Draft NISTIR 8344

Ontology for Authentication

Kim Schaffer

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8344-draft

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

Draft NISTIR 8344

Ontology for Authentication

Kim Schaffer
Computer Security Division
Information Technology Laboratory

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8344-draft

February 2021

U.S. Department of Commerce
Wynn Coggins, Acting Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

National Institute of Standards and Technology Interagency or Internal Report 8344
47 pages (February 2021)

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8344-draft

Public comment period: *February 8, 2021* through *April 9, 2021*

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive (Mail Stop 8930) Gaithersburg, MD 20899-8930
Email: NISTIR-8344-comments@nist.gov

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

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.

Abstract

Authentication appears to be headed into crisis with the difficulties of passwords, the need for derived credentials, and the uncertainty of quantum processing, mobile platforms, and the Internet of Things. The establishment of an ontology of authentication can better manage the requirements placed upon both systems and users. This document includes a survey of authentication mechanisms, establishing the need and basis for authentication metrology, as well as key factors in determining strength and management requirements when assessing an authentication system in a given environment.

Keywords

IAA process; attestation; authentication; confirmation; continuous authentication; measurement; ontology; static authentication; usability.

Acknowledgements

The efforts of Mary Theofanos to inform and educate the author concerning the insertion of Usability into Authentication are greatly appreciated.

Document Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in Request for Comment (RFC) 2119[1]. When these words appear in regular case, such as “should” or “may”, they are not intended to be interpreted as RFC 2119 key words.

Call for Patent Claims

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

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

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

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

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

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

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

Such statements should be addressed to NISTIR-8344-comments@nist.gov with the Subject: "Draft NISTIR 8344 Call for Patent Claims"

Executive Summary

This document is intended for anyone who must implement or manage the authentication component of an identity management, authentication, and authorization (IAA) or attestation process. A better understanding of these general processes can improve future development of authorization components and interoperation with identity management and authentication. This document is not meant to replace authentication-related standards but to provide an understanding of authentication in general. Additionally, it may help future authentication standards development in using a common framework.

This document recommends an authentication *ontology*—associations and relationships common to all methodologies meant to verify a construct previously associated with an entity or object. The document begins with how entity authentication fits into the *IAA process* and how it relates to the other components of that process. A taxonomy of authentication is presented for both entity- and object-focused authentications. Entity authentication is given the term confirmation and is broken into three areas: human-machine authentication, machine-machine authentication, and human-human authentication. The authentication of objects, given the term attestation, is then presented. Following the discussion of the taxonomy, authentication attributes are presented along with one of the most debated aspects of authentication—strength. Addressing the need to definitively measure authentication strength, four areas are identified: security, usability, deployability, and manageability. For each area, a set of environmental factors suitable for measurement are discussed. Figure 1 provides a concept map of the ontology.

Human-machine authentication takes up much of this document due to the number and complexities of this type of interface. Social environment and individuals' limitations put severe constraints on human-machine authentication mechanisms. As such, much more work continues to be done to try and bridge the gap between security and usability. To state the issue another way, there appears to be a relation between how much is asked of the operator and how willing the operator is to support security rather than (mis)manage it.

Table of Contents

List of Figures

List of Tables

1 Introduction

Authentication has been in existence since man started living in groups or tribes: a symbol, a secret word, or handshake provided a means to prove membership or hierarchy within the membership. Now, digital forms of authentication have become increasingly complex, driving the need to better understand what purpose authentication is attempting to fulfill and the components necessary for successful authentication. While there are many existing standards that focus on a specific method, this document addresses the overarching topic of authentication.

This document represents the result of an effort to define authentication by examining mechanisms used to prove position or membership; analyzing existing methods, tools, and techniques; and developing an abstract representation of authentication features and services. Basic mechanisms used to accomplish authentication are identified and discussed in general terms. While most authentication mechanisms utilize cryptography, specific implementations of the cryptography are left to standards that address the authentication mechanism and are not included in this document.

A high-level discussion of business processes for implementing an authentication system is included. Authentication impacts several different areas of an organization, especially policy generation and coordination, and is often not addressed in standards that focus on a specific mechanism. A common set of measurements that pertain to all authentication mechanisms includes:

- The uniqueness of the hardware, software, or processes that represent the entity to the entity being authenticated
- The resistance of the representation to being duplicated or otherwise compromised
- The protection of the representation during delivery to the validating mechanism and the protection of the mechanism containing the *authentication reference*
- The usability of human-machine authentication

Management considerations for establishing or replacing an authentication scheme are identified. These attempt to characterize the proposed and existing environment to identify a reasonable *authentication scheme*.

Authentication is the component of the IAA process that provides a degree of assurance that the entity's assigned identity is verified. Understanding the process of properly gaining access to a system is often complicated by the inconsistent use of the terminology. Section 4.1 is an overview of the IAA process.

2 The Authentication Ontology

This document proposes an overarching *ontology* of authentication. The concept map shown in Figure 1 identifies key factors observed from assessing authentication methodologies. Some aspects of the ontology are hierarchical or structural in nature, such as the taxonomy of authentication mechanisms provided in Figure 2. There are also several items in an ontology that may not be relational in nature; the structure is either not known or not well-defined. Relational examples include trust and the strength of authentication mechanisms. Today, strength often has a relative magnitude or structure. Similarly, only a rough overview of authentication management can be provided, as the environment is a critical element for a successful implementation.

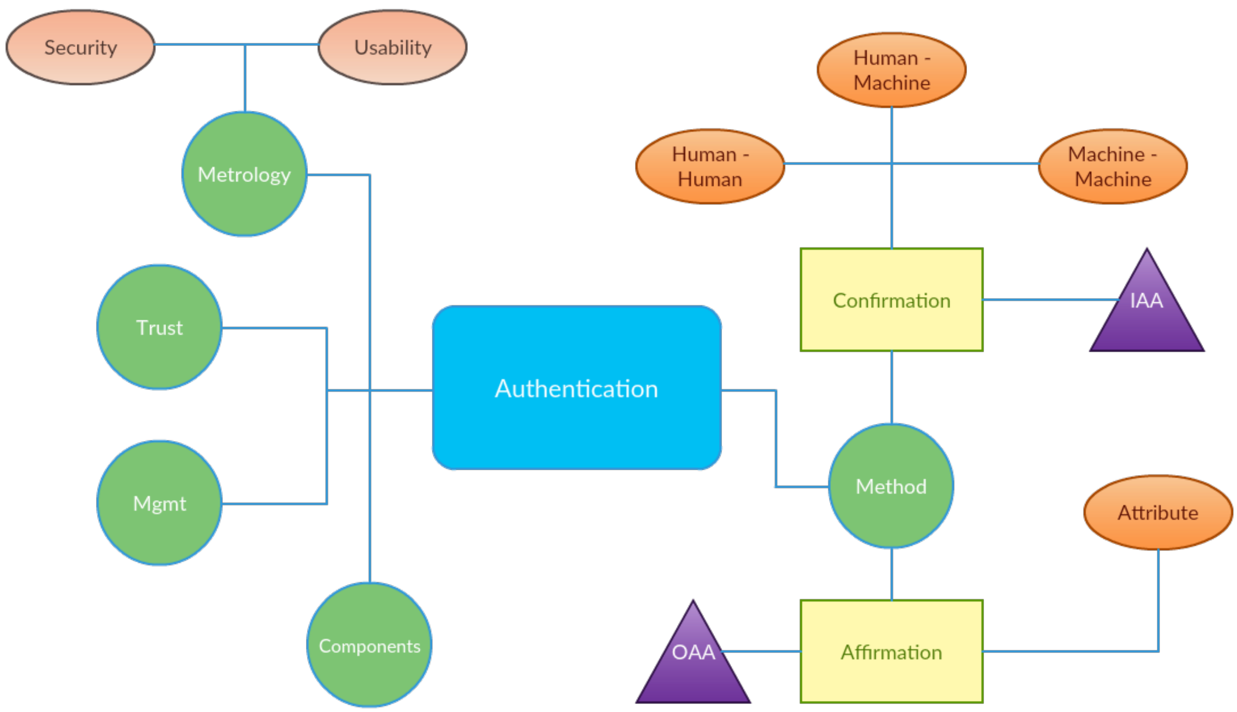


Figure 1 - Concept Map for Authentication Properties

The management of authentication includes the relationship between identity management (IM) and authorization. The development, implementation, maintenance, and operation of an authentication site have both structural and relational aspects. As authentication becomes better understood, these aspects can be described in more detail.

Little guidance can be found for determining the criteria for selecting authentication mechanisms. As an example, FIPS 140-2, which is being used through 2025, discusses authentication strength by simply stating that "the probability shall be less than one in 1,000,000 that a random attempt will succeed…(e.g., guessing a password or PIN, false acceptance error rate of a biometric device, or some combination of authentication methods)." and that multiple attempts in a one-minute period should have a probability of success of less than one in 100,000 [2]. Similarly, FIPS 140-2 minimally addresses usability by stating that feedback to an operator should not provide any information that would weaken the strength of the authentication. While

FIPS 140-2 has recently been updated, FIPS 140-3 leaves these types of requirements to the validation authority.

Providing guidance across different mechanisms is difficult because comparisons across different mechanisms are difficult; implementation paradigms vary, and assessing strengths vary. For example, comparing the randomness of passwords with the error rates of biometrics and the key lengths of PKI solutions is subjective at best. It could be argued that much of the authentication mechanisms were selected by policy or historical precedence. While this is likely to continue for many authentication systems in the short-term, it is hoped that confidence can be gained in assessing the impact of all aspects of authentication. As authentication schemes become more sophisticated, identifying these factors can aid in achieving usable and secure systems. As technologies mature, authentication systems may no longer support the increasing requirements, and alternatives must be evaluated.

To understand this ontology, it is best to consider the authentication mechanisms examined. The taxonomy groups certain mechanisms according to their similarities and aid in the understanding of further properties identified from this study. The next section covers the taxonomy of authentication.

3 A Taxonomy of Authentication Mechanisms

The plethora of authentication mechanisms can be overwhelming. By grouping similar uses into a hierarchy, it becomes possible to create a taxonomy. An authentication mechanism taxonomy provides a structure to categorize different but related types of authentication mechanisms. This document proposes a taxonomy that is composed of two major classes of authentication: confirmation and attestation. Confirmation is generally used as verification of an entity to manage permissions or access. Attestation is generally the verification of a direct or indirect attribute of the object (not entity) of interest.

