces), thus aiding memorization. * Do not impose other composition rules (e.g. mixtures of different character types) on memorized secrets. * Do not require that memorized secrets be changed arbitrarily (e.g., periodically) unless there is a user request or evidence of authenticator compromise. (See Section 5.1.1 for additional information) * Provide clear, meaningful and actionable feedback when chosen passwords are rejected (e.g., when it appears on a “black list” of unacceptable passwords or has been used previously). Advise users that they need to select a different secret because their previous choice was commonly used.

10.2.2. Look-up Secrets

Typical Usage

Users use the authenticator (physical or electronic record) to look up the appropriate secret(s) 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 prompts’ complexity and size. The larger the subset of secrets a user is prompted to look up, the greater the usability implications are, both the cognitive workload and physical difficulty during entry.

10.2.3. Out of Band

Typical Usage

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

Usability considerations for typical usage:

- Notify users of the receipt of a secret on a locked device. However, if the user locked the out of band device, the user is required to authenticate to the device to access the secret (so that the secret is not displayed on a locked device screen).
- Depending on the implementation, consider form-factor constraints as they are particularly problematic when users must enter text on mobile devices. Providing larger touch areas will improve usability for entering secrets on mobile devices.
- A better usability option is to offer features that do not require text entry on mobile devices (e.g., a single tap on the screen, or a copy feature so users can copy and paste the out of band secrets). Providing users such features is 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 on a smartphone because they must switch back and forth—potentially multiple times—between the out of band application and the primary channel.

10.2.4. Single-factor OTP Device

Typical Usage

Users access the OTP generated by the single-factor OTP device. The authenticator output is typically displayed on the device and the user enters it for the verifier.

Usability considerations for typical usage include:

- Authenticator output allows at least one minute between changes, but ideally allows users the full 2 minutes as specified in the requirement that the nonce be changed at least once every 2 minutes. Users need adequate time to enter the authenticator output (including looking back and forth between the single-factor OTP device and the entry screen).

- Depending on the implementation, the following are additional usability considerations for implementers:
    - If the single-factor OTP device supplies its output via an electronic interface such as USB, this is preferable since users do not have to manually enter the authenticator output. However, if a physical input (such as 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 will be difficult for users to reach.
    - Limited availability of a direct computer interface such as a 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 in order to use the single-factor OTP device.

10.2.5. Multi-factor OTP Device

Typical Usage

Users access the OTP generated by the multi-factor OTP device through a second authentication factor. The OTP is typically displayed on the device and the user manually enters it for the verifier. The second authentication factor may be achieved through some kind of integral entry pad to enter a memorized secret, an integral biometric (e.g., fingerprint) reader or a direct computer interface (e.g., USB port). Usability considerations for the additional factor apply as well (see Section 10.2.1 for memorized secrets and Section 10.4 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. For example, if the authenticator output changes every minute, allow users the full 2 minutes specified in Section 5.1.4.1. Users need adequate time to enter the authenticator output (including looking back and forth between the multi-factor OTP device and the entry screen).
    - Consider form-factor constraints if users must unlock the multi-factor OTP device 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. The smaller the integral entry pad and onscreen keyboard, the more difficult it is to type. Providing larger touch areas improves usability for unlocking the multi-factor OTP device or entering the authenticator output on mobile devices.
    - Limited availability of a direct computer interface like a 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 device.

10.2.6. Single-factor Cryptographic Software

Typical Usage

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

Usability considerations for typical usage include:

- Give cryptographic keys appropriately descriptive names that are meaningful to users since users have to 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 size.

10.2.7. Single-factor Cryptographic Device

Typical Usage

Users authenticate by proving possession of the single-factor cryptographic device.

Usability considerations for typical usage include:

- Requiring a physical input (such as pressing a button) to operate the single-factor cryptographic device could pose usability difficulties. For example, some USB ports are located on the back of computers, making it difficult for users to reach.
- Limited availability of a direct computer interface like a 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 single-factor cryptographic device.

10.2.8. Multi-factor Cryptographic Software

Typical Usage

In order to authenticate, users prove possession and control of the cryptographic key stored on disk or some other "soft" media that requires activation. The activation is through the input of a second authentication factor, either a memorized secret or a biometric; usability considerations for the additional factor apply as well (see Section 10.2.1 for memorized secrets and Section 10.4 for biometrics used in multi-factor authenticators).

Usability considerations for typical usage include:

- Give cryptographic keys appropriately descriptive names that are meaningful to users since users have to 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 size.