Further analysis has led to the creation of three domains under the confirmation class: human-machine (e.g., a human user authenticating on a device), machine-machine (e.g., an automated corporate internet access), and human-human authentication (e.g., in-person password recovery). Human-machine and machine-machine have been extensively discussed and researched in multiple arenas. However, while human-human methods have been popular options for authentication recovery, they are difficult to automate and are often considered susceptible to social engineering.

Attestation is the second class of authentication. The purpose of attestation is to verify the object rather than use the object to verify the entity it represents. Attestation is used on objects from digital and physical watermarking to digital signatures. This class of authentication has a wide range of assurance goals, from indications that an object was not changed to preventing duplication. Currently there is only one domain for attestation: attribute.

Figure 2 presents the current structure of the authentication taxonomy with the classes of confirmation and attestation, as well as the domains human-machine, machine-machine, human-human, and attribute. Examples of mechanisms for each family under the domains are presented. It is expected that there will be a great deal more structure as individual mechanisms are identified and added.

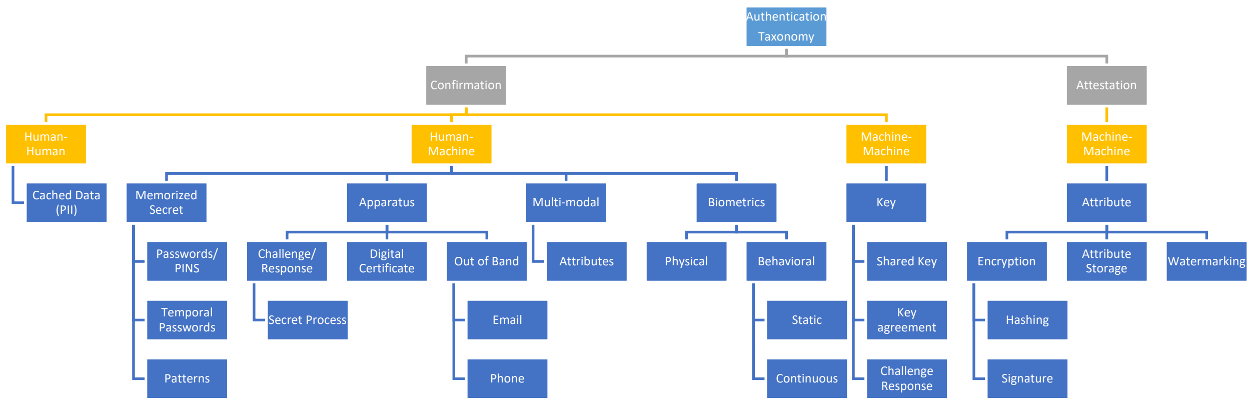


Figure 2 - Authentication taxonomy

3.1 Class: Confirmation

The first of the two currently identified classes is confirmation. The authentication mechanism confirms that the provided hardware, software, or process representing the entity is valid for access. This taxonomy was created using existing standards and technologies. The structure was developed based on commonalities in the use of the mechanisms. There are currently three domains under the class confirmation: human-machine, machine-machine, and human-human. The remaining paragraphs of this section focus on a basic understanding of the different mechanisms for human-machine (Section 3.1.2), machine-machine (Section 3.1.3), and human-human (Section 3.1.4). The other class—attestation—is discussed in Section 3.2.

3.1.1 Confirmation domains

The confirmation class authenticates an entity that is typically represented by one but sometimes a group of entities. Human interaction is a strong component of confirmation; two of the three domains are dependent on aspects of human capabilities or physiology. The authentication that is best known by the public is a human interacting with some interface or sensor that allows access by an individual. This domain is human-machine.

For a connection resulting from a human-machine authentication to be successful, the entity often crosses several boundaries. Authentication mechanisms are often necessary to support connections across and within each layer of the Open Systems Interconnection (OSI) model. Even staying within TCP/IP communications, authentications have optimized for and across layers of abstractions, such as those presented in Figure 3 below.

While authentication technology is not restricted to IP communications, it is worthwhile to demonstrate some of the applications of authentication using IP networks. Figure 3 demonstrates the common IP hierarchy of modern computing. The machine-machine authentication technology often gates the interface of different communication layers. The application layer is typically within a single system and often requires login at the console level as a minimum. The user login at the console is managed by the administrator of the system, though it may also require the permissions of the internal network through the Active Directory or similar.

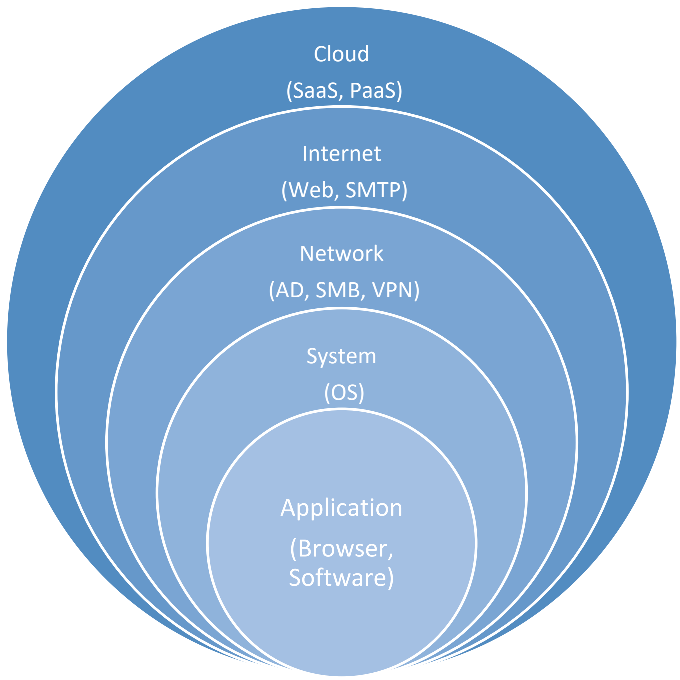


Figure 3 - Authentication Implementation Complexity (not user experience)

With the increase in outsourcing web services, many enterprises look to the internet for corporate services. When using web services under the control of a provider, the user and corporate entities must agree to the provider's policies. However, cloud services may provide platforms, services, and applications while being closely tied to each corporate policy they serve. This is the domain of machine-machine confirmation authentication.

A user will typically consider authenticating to a website from an enterprise network to be a simple authentication process. However, Figure 4 demonstrates the complexities in interweaving human-machine and machine-machine authentications, including the options for single sign-on for services that may support the enterprise outside of the network.

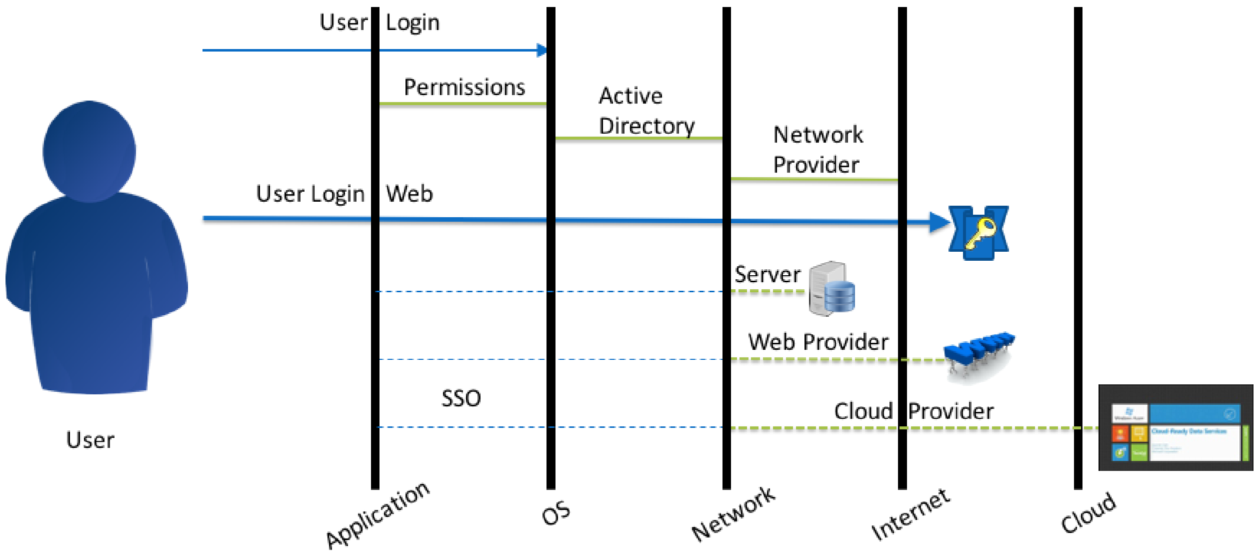


Figure 4 - Human-Machine and Machine-Machine Resources

The last domain is usually the least considered but most expensive to manage. Human-human authentication is often used as a last resort after human-machine has failed. Hackers have been known to purposely lock a human-machine authentication account to try to manipulate administrators who support human-human authentication into giving the hacker access to the account.

3.1.2 Domain: Human-Machine

Human-machine authentication is one of the most difficult interactions to address, and the difficulty is often attributed to the differences in the capabilities between humans and machines. Initially, human-machine authentication was primarily for billing purposes on shared mainframe computers. However, as public access to computers has become more prevalent, stronger authentication requirements for human-machine interactions have become necessary. While humans have a large range of capabilities, they also appear to be limited in remembering specific information (e.g., keys, passwords of sufficient strength for today's requirements), especially for the multiple systems with which they interact on a daily to yearly basis. Much work has gone into establishing and optimizing these authentication mechanisms and the supporting systems.

In the human-machine domain, a human is in control of the hardware, software, or process that represents the entity. To accommodate the multitude of differing mechanisms, human-machine authentication has been further divided into initial, multi-modal, and continuous. Most of today's authentication mechanisms are considered a type of initial authentication mechanism, which responds with a single response (i.e., yes or no). Three major categories of initial authentication mechanisms currently used today include passwords, dedicated authentication devices, and biometrics, with their usage as primarily one time per session. Continuous authentication is currently rare in today's environment, but it holds much promise. It uses a mechanism that is often based on behavioral biometrics used in a continuously sampling mode. The final subdomain of human-machine authentication, multi-modal, is any combination of initial and/or continuous authentication. While an easy concept to describe, it can be very difficult to integrate, support, and assess.