10.2.9. Multi-factor Cryptographic Device

Typical Usage

Users authenticate by proving possession of the multi-factor cryptographic device and control of the protected cryptographic key. The device is activated by a second authentication factor, either a memorized secret or a biometric; usability considerations for the additional factor apply as well (see Section 10.2.1 for memorized secrets and Section 10.4 for biometrics used in multi-factor authenticators).

Usability considerations for typical usage include:

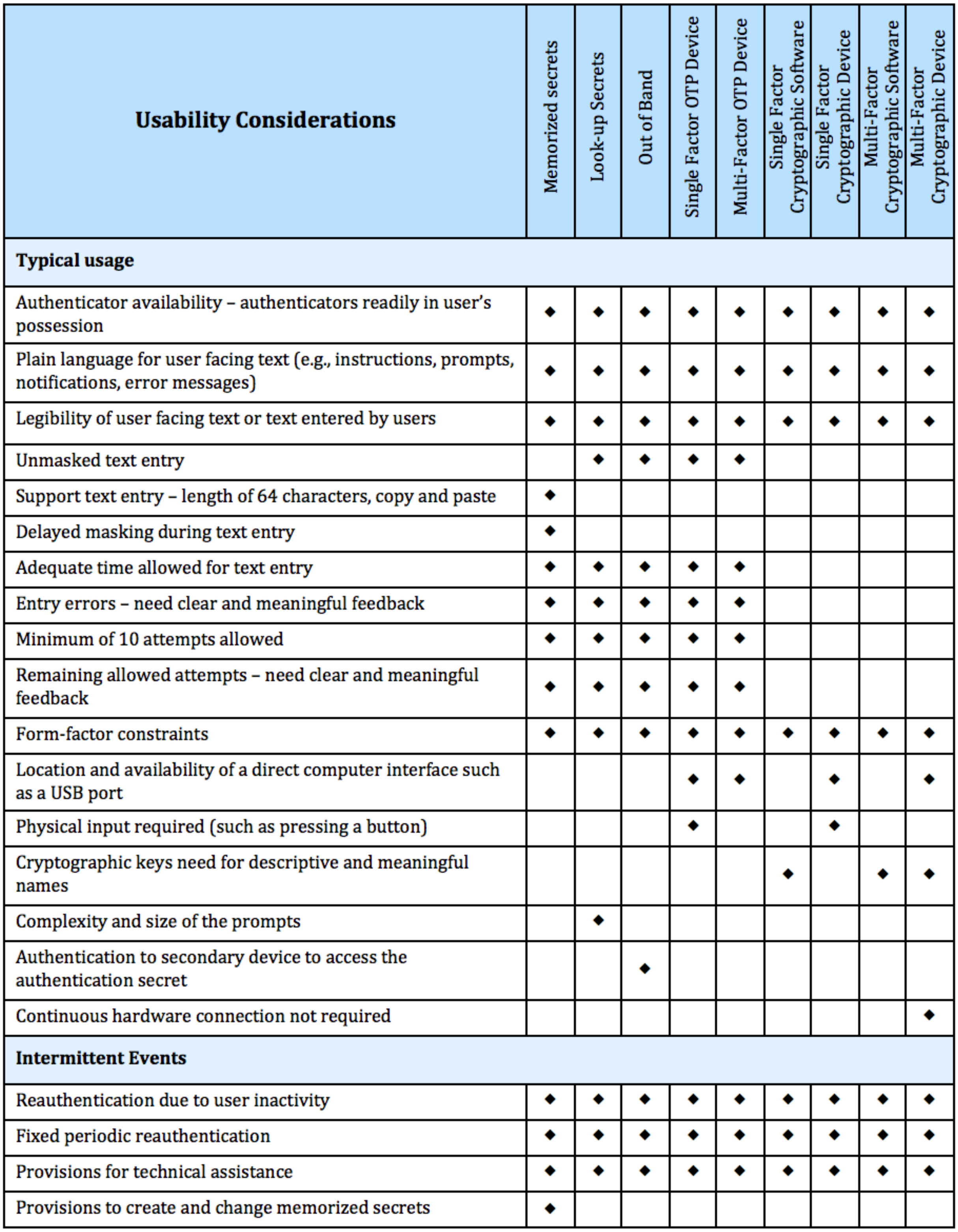
- Do not require users to keep multi-factor cryptographic devices connected following authentication. Users may forget to disconnect the multi-factor cryptographic device when they are done with it (e.g., forgetting a smartcard in the smartcard reader and walking away from the computer).
    - Users need to be informed regarding whether the multi-factor cryptographic device is required to stay connected or not.
- Give cryptographic keys appropriately descriptive names that are meaningful to users since users have to recognize and recall which cryptographic key to use for which authentication task. This prevents users being faced with multiple similarly and ambiguously named cryptographic keys. Selecting from multiple cryptographic keys on smaller mobile devices (such as smartphones) may be particularly problematic if the names of the cryptographic keys are shortened due to reduced screen size.
- Limited availability of a direct computer interface like a 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 cryptographic device.

10.3. Summary of Usability Considerations

Table 10-1 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 users' goals and context of use, users may value certain usability attributes over others. Whenever possible, provide alternative authenticator types and allow users to choose between them.

It is important to note that multi-factor authenticators (e.g., multi-factor OTP devices, multi-factor cryptographic software, and multi-factor cryptographic devices) also inherit their secondary 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 Table 10-1 and are discussed in Section 10.4.

Table 10-1 Usability Considerations Summary by Authenticator Type



Usability Considerations	Memorized secrets	Look-up Secrets	Out of Band	Single Factor OTP Device	Multi-Factor OTP Device	Single Factor Cryptographic Software	Single Factor Cryptographic Device	Multi-Factor Cryptographic Software	Multi-Factor Cryptographic Device
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 memorized secrets	◆								

10.4. Biometrics Usability Considerations

This section provides a high-level overview of biometrics general usability considerations. For a more detailed discussion of biometric usability, see *Usability & Biometrics, Ensuring Successful Biometric Systems* [NIST Usability].

Although there are other biometric modalities, the following three biometric modalities are more commonly used for authentication: fingerprint, face and iris.

Typical Usage

- For all modalities, user familiarity and practice with the device improves performance.

- 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 they are most comfortable with for their second authentication factor. The user population may be more comfortable and familiar with—and accepting of—some biometric modalities than others.
- User experience with biometrics as an activation factor.
- Provide clear, meaningful feedback on number of remaining allowed attempts. For example, for rate limiting (throttling), inform users of the time period they have to wait until next attempt to reduce user confusion and frustration.
- Fingerprint Usability Considerations:
    - Users have to remember which finger(s) they used for initial enrollment.
    - The amount of moisture on the finger(s) affects the sensor's ability for successful capture.
    - Additional factors influencing fingerprint capture quality include age, gender, and occupation (e.g., users handling chemicals or working extensively with their hands may have degraded friction ridges).
- Face Usability Considerations:
    - Users have to remember whether they wore any artifacts, such as glasses, during enrollment because it affects facial recognition accuracy.
    - Differences in environmental lighting conditions can affect facial recognition accuracy.
    - Facial poses affect facial recognition accuracy (e.g., smiling versus neutral expression).
- Iris Usability Considerations:
    - Users wearing colored contacts may affect the iris recognition accuracy.
    - Users who have had eye surgery may need to re-enroll post-surgery.
    - Differences in environmental lighting conditions can affect iris recognition accuracy, especially for certain iris colors.

Intermittent Events

As 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 with biometrics use include, but are not limited to, the following, which may affect recognition accuracy:

- If users injure their enrolled finger(s), fingerprint recognition may not work. Fingerprint authentication will be difficult for users with degraded fingerprints.
- The time elapsed between the time of facial recognition for authentication and the time of the initial enrollment can affect recognition accuracy as a user's face changes naturally over time. A user's weight change may also be a factor.
- Iris recognition may not work for people who had eye surgery, unless they re-enroll.

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

- An alternative authentication method must be available and functioning. In cases where biometrics do not work, allow users to use a memorized secret as an alternative second factor.
- Provisions for technical assistance:
    - Clearly communicate information on how and where to acquire technical assistance. For example, provide users information such as a link to online self-service feature and 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).

11. References

This section is informative.

To be completed

11.1. General References

[OMB M-03-22] OMB Memorandum M-03-22, *OMB Guidance for Implementing the Privacy Provisions of the E-Government Act of 2002*, September 26, 2003, available at: http://www.whitehouse.gov/omb/memoranda/m03-22.html (http://www.whitehouse.gov/omb/memoranda/m03-22.html).

[M-04-04] OMB Memorandum M-04-04, E-Authentication Guidance for Federal Agencies, December 16, 2003, available at: https://www.whitehouse.gov/sites/default/files/omb/memoranda/fy04/m04-04.pdf (https://www.whitehouse.gov/sites/default/files/omb/memoranda/fy04/m04-04.pdf).