It is worth noting that in cases where the user is asked to authenticate for a set of services under a central administration, a caching scheme is used by the administration for the user. Once the user successfully authenticates, the authentication mechanism may cache alternate credentials to alleviate the burden of authenticating to each system when the level of risk is expected to be sufficiently low. In these cases, it is addressed as a machine-machine authentication that is representing the human in place of a human-machine authentication. This cached authentication is discussed in this document under machine-machine authentication (Section 3.1.3) as it is an automated authentication.

3.1.2.1 Family: Memorized Secret

The most generic definition of memorized secret is “something you know” that is shared with only the machine confirming the user. While there are several different forms of memorized secrets—including password, personal identification number (PIN), picture, and sound—they are all used to demonstrate the user’s knowledge of the secret information to be shared only with the authentication server. Many popular articles have called for the death of passwords, yet passwords remain the most used form of authentication and are often favored as an additional or alternative form, such as to unlock a smartcard or as a backup means of authentication.

A guide for enterprise password management is available and addresses common defense mechanisms against threats for enterprise password mechanisms. It also outlines possible defenses against these threats, including single sign-on solutions and password management aids that may be permitted. Organizations that use memorized secrets for authentication often follow the latest trends without assessing the usability, making the selection and use of memorized secrets difficult if not onerous.

Personal information

Cognitive passwords are sometimes used as a secondary or backup authentication mechanism. The interface presents previously answered and often commonly asked questions that could easily be recalled and answered from memory. As an alternative, the server may query the user to select multiple choice questions based on historical, publicly available records to supplement proof of identity as a form of authentication. However, this has the negative side effect of collecting additional privacy information, which is typically considered to be of low value.

3.1.2.2 Family: Biometric

Authentication based on “something you are” often refers to biometric authentication. Common examples include fingerprint, facial, iris, and voice recognition. Biometrics used in initial authentication make a one-time determination as to the confidence that the active scan and the biometric data collected prior to authentication are from the same user. Biometrics that continuously scan and determine the level of confidence that the right person continues to use the system are forms of continuous authentication.

There continue to be many attacks and countermeasures for biometrics as the field matures. A biometric typically creates a template that encapsulates the minutia of the object into a hardware, software, or process that represents the entity, which is compared to a reference. While a single sample using a given template may be compromised, it typically does not compromise the biometric object from future use for other templates. An example of NIST recommendations for the use of biometrics in authentication mechanisms is SP 800-76-2[4].

3.1.2.2.1 Category: Initial

Currently, the most common human-machine authentication is initial authentication. Initial authentication quickly validates a credential (such as a fingerprint) that the user has previously provided so that authorization can allow the user to access the requested information or functionality. Once initial authentication is completed, the connection remains until broken by the user or another monitoring mechanism.

3.1.2.2.2 Category: Continuous

Occasionally, users intentionally or accidentally leave the access open and available to others. Several timing-based applications or other dedicated hardware attempt to minimize this exposure. Research has focused on mechanisms that would continuously sample (usually a form of biometrics) user activity and periodically report a confidence factor as to whether the correct user is still using the system. As the factor reaches a predetermined threshold, the user is authenticated for some span of time, more closely tying the authentication to the user. However, these continuous authentication mechanisms are often limited in their use due to the non-uniformity of the users (e.g., mental or physical limitations or changes). To address these issues, multiple authentication mechanisms, or multi-modal mechanisms, are being investigated for use.

Behavioral Biometrics

Behavioral biometrics continuously assess the user by monitoring some activity of the user, such as typing, while analyzing aspects of the typing to make sure the operator has not changed. Unlike initial authentication, continuous authentication repeatedly assesses the current user for activity and identity. Cognitive biometrics can be considered a form of behavioral biometrics that focuses on the analysis of the emanations of the brain. It may be used directly or through a translator, depending on the biometric modality. Cognitive biometrics interprets biometric data into human action, such as something heard or visualized. An example of this is electromagnetic sampling of brain activity into actions such as “virtual” movement or speech, adding a truly dynamic aspect to authentication.

3.1.2.3 Family: Apparatus

An authentication apparatus is often considered to be “something you have” and may include time- or event-based changing PINs or passwords in hardware devices, smartcards, or RFID-based devices. A common weakness is that it is relatively easy to lose the device. This is typically countered by the use of an additional authentication mechanism, such as PINs, bundled into a stronger solution. Challenge response and signature verification protocols are two methods that are often utilized for strong solutions.

Software forms of these methods are also available, though they may be considered weaker solutions. For example, a smartcard might support a PKI infrastructure and is typically considered one of the strongest forms of authentication. Related functionality can be found in software such as a web browser using SSL, though it is typically not considered to be as secure as a hardware embodiment.

Devices such as cell phones are sometimes used as a secondary authentication mechanism. However, this is more of an out-of-band authentication source than a strong authentication token. Though seldom used now, memory devices were popular. The memory device either stored a token (such as a password) or could process a simple algorithm. The physical embodiment made it difficult for attackers to replicate the device, but it would not necessarily resist sophisticated assessment techniques. Memory devices appear to be increasingly more difficult to find.

It should be noted that hardware devices acting for the validation server are not considered to be a user authenticator for this taxonomy.

3.1.2.4 Family: Multi-Modal

Multi-modal authentication is defined as combining two or more human-machine authentication methods, whether initial or continuous, to increase the robustness of a system. Adding additional forms of authentication to increase the difficulty of compromising a system is referred to as multi-factor authentication. This is based on the three types of authentication: something you know, something you have, and something you are. In this document, multi-factor authentication will be considered a subset of multi-modal authentication.

Multi-factor authentication often references a smartcard token with the user entering a password or PIN to unlock the smartcard. Indeed, there has been much discussion as to whether it would be stronger if the password or PIN were not used to unlock the card but rather as a separate authentication. However, this is not the only type of multi-factor authentication, and there is ongoing research into a wide range of methods that may be used either as one-time per session or as a continuous monitoring authentication system [5].

While it is easy to understand that each additional factor should increase the strength of the authentication, it appears to be an oversimplification. The greater security strength of one factor may appear to make the other unnecessary or overly expensive. Factors that should be considered include offsets of known vulnerabilities or exposures, as well as impacts on usability. As an example, it has been noted that when using a two-factor mechanism, such as a time-varying apparatus and a pin, users often select a weak pin. By relying heavily on the time-varying component and not being zealous with the ownership of the device, the overall strength may not be justifiably increased.

Multi-modal authentication can add flexibility to many of the authentication systems in use today. With the additional capabilities of modern mobile devices and workstations, as well as the use of distributed networks, more options can be weighed. When supporting multiple types of devices, authentication may be considered not just for its added strength but also for usability. The implementation may impact the susceptibility for compromise as well as the usability for the user. Through the selection of appropriate multi-modal authentication, it may be possible to address several different environmental vulnerabilities while maintaining a robust posture. Additional considerations should include how they are integrated, architected, and managed.

3.1.2.4.1 Attributes

The addition of certain attributes can also aid in strengthening the authentication process. Prescribing the user environment in any meaningful manner may provide greater confidence. Attributes may be used for authentication, authentication and authorization, or just authorization, depending on the mechanisms of each and how compartmentalized the access may need to be. More information about attributes used in authorization is available [6].

Time

Authentication gated on certain days of the week or hours of the day has been supported in many systems but is seldom utilized. Similarly, organizations may choose to disable authentication for certain users during vacation or extended illness. Time limits are often employed and coupled with activity monitors to minimize exposure of accessibility if it appears that the user has abandoned the access. Time limits may be implemented in authentication, authorization, or both.

Location

Additional verification may be gained by attributes related to geographical location. Physical locations may include GPS, proximity sensors, and internal (controlled) IP addresses. Logical locations may include identified or expected IP address, expected time to respond, or trusted VPN. The number of simultaneous logins may also be a gating factor, though it is now used less often due to the number of devices that users access on a daily basis.

3.1.3 Domain: Machine-Machine

Another domain under the confirmation class is machine-machine authentication. This is often used for organizational or network system authentication, such as workstation and mobile device network connections, VPNs, or business to business communications. Early implementations often depended on shared secret keys, but it was difficult to protect the keys. Machine-machine based authentication is often based on a cryptographic scheme, such as PKI or other key agreement or key negotiation scheme. Single-sign-on schemes that support multiple authentications for a user after the initial user login should also be considered in this domain.

Machine-machine authentication is used to:

- Authenticate across a communications link
- Support a trusted devices network
- Support an automated (cached) human-machine authentication
- Provide other authentication data, such as location (example enterprise access to services)
- Provide trusted services (e.g., DNS, NTS, location, etc.)

Additionally, machine-machine authentication:

- Is usually cryptographic in nature
    - Often uses NIST-recommended protocols (e.g., IPSEC, TLS)
    - Uses either a pre-shared (symmetric) key or a digital signature
- Is set up by an administrator

- Is often transparent to the user
- Can be a cached human-machine authentication
- Can link temporally (recurring or not) or can be self-checking (see attestation)

3.1.4 Domain: Human-Human

The final domain in the confirmation class is human-human authentication. This is often used when a user is not able to gain access through the human-machine system. It is considered the easiest target and most susceptible to attack, primarily by social engineering. If the information used as authenticators is not sufficiently protected, the authenticator "database" becomes another source of attack.

There are two primary uses for human-human authentication. In the first case, an identity is established through credentials from other approved sources. This is typically done through identity management and is not associated with authentication as it is used here. An important aspect of this identity management human-human authentication is that the credentials, though provided by the user, have been authenticated from recognized sources outside of the authentication scheme.

The most common use for human-human authentication is as a backup system when the primary authentication mechanisms are either failed or locked out. When used as a backup system, the authentication relies on cached data—information that is typically given by the user for the purposes of reestablishing the identity of the user. When considering the strength of an authentication system, the backup system should also be considered. The human-human authentication can be quite costly due to the staffing involved. The use of user email addresses as a point of communication for reset information may mitigate some attack and cost issues. For these reasons, other methodologies such as text messaging through outside networks have become popular automated tiered mitigation techniques to human-human authentication.

3.2 Class: Attestation

Another class of authentication is attestation, which authenticates an object rather than an entity. A common example may be to hash a file to verify later that it has not changed. There appears to be a much wider spread of assurance requirements for attestation for many reasons, such as that the objects may be additionally protected by IAA mechanisms. Many of the same components and mechanisms are similar but not used for the same purpose. Currently, only one domain—attribute—has been identified, but this is expected to grow.

3.2.1 Domain: Attribute

This domain confirms an object by verifying an attribute of the object. To acquire some property of the object, reliance on an application or OS is typical due to operational constraints. While an attestation can be as simple as a CRC check, the assurance often relies on a cryptographic operation, such as a predetermined seed or key, to make it more difficult to substitute a new object and determine a new value. Many of the types of mechanisms used for machine-machine confirmation authentication may also be used in attribute attestation authentication.

Attributes should be selected such that the greater the confidence needed, the more difficult it is to change the object without being able to detect the change in the attribute. This does not necessarily mean that other attributes cannot be permitted to change. As an example, a keyed hash [7] or a digital signature [8] of a file can ascertain if the file remains unchanged, but it does not prevent a user from changing the association of the file by changing the extension of the filename. Simpler indications of a suspected file change may be sufficient, such as a change in date, a change in file size, or a dynamic measurement (e.g., monitoring a log file to make sure it only increases in size). Monitoring multiple attributes tends to increase the confidence attained when there are complex assurance requirements. While cryptographically defined attributes provide a significant amount of strength compared to other methods, they may not be able to characterize the object as needed.

The object most often used as the basic block for attestation is a file. In this document, a file may be a data file, an executable, or a collection of disassociated files grouped together by directory, compression process, memory location, or other compilation process. The file may be evaluated in dynamic memory or in storage. Hardware often has a collection of one or more software or firmware files that are verified at startup as a part of initialization. The identifying authentication, such as a digital signature, is stored as a separate segregated part of the file or externally in a protected area. Three families of attribute attestation are encryption, storage, and watermarking. The family depends on the focus of the attribute rather than the mechanism used.

3.2.1.1 Family: Encryption

3.2.1.1.1 Category: hashing

Hashing is often used to identify data that has not been changed since the hash was taken. Hashing is typically chosen when the use of the file is permitted but changes to the file are not. Once a hash is generated from the file, the resulting information cannot be reversed, and the "fingerprint" size is reduced to a length dependent on the hash algorithm. Protection of the hash is important to prevent the file from being changed or a new hash generated to replace the old. Protection of the hash can include secured storage or hashing the data combined with a secret key.

Digital signatures

Digital signatures provide verification that a file has not been changed. Typically, this type of attestation hashes the file of interest before encrypting the hash with a digital signature that can be traced back to the user and the certificate authority. Two major forms of digital signatures are DSA and PKI. However, Merkle signatures schemes are often used for blockchain protection against change.

Symmetric encryption

If it is not necessary to have unrestricted access to the file of interest, encrypting a file can also be used to ensure that it has not been unknowingly changed. Any changes to the encrypted file will result in the encryption being broken and non-recoverable unless the change is identified and reversed. This is especially useful for data transfer, which may include encryption prior to transfer or a transport scheme such as TLS or SSH.

3.2.1.2 Family: Storage

This is one of the few attestation attribute methods that does not necessarily rely on cryptography for protection but rather on separation from the object. Attributes may be stored separately from the object, usually under an IAA protection scheme or in a format that cannot be easily changed, such as using a keyed hash or similar mechanism. Some assurance products depend on attribute storage as a means of managing user or network systems.

3.2.1.3 Family: Watermarking

Watermarking differs from the other attestations in that it is typically focused on the representation embodied by the data rather than on the data itself. For example, a digitized color photograph is often not recognized by looking at the data. However, when the correct structure for the data is provided, the image can be displayed. In the same way, watermarking typically creates an embedded object on the representation of the data, such as an image. There are many uses for watermarking, including identifying protected work in an obvious or hidden manner, maintaining marking when copied or adjusted, or becoming obvious when the image is copied. While watermarking is not necessarily cryptographic, cryptography is often used to prevent manipulation of the watermark.

4 Properties

Several properties were observed in the creation of the taxonomy. Confirmation and attestation use many of the same authentication mechanisms. However, they are used very differently between the identity management, authentication, and authorization (IAA) process and the object management, authentication, and (sometimes) authorization (OAA) process. The authentication mechanisms between humans and machines have exposed the need to better understand trust relationships.

4.1 Overview of the IAA process for Confirmation

Figure 5 - IAA process

Authentication is a component of the IAA process, as shown in Figure 5. The IAA process consists of three unique tasks: identify, authenticate, and authorize. Historically, an IAA process was typically implemented as a single monolithic solution. Given the lack of any standards, the developer used best practices to provide a solution that combined the authentication and authorization components, leaving much of the identity management to the organization as a manual process. Some IAA process designs, such as Kerberos[9], were verified using formal methods to give a high assurance of proper design. Many solutions, however, were developed or modified for a specific environment with little or no formal process evaluation.

Each component of the IAA process should be defined with a common set of requirements applicable to all products. These requirements include assurance in the deployment and management of the systems. In this way, vendors can provide products that deliver focused solutions that are amiable to the other components. System integrators and those responsible for operational assurance can then better understand the requirements of the systems and deliver manageable, secure solutions by procuring products appropriate to their needs.

4.1.1 Identity Management (IM)

Entity authorization systems and object authentication systems are typically separate. However, both support similar requirements. The purpose of identity management is the issuance or adoption of a digital identity that is logically tied to a physical entity. The physical entity is based on the receipt of identification credentials from trusted parties, such as a passport, license, or organizational registration. The digital identity is an artifact produced to establish a presence on the systems of interest. It is this digital entity that the authentication gates and that the authorization component permits or restricts once authenticated.

The assurance of trust for the physical entity is usually related to the amount and quality of the third-party documentation, whereas the assurance of trust for the authentication of the digital entity is relative to the strength of the authentication used and the protection level of the resources to be accessed. Assurance of trust for both should be considered when designing and maintaining a system. In addition to the identity concerns, IM must communicate with both the authentication and authorization components to enforce the digital entity entitlements.

IM can be performed by a small, weak organizational component or be a formal entity. Examples may include a website administrator, a human resource department or manager, or a joint, multi-faceted umbrella organization. The IM sets the requirements for sufficient proof of identity for a user. Once the IM is satisfied that it has sufficient information, it will create a digital entity and enroll the virtual entity as some level of operator, directing the system's accesses on where and in what manner to provide access or support. The IM may direct facility and system administrators to enroll users in authentication systems or enroll the user directly. If done directly, the IM may issue the user a token, such as a PIV card, that permits access to any system that recognizes the IM as an authority. The IM may also be part of a federated or hierarchical network that manages user permissions beyond directly controlled assets.

Efforts such as the National Strategy for Trusted Identities in Cyberspace[1] (NSTIC) and REAL ID[2] provide insight into the capabilities and challenges of identity management. FutureID[10] is another large identity management effort by which credentials are used by credential transformers to create additional credentials. Though the lexicon differs, the management of identity is basically the same.

Of paramount importance to authentication is the communication and agreement between identity management and the authentication. At a minimum, communication between IM and authentication should support request permission, revocation, and acknowledgement of requests. In addition, if the hardware, software, or process that represents the entity is provided by the IM authority, parameters must be coordinated between IM and authentication to enable or update usage. In some cases, multiple authentication mechanisms must be managed simultaneously for independent multi-factor authentication mechanisms. This management must be interfaced into the IM controls.

Identity management may also communicate directly to authorization providers to manage access control parameters. As technology becomes increasingly complex, it is envisioned that the level of trust may be dependent on the type and number of authentication mechanisms, which may lead to dynamic trust levels. These trust levels and the resultant authorization must be communicated to the authorization provider, often following the governance of the IM.

[1] See http://www.nist.gov/nstic/.
[2] See https://www.dhs.gov/real-id-public-faqs.

4.1.2 Authorization

The last step of the IAA process is the enforcement of permissions: authorization. Upon receipt of a successful report from the IAA authentication component, authorization permits the digital entity access to execute programs or manipulate information. Often, the permissions offer some granularity, such as read-only, permission to execute, or allow the entity to edit the information.

The controls and constraints of authorization are addressed through role-based access control (RBAC) and attribute-based access control (ABAC) implementations. Mandatory access control (MAC) and discretionary access control (DAC) were early implementations of access control that either denied all unless allowed (i.e., MAC) or permitted all unless denied (i.e., DAC) [6]. It is not uncommon for data centers to manage access control implementations that are dependent on the operating systems controlling them. It should be noted that the above-mentioned controls are under the IAA component of authorization.

Communications between components focus primarily on allowing or denying a digital identity access. In conjunction with authorization, identity management permits or denies access to digital entity. Future developments may facilitate multiple authentication trust levels and are likely to place a heavier burden on the facilitation and management of authorization.

4.1.3 Authentication

The purpose of authentication is to confirm a digital identity through the manipulation of a hardware, software, or process that represents the entity. The identity represented is defined by identity management and communicated along with necessary information—often, just a permission—to the organization responsible for the authentication component. Upon successful manipulation of the hardware, software, or process representing the entity, the authentication component communicates to the authorization component a confirmation or denial to permit access.

Authentication of a digital identity is enabled by identity management. IM does this by either providing to the authentication component or requesting that the authentication component provide the hardware, software, or process. Costs of the provisioning of the authentication component may be a deciding factor. However, final permissions to or disallowing of (such as revocation) authentication for each digital identity are provided by the IM.

Authentication may disallow further attempts of authentication when a failed attempt threshold is exceeded. When the entity fails the authentication, the authentication owner decides whether the entity must authenticate through a different, typically separate process. As an alternative, the authentication mechanism may wait before allowing the entity to re-authenticate. The mechanism may increase the waiting period with each failed attempt before finally locking. Operational and time sensitivity may dictate the choice of re-authentication.

Communication with authorization is also required. While access oversight is typically administered by IM or the authorization management, an indication of success or failure is typically provided to the authorization mechanism by authentication. If multi-factor authentication is used, the outcome of each mechanism may be reported separately or as a single outcome depending on the sophistication of the authentication, IM, and authorization management. In some cases, attributes such as location may also be passed to the authorization component.