[RFC 20] Cerf, V., *ASCII format for network interchange*, STD 80, RFC 20, DOI 10.17487/RFC0020, October 1969, http://www.rfc-editor.org/info/rfc20 (http://www.rfc-editor.org/info/rfc20).

[RFC 5246] IETF, *The Transport Layer Security (TLS) Protocol Version 1.2*, available at: https://tools.ietf.org/html/rfc5246/ (https://tools.ietf.org/html/rfc5246/).

[RFC 5280] IETF, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, available at: https://tools.ietf.org/html/rfc5280/ (https://tools.ietf.org/html/rfc5280/).

[RFC 6960] IETF, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*, available at: https://tools.ietf.org/html/rfc6960/ (https://tools.ietf.org/html/rfc6960/).

[BCP 195] Sheffer, Y., Holz, R., and P. Saint-Andre, *Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)*, BCP 195, RFC 7525, May 2015, http://www.rfc-editor.org/info/bcp195 (http://www.rfc-editor.org/info/bcp195).

[ICAM] National Security Systems and Identity, Credential and Access Management Sub-Committee Focus Group, Federal CIO Council, *ICAM Lexicon*, Version 0.5, March 2011.

[ISO/IEC 10646] International Standards Organization, *Universal Coded Character Set*, 2014, available at: http://standards.iso.org/ittf/PubliclyAvailableStandards/c063182_ISO_IEC_10646_2014.zip (http://standards.iso.org/ittf/PubliclyAvailableStandards/c063182_ISO_IEC_10646_2014.zip).

[ISO/IEC 24745] International Standards Organization, *Information technology – Security techniques – Biometric information protection*, 2011, available at: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=52946 (http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=52946).

[OWASP-XSS-prevention] Open Web Application Security Project, *XSS (Cross Site Scripting) Prevention Cheat Sheet*, available at: https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet (https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet).

[ISO 9241-11] International Organization for Standardization, *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on Usability*, ISO 9241-11:1998, ISO: Geneva, Switzerland, 1998.

[Section 508] Section 508 Law and Related Laws and Policies (January 30, 2017), available at: https://www.section508.gov/content/learn/laws-and-policies (https://www.section508.gov/content/learn/laws-and-policies).

[NIST Usability] National Institute and Standards and Technology, *Usability & Biometrics, Ensuring Successful Biometric Systems*, June 11, 2008, available at: http://www.nist.gov/customcf/get_pdf.cfm?pub_id=152184 (http://www.nist.gov/customcf/get_pdf.cfm?pub_id=152184).

[UAX 15] Unicode Consortium, *Unicode Normalization Forms*, Unicode Standard Annex 15, Version 9.0.0, February, 2016, available at: http://www.unicode.org/reports/tr15/ (http://www.unicode.org/reports/tr15/).

[Shannon] Shannon, Claude E. “A Mathematical Theory of Communication,” *Bell System Technical Journal*, v. 27, pp. 379-423, 623-656, July, October, 1948.

11.2. NIST Special Publications

NIST 800 Series Special Publications are available at: http://csrc.nist.gov/publications/nistpubs/index.html (http://csrc.nist.gov/publications/nistpubs/index.html). The following publications may be of particular interest to those implementing systems of applications requiring e-authentication.

[SP 800-52] NIST Special Publication 800-52, Revision 1, *Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations*, April, 2014.

[SP 800-53] NIST Special Publication 800-53, Revision 4, *Recommended Security and Privacy Controls for Federal Information Systems and Organizations*, August 2013 and Errata as of January 2015.

[SP 800-57 Part 1] NIST Special Publication 800-57 Part 1, Revision 4, *Recommendation for Key Management, Part 1: General*, January 2016.

[SP 800-63C] NIST Special Publication 800-63C, *Assertions and Federation*. ***To be updated at publication***

[SP 800-131A] NIST Special Publication 800-131A , *Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, Revision 1, November 2015.

[SP 800-132] NIST Special Publication 800-132, *Recommendation for Password-Based Key Derivation*, December 2010.

11.3. Federal Information Processing Standards

FIPS can be found at: http://csrc.nist.gov/publications/fips/.

[FIPS 140-2] Federal Information Processing Standard Publication 140-2, *Security Requirements for Cryptographic Modules*, May 25, 2001.

[FIPS 198-1] Federal Information Processing Standard Publication 198-1, *The Keyed-Hash Message Authentication Code (HMAC)*, July 2008.

[FIPS 201] Federal Information Processing Standard Publication 201-2, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, August 2013, available at: http://dx.doi.org/10.6028/NIST.FIPS.201-2 (http://dx.doi.org/10.6028/NIST.FIPS.201-2).