An important aspect of authentication is providing assurance that the mechanism prevents others from gaining access. Assurance is a variable, not an absolute, and the strength of authentication is its primary driver. Current authentication strengths are dependent on the type of mechanism used: biometrics depend on low false positives; passwords depend on unsuccessful guesses; and PKI implementations depend on strong public and private keys. However, these do not easily allow for comparison of the strengths of the mechanisms. Different authentication mechanisms have different balances of environmental factors, making the choice of authentication mechanism not solely a matter of the strongest or the most usable for every installation. There is no agreed upon methodology to compare the relative assurances of today's authentication mechanisms.

The hardware, software, biometric source, or knowledge under the control of the user is often referred to as the token or authenticator. It can take many different forms depending on the authentication process and the mechanisms used. In human-machine authentication, there are three basic forms that are often discussed: something you know, something you have, and something you are[11]. While these are not directly associated with authentication strength, the combination of these differing forms of authentication have historically been used to increase trust in the authentication process.

This section has discussed the IAA process for confirmation. Attestation is part of a similar process; however, it is not the same. Table 1 provides a high-level comparison of the two processes. Further information about the process when using attestation is provided in the next section.

Table 1 - IAA Confirmation vs. OA Attestation

	Identity Management	**Authentication**	**Authorization**
Confirmation	Validate entity docs Manage entities	Affirm virtual identity	Manage virtual identity rights to objects
Attestation	Manage Objects Manage IM and Object Credentials	Verify Object Goodness	*Authentication might gate object execution*

4.2 OA process for Attestation

The OA process provides assurance that an object is as expected by using attributes of that object. The process consists of two components: object management (OM) and authentication. Each component has a common set of requirements, which include assurance in the deployment and management of the systems. The OA process examples include data replication for multi-instance systems, such as banking or data transfer for warehousing, and typically exists inside of a system implementing an IAA process.

The amount of trust for the object is dependent on the selection of one or more object attributes and the environment, whereas the assurance of trust is relative to the strength of the authentication used to verify the object elements. Requirements for assurance of trust for each should be considered when designing or maintaining the OA system. OM and authentication may be combined or separated depending on the OA design. However, they must communicate with each other, even if separated, to manage entitlements.

4.2.1 Object Management

Object management provides oversight of the program or scheme to manage the trust of object embodiments. OM may either issue or delegate the issuance of an **artifact** to the authentication mechanism. If delegated, the authentication implementation is responsible for the creation of the artifact used to confirm object attributes. OM may also be responsible for identifying a specific version of an object or the retirement of that object in systems such as those that support version control.

OM can be performed as a stand-alone procedure, as part of an application, by a small organizational component, or as part of a federated system. Examples include applications supporting protected worksheets, applications monitoring operating system files, agencies supporting a standards library, or a database supporting worldwide banking. The OM sets the requirements for sufficient proof for the object. The OM may direct apps, users, or facility and system administrators to enroll objects, or it may enroll the object directly. The OM may direct authentication artifacts to be stored in places that restrict access, or it may direct that the enrollment material be embedded within an object container. The OM may also be part of a federated or hierarchical network that manages objects beyond directly controlled assets.

The communication between OM and authentication should support, as a minimum, request permission, revocation, and acknowledgement of the request. In addition, if the hardware, software, or process representing the object is provided by the OM authority, parameters must be coordinated between OM and authentication to enable or update usage. In some cases, multiple authentication mechanisms must be managed simultaneously for independent, multi-factor authentication object attributes. This management must be interfaced into the OM controls.

Object management may also communicate directly to IAA providers to manage access control parameters. As complexity increases, the level of trust may be dependent on multiple authentication object attributes. This may lead to dynamic trust levels. These trust levels and the resultant authorization must be communicated to the authorization provider, often following the governance of the OM.

4.2.2 Authentication

Authentication of an object is based on verification of one or more aspects of an object. The verification artifact produced from the authentication mechanism on one or more aspects of an object establishes a credential for the object of interest. It is this digital artifact that is used for the basis of the authentication processes, and it is typically protected. When authentication of the object is required, the authentication uses the digital artifact to validate the object to the assurance level determined by the choice of attribute selection and the authentication method used.

4.2.3 Authorization

Authorization is not considered part of the OA process but may be necessary for the management of an object. The authorization is done under the IAA process since an entity is given authorization permissions, whereas no case has been made to date that an object may need different authorizations. Upon receiving a successful report from the IAA authentication component, authorization permits an entity access to execute programs or manipulate information. Often, the permissions offer some granularity, such as read-only or permission to execute, or they allow the entity to edit the information once sufficient confirmation and attestation authentication have been achieved.

4.3 Trust relationships in Confirmation Authentication

Confirmation is based on at least one trust relationship. To identify and compare ways to authenticate, it is necessary to understand the trust relationships and define the common properties needed to support those relationships. The interweaving of authentications, such as those in federated systems or cloud computing, can obfuscate trust relationships. A single human-machine authentication may depend on several established machine-machine authentications, each of which is also a trust relationship. This section breaks down normal authentication processes into trust relationships and considers why they are established.

A successful authentication represents a trust relationship with sufficient confidence between parties. As an example, a simple handshake between people in an office environment may begin an introduction between the two parties, with one or both known as being associated with the organization. This provides a degree of confidence, and the organization is the identity manager. Similarly, an introduction in a public gathering may establish a relationship between an audience and a speaker or a choir. In daily life, these meetings appear as social norms. The amount of trust depends on the organization, the purpose of the exchange, the people involved, and the recognition of the participants.

4.3.1 Assignment Considerations

Digital authentication emulates real-life situations, whether it is human-machine or machine-machine authentication. However, social norms in the digital world are still being established, such as the digital handshake—a process that completes a negotiation and reaffirms trust. A digital handshake can be used to represent an individual but can also represent a more generic group of individuals, such as a role. A salesperson or service professional might be a real-world example of a role. Typically, role-based authentication is not considered as strong as an individual credential. In the role-based entity, it is one of several who share a credential, whereas the individual credential represents one specific entity. The strength of the mechanism used for authentication should not be confused with the strength of the role-based or individual-based authentication credential.

4.3.2 Links of Trust

Whether a credential is used by one person or many corporations, there is also a question as to how many authentications are being processed when establishing a communications link. For example, a brick and mortar store is usually easily identified, but shoppers are often anonymous until they decide to purchase. In a case where each entity of a two-way communication needs assurance of the other—perhaps the store has special pricing for store card holders—mutual authentication is sufficient. When multiple authentications must occur, such as in a credit card purchase, a multi-tiered authentication trust model is often necessary. This section addresses methods for establishing or re-establishing digital trust relationships.

4.3.2.1 One-Way Trust Authentication

One-way authentication is used when only one party needs to establish credentials, such as when a user or administrator logs onto a stand-alone workstation. When a user has an account on a workstation, the user must present a set of credentials that match one of the accounts that has been set up on the system. The user has no *digital* trust that the machine is the correct machine. However, the machine has confirmed a credential of the user.

In some circumstances, the system may be set up for multiple operators to access devices with the same credentials. The is referred to as role-based authentication. Typically, the authentication is the same as it would be for identity-based authentication. However, the IM has permitted several operators to share the same credentials (e.g., the administrators of a set of network routers). Though role-based authentication is losing popularity, it still exists and should not be confused with role-based access control (RBAC), which refers to controlling the access permissions of an authenticated operator rather than who can use the authentication process.

In web-based systems, it is common for the trust model for the workstation discussed earlier to be reversed. This is especially important because when using the internet, the user has no assurance that they have reached the correct machine. In this case, the user does not log in, but the server can be validated using a PKI TLS-based solution or similar. In Figure 6, a one-way authentication is represented by visiting a secure website that uses a certificate (the successful authentication is typically indicated by an icon on the browser) to verify the server and then negotiate security functionality. It is important to note that the server has little knowledge of the user since the user is not required to log in to maintain the connection.

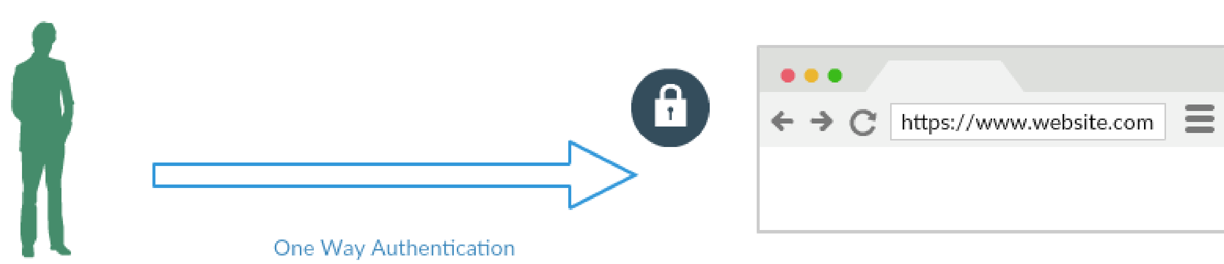


Figure 6 - One-way Authentication

4.3.2.2 Mutual Trust Authentication

Mutual authentication is typically used to validate both entities in a conversation. For example, if a shopper wishes to buy something from a store, they authenticate to the store through an account and/or payment, creating levels of trust in each direction. In this example, there are usually two different authentication methods. However, a single mechanism supporting mutual authentication is common.

Often, enterprises want stronger authentication when employees access services from outside of the corporate network. In that case, they might use a mutual TLS session, which is often considered to have a higher assurance due to the user obtaining a certificate that has been issued by the same or recognized certification authority.

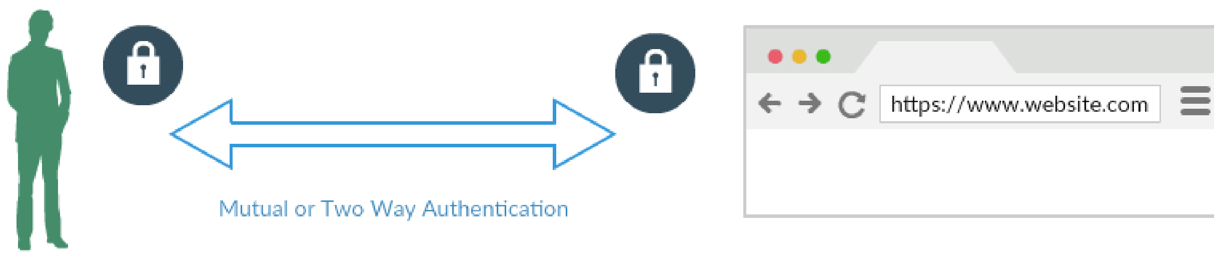


Figure 7 - Mutual Authentication

Mutual authentication is demonstrated in Figure 7. Both the user and the server have valid certificates so that they may authenticate each other through something like the TLS protocol. While there are other ways to perform mutual authentication, this is a good example of how the same authentication mechanism can be used for both parties.

Federated vs Hierarchical

The above discussion of mutual authentication has an important aspect to it: a hierarchical structure can be traced back to a primary server. PKI services are often managed in this manner, with the primary server identified as the certificate authority. However, a single authority is not the only structure that can be used. Federated systems may have several central servers or elements. While this can become quite cumbersome, it may make providing services easier. Browsers often support multiple hierarchical PKI services, which in turn support a simple form of federated authority systems.

4.3.3 Multi-Level Trust Authentication

Multi-level authentications are achieved by a combination of one-way and mutual trust relationships. Using a previous example, it is typical for a server to provide SSL protection using the server certificate when purchasing. The browser supports the user protection by checking for a valid credential from the online storefront. However, the store vendor does not know who is browsing unless they log on with some credentials, such as a username and password. An online purchase with a credit card presents a very complex set of relationships.

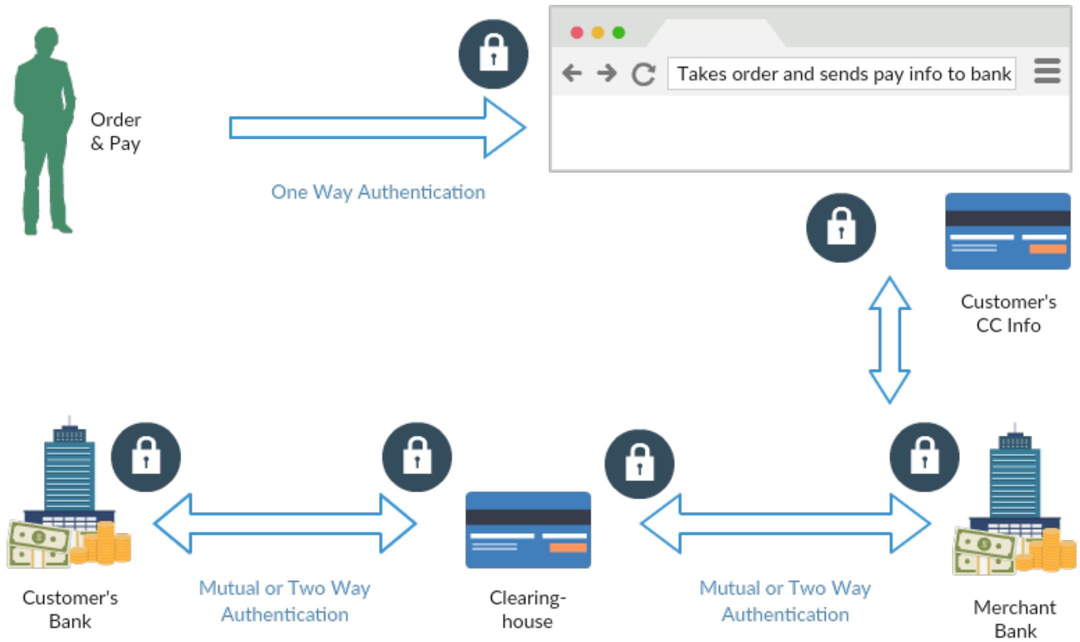


Figure 8 - Multi-path authentication

Figure 8 depicts three trust relationships with three different authentication types. Authentication using PKI certificates is indicated in every entity apart from the user. To make a purchase on a website, the user may log in with a username and password or a similar mechanism for the storage of user information, enhancing the convenience of the user and providing additional assurance to the shopkeeper. Either as part of that information or separately, the shopper's credit card information is used as an authentication mechanism to transfer money from the user's account to the merchant's account. This process uses multiple connections and relationships—including the credit card clearinghouse, the merchant's bank, and the shopper's bank—to manage and verify accounts and fees.

Trust relationships form the basis for authentication paths. The assurance necessary to support the needed IAA process can only be assessed by following each relationship with the authentication path. The trust relationship begins with identity management and ends with authorization. However, it is the mapping of those relationships during authentication that provide much needed assurance.

4.4 Trust Relationships in Attestation Authentication

Attestation is typically based on only one trust relationship; the object is the same as what was expected. The selection of the attribute used for "what was expected" is important as it provides the uniqueness of the attribute and may constrain the methods of protection that are reasonable for the comparison artifact. For example, a filename and date may be perfectly adequate to specify a file, but they give little assurance as it would not be hard to change the contents of the file. However, hashing a file might be a clever way to affirm that the file representing an object did not change, and digital signatures are sometimes used to verify a part of an object. In other cases, some files (e.g., log files) are expected to change but typically should only increase in size unless audit material was removed (i.e., tampered with). Focusing on what a data set might represent instead of what it is may move the evaluation of trust in a different direction. Some objects may appear as random information unless processed, such as in a digital picture. The importance may not be in the "file" aspect but rather in the display aspect, so watermarking may be more appropriate for identifying the display of the original versus copies obtained from entities other than the original source object.

The trust of these objects depends in large part on the management that should understand the purpose of the object, the manner of establishing trust, and the amount of trust needed. Authentication provides the amount of trust and depends on several things: the aspect of an object, the uniqueness of the artifact generated, the strength of protection provided by the artifact, and possibly the network protection—though it is outside of the OA's control—provided by the authorization of an IAA system. OM would select the aspect of the object, which would impact the uniqueness of the artifact. The strength of the authentication determines the strength of the artifact protection. The host IAA system, if available, limits access to the object and can increase the trust.

4.5 Basic Mechanism Components

While the primary function of authentication is to investigate the entity's credential, the schemes necessary to do this vary depending on the delivery mechanisms used to communicate between the user authentication evidence and the system doing the evaluation. Key aspects of authentication may have environmental considerations dependent on the region. For example, a remotely implemented sensor that needs to communicate across several networks will also need a more sophisticated implementation than that of one directly connected to a non-networked device containing internal storage. Special considerations may be noted for application, local platform, internal network, web, and cloud environments. While physical security has been relied upon for local implementations, protection across networks is commonly provided using encryption technologies. In general, as authentication mechanisms are used across greater distances and multiple platforms, more diverse implementations and interactions are needed for stronger, versatile protection. Five basic components have been identified in the mechanisms: identity representation, sensors, communications, storage, and processing.

4.5.1 Identity Representation

Identity representation is the information or hardware that the entity or object presents for authentication. Examples include PIV cards, passwords, time-synched number generators, a face, a finger, or an equivalent object, such as a hash or signature. These are typically provided to the sensor.

4.5.2 Sensors

The authentication sensor provides the connection between the user and the system, representing the system. Examples of sensors for authentication could be a keyboard for passwords, a smartcard reader for PIV, a camera for facial recognition continuous authentication, or an IP address for location. Location services, such as GPS, may also be used as sensors to supplement authentication information. Considerations in the choice of apparatus and location may include mitigations of issues caused by false acceptance, bypassing or replacing, skimming, wear, passive sensing, or abuse.

4.5.3 Communications

Communications provide the link between the location of the entity or object and the location of the authentication system, linking between the authentication service and those of IM, OM, and authorization. The links are often protected by an encrypted tunnel, though alternate methods may be acceptable. Encryption methods that are typically used for normal, secured communications are also utilized for authentication and often have separate authentication services for control of the authentication system being protected.

4.5.4 Storage

Secure storage is one of the most critical elements of authentication mechanisms. Hackers have access to collections of compromised passwords and user information acquired through the exploitation of security flaws or misconfigurations in actual systems. Most of these vulnerable systems used little or no encryption protection, allowing hackers to access authentication server databases. This has demonstrated that layers of protection are important. Even protecting each password by a fixed keyed hash can be insufficient because, once acquired, the attacker has time and access to sufficient computational power.

The storage of private data is crucial to every form of authentication, and some biometric data could result in permanent losses if compromised. Schemes like fuzzy vault may provide protection for biometrics. However, these often lack the scrutiny of other forms of protection, and malware may compromise even well-crafted protection mechanisms. Secure storage is best addressed by supporting multiple layers of protection with proper assurance controls.

4.5.5 Processing

Historically, authentication has been primarily on server level equipment. Certainly, there are authentication mechanisms that require moderate to fast processing power when used at the number of authentications needed; for example, cloud computing is seen as escalating complexity and volume requirements. However, in trying to protect smaller communication channels, such as for IoT devices, other limitations posed on the processing require consideration, such as low power and memory constraints. Additionally, newer authentication methods, such as continuous authentication, may require some additional processing for multi-modal analysis and decision making, even at the mobile level of processing.

5 Building and Maintaining Authentication

One of the biggest factors in deciding which type of authentication mechanism to deploy in a new system is the appropriateness or suitability of the mechanism. Historically, the system was tied to a mainframe or networked workstations, and system designers could optimize authentication controls in a rather well-defined environment. While it is still considered easier to implement authentication in a well-defined and secured environment, most of today's environments are constantly changing and often openly accessible. Mobile device integration and other concerns are making the environmental extremely diverse. The implementer and management can address most common issues by considering four major categories: security, deployability, usability, and manageability.

Security focuses on common environmental aspects that an attacker can use to compromise a user's credentials. It addresses being in proximity to a user, such as overhearing a user vocally give out a credit card number when contacting the bank. It also addresses an attacker using techniques to remotely gain access, such as a guessing a password or tricking the user through false emails to compromise sensitive information.

Deployability is focused on aspects of the process that are important to designers and implementers. Deployment issues are often related to cost drivers of standing up or renewing an existing capability. It addresses the selection of the user authentication and the resulting cost of purchase, possible enrollment costs (separate from identity management enrollment), delivery, policy creation, support establishment, and the creation and implementation of initial training for users and support.