11.4. Legislation

[Privacy Act] *Privacy Act of 1974* (P.L. 93-579), December 1974, available at: http://www.justice.gov/opcl/privacy-act-1974 (http://www.justice.gov/opcl/privacy-act-1974).

[E-Gov] *E-Government Act* [includes FISMA] (P.L. 107-347), December 2002, available at: http://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf (http://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf).

Appendix A: Strength of Memorized Secrets

This appendix is informative.

A.1. Introduction

Despite widespread frustration with the use of passwords from both a usability and security standpoint, they remain a very widely used form of authentication. Humans, however, have only 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 in an effort to increase the complexity of these memorized secrets. The most notable form of these is composition rules, which require the user to choose passwords constructed using a mix of character types, such as at least one digit, uppercase letter, and symbol. However, analyses of breached password databases reveals that the benefit of such rules is not nearly as significant as initially thought, although the impact on usability and memorability is severe.

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 having deterministic distribution functions, estimating the entropy for user-chosen passwords is difficult and past efforts to do so have not been particularly accurate. For this reason, a different and somewhat simpler approach, based primarily on password length, is presented herein.

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

A.2. Length

Password length has been found to be the primary factor in characterizing password strength. Passwords that are too short yield to brute force attacks as well as to dictionary attacks using words and commonly chosen passwords.

The minimum password length that should be required depends to a large extent on the threat model being addressed. Online attacks where the attacker attempts to log in by guessing the password can be mitigated by throttling the rate of login attempts permitted. In order to prevent an attacker (or a persistent claimant with poor typing skills) from easily inflicting a denial-of-service attack on the subscriber by making many incorrect guesses, passwords need to be complex enough that throttling does not occur after a modest number of erroneous attempts, but does occur before there is a significant chance of a successful guess.

Offline attacks are sometimes possible when one or more hashed passwords is obtained by the attacker through a database breach. The ability of the attacker to determine one or more users' passwords depends on the way in which 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 with no throttling requires passwords intended to resist such attacks to be orders of magnitude more complex than those that are 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 not to permit the use of lengthy passwords (or pass phrases) if the user wishes. Extremely long passwords (perhaps megabytes in length) could conceivably require excessive processing time to hash, so it is reasonable to have some limit.

A.3. Complexity

As noted above, composition rules are commonly used in an attempt to increase the difficulty of guessing user-chosen passwords. Research has shown, however, that users respond in very predictable ways to the requirements imposed by composition rules. For example, a user that 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 attempts to create complex passwords are rejected by online services. Many services reject passwords with spaces and various special characters. In some cases the special characters that are not accepted might be an effort to avoid attacks like SQL injection that depend on those characters. But a properly hashed password would not be sent intact to a database in any case, so such precautions are unnecessary. Users should also be able to include space characters to allow the use of phrases. Spaces themselves, however, add little to the complexity of passwords and may introduce usability issues (e.g., the undetected use of two spaces rather than one), so it may be beneficial to remove spaces in typed passwords prior to verification.

Users’ password choices are very predictable, so attackers are likely to guess passwords that have been successful in the past. These include dictionary words and passwords from previous breaches, such as the “Password1!” example above. For this reason, it is recommended that passwords chosen by users be compared against a “black list” of unacceptable passwords. This list should include passwords from previous breach corpuses, dictionary words, and specific words (such as the name of the service itself) that users are likely to choose. Since user choice of passwords will also be governed by a minimum length requirement, this dictionary need only include entries meeting that requirement.

A.4. Randomly-chosen Secrets

Another factor that determines the strength of memorized secrets is the process by which they are generated. Secrets that are randomly chosen (in most cases by the verifier or CSP) and are uniformly distributed will be more difficult to guess or brute-force attack than user-chosen secrets meeting the same length and complexity requirements. Accordingly, at LOA2, SP 800-63-2 permitted the use of randomly generated PINs with 6 or more digits while requiring user-chosen memorized secrets to be a minimum of 8 characters long.

As discussed above, the threat model being addressed with memorized secret length requirements includes rate-limited online attacks, but not offline attacks. With this limitation, 6 digit randomly-generated PINs are still considered adequate for memorized secrets.

A.5. Summary

Length and complexity requirements beyond those recommended here significantly increase the difficulty of memorized secrets and increase user frustration. As a result, users often work around these restrictions in a way that is counterproductive. Furthermore, other mitigations such as blacklists, secure hashed storage, and rate throttling are more effective at preventing modern brute-force attacks. Therefore, no additional complexity requirements are imposed.