Usability focuses on two principal areas: the end-user experience and the support or administrator experience. Usability is an important but often not addressed factor for successful security implementations. Usability is an attempt to quantify the amount of effort that a valid user must endure to achieve a goal, such as authentication. It has been reported that when the barrier to security for valid users is too high, the users are often found to be highly effective in subjugating the security. A simple example of this might be the user posting the password on the monitor of the computer because the password was too difficult to remember. Since users' abilities often vary widely, sufficient usability is not easily defined.

Manageability is the final category and addresses the entire support effort necessary to maintain and assure proper operation of the authentication process. Though deployability is charged with the initial rollout of user enrollment, manageability includes ongoing provisioning, such as the addition, removal, and maintenance of user accounts, as well as the policies and procedures supporting them. As systems mature, policies and procedures must often change due to outside requirements, including legislation, equipment resources, technology improvements, and support for additional services.

Much of the framework for addressing these issues is based on [12], which discusses several different types of authentication mechanisms. A separate consideration for manageability has been added to address the resources necessary to maintain proper operation. Several considerations for each of these categories are synthesized below, many of which should be expanded upon and verified independently. To that end, the work should either support those chosen, possibly with adjustments, or should lead to the identification of additional or different attributes and supporting characteristics.

5.1 Security Attributes

Security weaknesses can be grouped into social engineering, malware, misconfiguration, and vulnerability.

Social engineering:

- Observation – Observation of user or user environment that is used to gain access
- Failover – Forcing a system to use other methods of gaining access
- Guessing – Unlimited attempts to retry authentication
- Strict following of guidelines – Policy guidance provides template, making attack easier
- Data acquisition – Use of readers collocated with valid readers to skim, scan, or record user data without interrupting the transaction
- Authenticator acquisition – acquisition of authentication hardware or software devices; biometric, location, or time-sensitive data; or other evidence of authenticity

Configuration vulnerabilities:

- Server evidence repository – Lack of sufficient protection to prevent being acquired and attacked offline
- Communication observance – MITM attacks, replay attacks, keylogger

Information leakage (including privacy considerations):

- Packaging – Labeling/branding of card
- Help Desk – Information associated with user
- Reporting – Logging of access, including location, time, etc.
- Feedback – Display of entry information, audible information, identity, etc.

5.2 Deployability Attributes

Deployability can be grouped into accessibility, cost, and compatibility.

Accessibility:

- Disability considerations – Authentication meets user accessibility requirements
- Restrictions – Environment supports necessary safety requirements

Cost:

- Acceptable cost per user – Cost for each user to be equipped, registered, and managed
- Acceptable cost for risk – Cost is supported by cost of loss or loss of access
- Acceptable implementation costs – Costs are within implementation or renewal budget, including recovery and re-enrollment

Compatibility:

- System – Works with system being protected, including platform, network, and apps or plug-ins
- Organization – Includes management and policy administration
- Authentication can be scaled – For number of users, number of servers, administration

5.3 Usability Attributes

Usability attributes are associated with effectiveness, efficiency, and satisfaction.

Effectiveness:

- Short authentication setup, delivery, service, and issue support
- User entry is not susceptible to errors, sufficient feedback to user
- Recovery requires minimal time and effort

Efficiency:

- Availability and ease of understanding authentication policies and procedures

Satisfaction:

- Light user requirements, no onerous memory requirements, no need to carry token, etc.
- Accounting for other user authentication requirements, including non-associated sites
- Integrated with user process workflow

5.4 Manageability Attributes

Considerations that address manageability concerns can be grouped into annual costs and long-term availability.

Annual Costs:

- Administrative support

- Tokens
- IT support for communication, server, and storage
- Reader support and maintenance

Long-Term Availability:

- Tokens
- Readers or other sensors
- Server hardware and software

6 Metrology for Authentication

Historically, the strength of an authentication has been directly attributed to the encryption used in the decision process. This does not apply to non-encryption-based human-machine mechanisms, such as passwords or biometrics. Using the strength of the encryption as a measure is an optimistic value. There are typically many design, implementation, maintenance, and operational issues that drastically reduce the actual strength of the system. Further, having it based only on the decision process encryption ignores any protection that was used for the transfer of authentication information, any protection of secret data during storage, and any implementation or configuration flaws that could result in compromise.

In authentication with a human-machine interface that is encryption-based, workarounds are made to deal with human limitations. Users are often limited when it comes to remembering key lengths of sufficient strength and the number of keys they would need to retain for the systems that they access. Alternatives have been developed that are not based on humans remembering encryption components directly but rather involve an additional step, such as "something you have."

For systems that support a human interface yet are not encryption-based, encryption may be employed to add complexity to the system to make it more difficult for the attacker. For example, alternative systems may be based on some form of password or biometrics. Much work has been done within the human-machine domain in trying to determine security metrics for each family of mechanisms, including the entropy of passwords, the false acceptance rates of biometrics, and the key strength of PKI solutions. However, these measurements are not easily compared across the different families. Yet again, there are several additional considerations. User interface and the surrounding environments also affect security strength. These are usability concerns and can easily compromise the authentication of an individual and the resulting access that is granted.

Determining the strength of an authentication that incorporates a human interface is a complicated process, even considering only one of the myriad implementations. Due to this complexity, current standards for human-machine confirmation appear to address multiple levels of security strength. However, there appear to be two solutions: anything or "two-factor" authentication. To improve the ability to set requirements for authentication, a set of measurements are needed to evaluate and compare authentication mechanisms and measurements for security and usability.

6.1 Security

One of the most important aspects in selecting authentication mechanisms should be minimizing compromise. While no specific methods of metrology for authentication have been identified to date, some candidates are discussed below. It is not expected that all mechanisms demonstrate high strength across all measurements. It is likely that multiple measurements will be necessary to adequately address the overall posture of the service.

6.1.1 Representation

This is a measurement of the linkage between the token and the entity being authenticated. It is expected that the more closely the token can be tied to the entity, the higher the assurance. However, the token must be selected in such a way that it can represent an aspect of the entity in a manner that would not be confused with another.

6.1.2 Inimitable

This is a measure of the resistance of the token to being duplicated or otherwise compromised. A compromise is often related to the type of authentication. It is the resistance to the compromise that is important, not necessarily the specific compromise applied. As there may be multiple applicable susceptibilities, the measure of the least resistance should be associated with the security strength of the mechanism implementation.

6.1.3 Secure Delivery

This consideration should measure the protection of the token from the point of input by the entity to the point of authentication assessment and the decision of the assessment to the authorization management. Protection should address a combination of vulnerabilities from non-deliberate user compromise, substitutions, and omissions. There may be multiple points of interface with the entities that may use multiple secure technologies, each of which should be addressed.

6.1.4 Secure Storage

This is a measure of the protection of the reference information that the authentication mechanism uses to verify the entity. The measure of protection should apply to both the active storage and any backup storage. As different methods may be used, different measurements can be expected. The protection level must be made commensurate with the maximum level of risk for the entire system.

6.2 Usability

Usability focuses on human-machine authentications and is a relatively new concern for authentication methods. Consideration for usability was pushed by Adams and Sasse [13], who claimed that security without considerations for usability could no longer be a supportable direction. It is difficult for most users to understand the cost of security, but they quickly discover how it impacts them operationally. When faced with difficult or overwhelming tasks to accommodate security requirements, users often utilize coping strategies that may weaken security. Developers and implementers attempt to address the limitations of human capabilities through the choices and policies of the authentication mechanism.

Operational processing requirements are seldom considered. Closer alignment of security barriers to workflow will make it easier for users to support and adopt the imposed operational requirements [14]. Measuring the usability of a process flow that contains authentication is more representative of what the user must deal with in their environment. The greater the pressure of time, obfuscation, or accuracy placed upon the user during authentication, the greater the chance of error. If it is possible to design the authentication to be aligned with the work and not the obstacle to overcome to do work, there is a greater degree of usability.

Usability is often assessed by the extent to which users can achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use. While usability is a critical component of security in authentication, it is often wrongly assumed that it has been addressed in previous similar implementations. To date, most work in the assessment of authentication usability has utilized a standard that addresses the usability of video displays, ISO 9241-11. Under IOS 9241-11, there are three areas of focus: *satisfaction*, which is a subjective measurement, and *effectiveness* and *efficiency*, which can be calculated. These are likely to have low correlation factors, according to [15]. If usability is measured in this manner, it should be measured in all three areas.

Being **effective** is about doing the **right** things, while being **efficient** is about doing things **right**.

6.2.1 Effectiveness

Effectiveness is a measure of the accuracy and completeness with which users achieve specified goals. This measurement is often achieved by compiling operator errors, such as mistyping, inserting cards backwards, or biometric errors due to user habits. Additional measures could include the availability of aides, such as procedures and expectations, use of password safes, or single sign-on implementations.

6.2.2 Efficiency

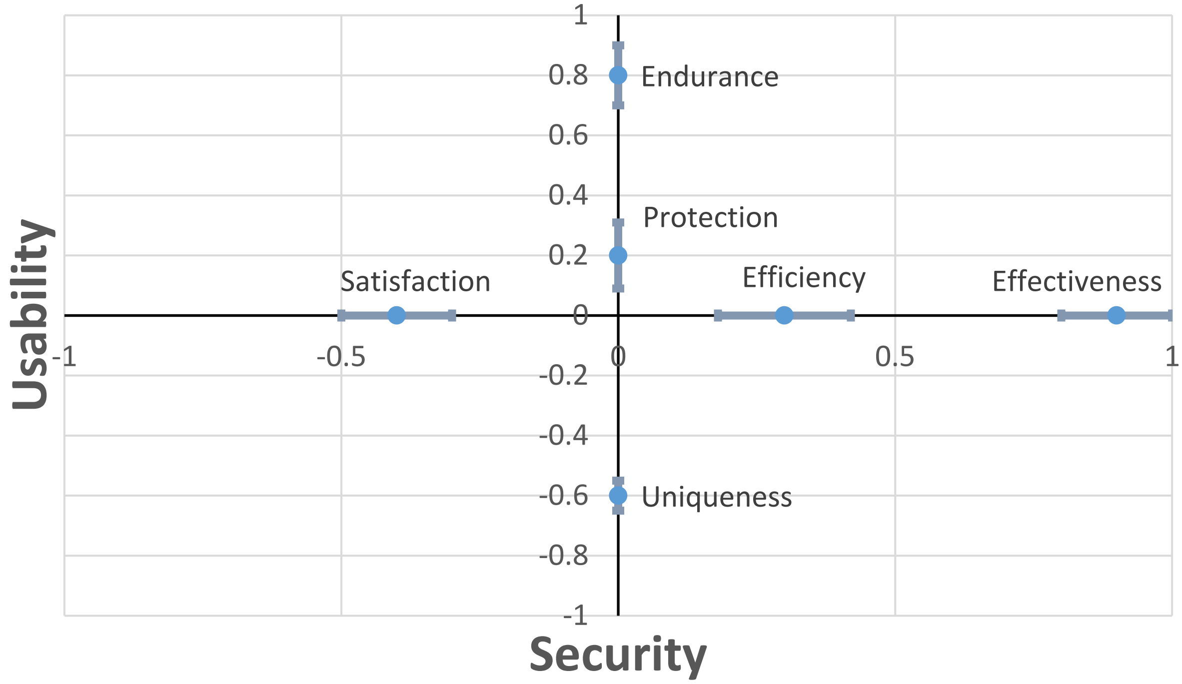
Efficiency is measured as the resources expended in relation to the accuracy and completeness with which users achieve goals. Password vaults, written passwords, and the reuse of passwords are examples that impact the efficiency of the authentication. Bitcoin’s level of effort to process the blockchain is an example where efficiency is compromised to elevate security.

6.2.3 Satisfaction

Satisfaction is a goal to achieve freedom from discomfort and positive attitudes towards the use of the product. The measurement of satisfaction is a qualitative measurement and, as such, is more subjective. It may be less relied upon than effectiveness or efficiency in decision making, but it is an important measure of the willingness of the user to support authentication.

6.3 Usability vs. Security

Several password authentication studies since Adams and Sasse have noted what appears to be an inverse correlation between usability and security. If this is an indicator for all types of human-machine authentication, measurements in security and usability may indeed demonstrate causal interactions. It seems reasonable that similar effects can be evaluated for all types of human-machine authentication. If there is an association between usability and security, the relationship may be demonstrated by visualizing these measurements. Figure 9 is an example of how this data may be used to evaluate the trade-offs and gain a better understanding of the relationship between security and usability.



1243

1244 **Figure 9 – Security vs. Usability (Conceptual)**

1245

References

[1] Bradner S (1997) Key words for use in RFCs to Indicate Requirement Levels. https://doi.org/10.17487/rfc2119

[2] Security Requirements for Cryptographic Modules (2002) https://doi.org/10.6028/NIST.FIPS.140-2

[3] Grassi PA, Fenton JL, Newton EM, Perlner RA, Regenscheid AR, Burr WE, Richer JP, Lefkovitz NB, Danker JM, Choong Y-Y, Greene KK, Theofanos MF (2020) Digital Identity Guidelines: Authentication and Lifecycle Management. https://doi.org/10.6028/NIST.SP.800-63B

[4] Wilson C, Grother P, Chandramouli R (2013) Biometric Data Specification for Personal Identity Verification. https://doi.org/10.6028/NIST.SP.800-76-2

[5] Schaffer KB (2015) Expanding Continuous Authentication with Mobile Devices. *Computer* 48(11):92–95. https://doi.org/10.1109/MC.2015.333

[6] Hu V, Ferraiolo D, Chandramouli R, Kuhn DR (2017) *Attribute-Based Access Control* (Artech House).

[7] The Keyed-Hash Message Authentication Code (HMAC) (2008) https://doi.org/10.6028/NIST.FIPS.198-1

[8] Digital Signature Standard (DSS) (2013) https://doi.org/10.6028/NIST.FIPS.186-4

[9] Bella G, Riccobene E (1997) Formal analysis of the Kerberos authentication system. *Journal of Universal Computer Science* 3(12):1337–1381.

[10] Bruegger BP, Lipp P (2016) Lightest-a lightweight infrastructure for global heterogeneous trust management.

[11] Grassi PA, Garcia ME, Fenton JL (2020) Digital Identity Guidelines. https://doi.org/10.6028/NIST.SP.800-63-3

[12] Bonneau J, Herley C, van Oorschot PC, Stajano F (2012) The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. *Proc. IEEE Symposium on Security & Privacy*

[13] Adams A, Sasse MA (1999) Users are not the enemy. *Communications of the ACM* 42(12):40–46.

[14] Sasse A (2015) Scaring and Bullying People into Security Won’t Work. *IEEE Security \& Privacy* (3):80–83.

[15] Frøkjær E, Hertzum M, Hornbæk K (2000) Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp 345–352.

[16] Dang Q (2013) Recommendation for Applications Using Approved Hash Algorithms. (NIST Special Publication (SP) 800-107, Rev. 1). https://doi.org/10.6028/NIST.SP.800-107r1

[17] JTF (2020) SP 800-53 Revision 5 DRAFT PRE-DRAFT Call for Comments: Security and Privacy Controls for Federal Information Systems and Organizations. (NIST Special Publication (SP) 800-53 Revision 5; includes updates as of 12-10-2020;). https://doi.org/10.6028/NIST.SP.800-53r5

[18] Stine K, Kissel R, Barker WC, Fahlsing J, Gulick J (2008) Volume I: Guide for Mapping Types of Information and Information Systems to Security Categories. (Gaithersburg, MD). https://doi.org/10.6028/NIST.SP.800-60v2r1

[19] Burr W, Dodson D, Nazario N, Polk WT (1998) Minimum Interoperability Specification for PKI Components (MISPC), Version 1. (NIST Special Publication (SP) 800-15). https://doi.org/10.6028/NIST.SP.800-15

[20] Security and Privacy Controls for Federal Information Systems and Organizations (2013) https://doi.org/10.6028/NIST.SP.800-53r4

[21] Ross R, Pillitteri V, Dempsey K, Riddle M, Guissanie G (2020) Protecting Controlled Unclassified Information in Nonfederal Systems. (Gaithersburg, MD). https://doi.org/10.6028/NIST.SP.800-171r2

[22] Shirey R (2007) Internet Security Glossary, Version 2. Available at https://tools.ietf.org/html/rfc4949

[23] Dukes C (2015) Committee on national security systems (CNSS) glossary. *CNSSI, Fort Meade, MD, USA, Tech. Rep*

Appendix A—Acronyms

Selected acronyms and abbreviations used in this paper are defined below.

IAA	Identity Management, Authentication, and Authorization process
IM	Identity Management
OA	Object Authentication
OAA	Object Management, Authentication, (sometimes) Authorization process
OM	Object Management
PKI	Public Key Infrastructure
SP	NIST Special Publication
TLS	Transport Layer Security

Appendix B—Glossary

The term definitions are included here to allow clarity throughout this document. Where possible, a suitable external definition has been repeated, and the source document is listed. It is hoped that these definitions will encourage communications when discussing the IAA process.

algorithm [16]	A clearly specified mathematical process for computation; a set of rules that, if followed, will give a prescribed result.
artifact	For attestation authentication, the artifact is created by the OM or authentication component as a reference for validating the object attribute of interest.
attributes	Attributes are metadata to the information of interest. In confirmation authentication and authorization, an attribute is additional information, such as location, which may be necessary for successful authentication or authorization. In attestation, an attribute is information about an attribute previously sampled by an authority that is used to validate the object.
authentication	One of the steps in the IAA process: identify, authenticate, and authorize. A component of the IAA process in which a token is tested.
authentication mechanism	A method of implementing authentication instantiation, typically based on a method of confidentiality. The authentication taxonomy is organized by the mechanisms used for a type of authentication.
authentication reference	The information kept by the service to validate the user's token.
authentication scheme	Used in this document to characterize a mechanism or combination of mechanisms to implement authentication in an IAA process.
authenticator [17]	Something that the claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the claimant's identity. This was previously referred to as a token.
authorization	A component of the IAA process in which an entity is permitted select physical or digital access after successful authentication.
cryptology [18]	The science that deals with hidden, disguised, or encrypted communications. It includes communications security and communications intelligence.
digital entity	A digital entity is a representation of an actual entity created by identity management. It is not the token that may be assigned to the digital entity for authentication.

hash [19 adapted]	A function which maps strings of bits to fixed-length strings of bits, satisfying the following two properties: it is computationally infeasible to find for a given output an input which maps to this output; and it is computationally infeasible to find for a given input a second input which maps to the same output.
IAA process	A method used to allow a given entity one or more entitlements for digital or physical access or to accomplish a goal. In this document, the IAA process is implemented by the set of components: identity management, authentication, and authorization.
identity management	A component of the IAA process in which an entity is vetted and, if sufficient, either issues or permits a token for use in authentication.
ontology	Defines the organization, structures, properties, and interrelations of a complex idea or construct.
privileged account [20]	An information system account with approved authorizations of a privileged user.
multi-factor authentication [17]	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.
multi-modal authentication	Multi-modal authentication is defined as combining two or more human-machine authentication methods, whether initial or continuous, to increase the robustness of a system.
privileged user [21]	A user that is authorized (and therefore, trusted) to perform security-relevant functions that ordinary users are not authorized to perform.
properties	The basic objects for building the ontology for authentication.
protocol [22]	A set of rules (i.e., formats and procedures) to implement and control some type of association (e.g., communication) between systems.
system	In this document, system represents a collection of concepts or implementations that can be considered stand-alone.
taxonomy	A scheme of classification for a subject. For authentication, the classification is broken down into a hierarchy of classes, domains, families, and categories.
token	Though token is used differently in many authentication standards, it is the hardware, software, or process that represents the entity in the authentication process. Because this term is used to represent many

	different things in different authentication mechanisms, a different term is being sought. It is sometimes referred to as an authenticator.
validation [23]	Confirmation (through the provision of strong, sound, objective evidence) that requirements for a specific intended use or application have been fulfilled (e.g., a trustworthy credential has been presented, or data or information has been formatted in accordance with a defined set of rules, or a specific process has demonstrated that an entity under consideration meets, in all respects, its defined attributes or requirements).
verification [23]	Confirmation, through the provision of objective evidence, that specified requirements have been fulfilled (e.g., an entity's requirements have been correctly defined, or an entity's attributes have been correctly presented; or a procedure or function performs as intended and leads to the expected outcome.