DRAFT

Analysis of Methods of Trust Elevation Version 1.0

Working Draft 06

7 April, 2013

Technical Committee:

[OASIS Electronic Identity Credential Trust Elevation Methods (Trust Elevation) TC](https://www.oasis-open.org/committees/trust-el/)

Chairs:

Abbie Barbir ([abbie.barbir@bankofamerica.com](mailto:abbie.barbir@bankofamerica.com)), [Bank of America](http://www.boa.com)

Don Thibeau ([don@openidentityexchange.org](mailto:don@openidentityexchange.org)), [Open Identity Exchange](http://www.openidentityexchange.org/)

Editors:

Peter Alterman ([palterman@safe-biopharma.org](mailto:palterman@safe-biopharma.org)) SAFE-BioPharma Association

Shaheen Abdul Jabbar ([shaheen.n.abduljabbar@jpmchase.com](mailto:shaheen.n.abduljabbar@jpmchase.com)), [JPMorgan Chase Bank, N.A.](http://www.jpmorganchase.com/)

Jaap Kuipers ([jaap.kuipers@id-network.nl](mailto:jaap.kuipers@id-network.nl)), Individual

Thomas Hardjono ([hardjono@mit.edu](mailto:hardjono@mit.edu)), [M.I.T.](http://www.mit.edu/)

Mary Ruddy ([mary@meristic.com](mailto:mary@meristic.com)), [Identity Commons](http://www.identitycommons.net/)

Related work:

This document is related to:

* [Survey of Methods of Trust Elevation Version 1.0](https://www.oasis-open.org/apps/org/workgroup/trust-el/download.php/46987/trust-el-survey-v1.0-wd01.doc)

Abstract:

This document is an analysis of the methods of trust elevation identified in this TC’s previous survey of methods of trust elevation. In it, methods of trust elevation are systematically evaluated for vulnerabilities. This document, which is the Trust-Elevation TC’s second deliverable is intended to be used as input to subsequent work identifying ways of combining these methods to further elevate trust to achieve desired levels of assurance.

Status:

This preliminary [Working Draft](http://www.oasis-open.org/committees/process.php#dWorkingDraft) (WD) has been produced by one or more TC Members; it has not yet been voted on by the TC or [approved](http://www.oasis-open.org/committees/process.php#committeeDraft) as a Committee Note Draft. The OASIS document [Approval Process](http://www.oasis-open.org/committees/process.php#standApprovProcess) begins officially with a TC vote to approve a WD as a Committee Note Draft. A TC may approve a Working Draft, revise it, and re-approve it any number of times as a Committee Note Draft.

Copyright © OASIS Open 2013. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full [Policy](http://www.oasis-open.org/policies-guidelines/ipr) may be found at the OASIS website.

This document and translations of it may be and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Table of Contents

[1 Introduction 5](#_Toc349077011)

[1.1 Philosophical Approach 5](#_Toc349077012)

[1.2 Analysis Approach 6](#_Toc349077013)

[1.2.1 Vulnerabilities are a moving target 6](#_Toc349077014)

[1.2.2 Analysis Template (Vulnerabilities list) 7](#_Toc349077015)

[2 Analysis 10](#_Toc349077016)

[2.1 Method Category – What You Are 10](#_Toc349077017)

[2.1.1 Physical Biometrics 10](#_Toc349077018)

[2.1.2 Behavioral Biometric 14](#_Toc349077019)

[2.2 Method Category – What You Know 18](#_Toc349077020)

[2.2.1 Password 18](#_Toc349077021)

[2.2.2 KBA 23](#_Toc349077022)

[2.3 Method Category – What You Have 28](#_Toc349077023)

[2.3.1 End Point Identity 28](#_Toc349077024)

[2.3.2 Hard Token 42](#_Toc349077025)

[2.3.3 Soft Token 46](#_Toc349077026)

[2.3.4 Out of Band 50](#_Toc349077027)

[2.3.5 One Time Password (OTP) 54](#_Toc349077028)

[2.4 Method Category – What You Typically Do 58](#_Toc349077029)

[2.4.1 Time of Access 58](#_Toc349077030)

[2.4.2 Browsing Patterns 62](#_Toc349077031)

[2.4.3 Mouse patterns 66](#_Toc349077032)

[2.5 Method Category – Context 70](#_Toc349077033)

[2.5.1 Context 70](#_Toc349077034)

[Appendix A. Acknowledgments 75](#_Toc349077035)

[Appendix B. References 78](#_Toc349077036)

[Appendix C. Revision History 79](#_Toc349077037)

Introduction

This document is the second deliverable of the OASIS Electronic Identity Credential Trust Elevation Methods Technical Committee. In our first deliverable, [Survey of Methods of Trust Elevation Version 1.0 [1],](https://www.oasis-open.org/apps/org/workgroup/trust-el/download.php/46987/trust-el-survey-v1.0-wd01.doc) we surveyed methods of trust elevation and identified five categories. In this deliverable, we analyze the various methods of trust elevation in these five categories by assessing their effectiveness at assuring the identity of the electronic claimant, in preparation for creating a general model of how effective the trust elevation / risk mitigation efforts are in generating trusted online transactions. The intention is to use the results of this analysis as an input to a subsequent phase in which methods of trust elevation are combined or factored in specific ways to further reduce vulnerabilities and elevate trust to specific levels of assurance or risk mitigation, representing increasing degrees of authentication certainty.

## Philosophical Approach

A goal of the TC is to identify groups of methods of trust-elevation that can be combined with an initial credential to achieve specific, desired levels of assurance of identity during an online transaction. This approach is derived from and is an extension of the well-established use of multi-factor authentication to increase trust.

”Authentication methods that depend on more than one factor are more difficult to compromise than single-factor methods.”[2]

As more organizations accept third party identities, there is a need not only to use multiple factors to thwart threats, but also to align the level of assurance of the credential initially presented on behalf of a user to the level of assurance required by the resource provider (RP). This is illustrated in Figure 1.

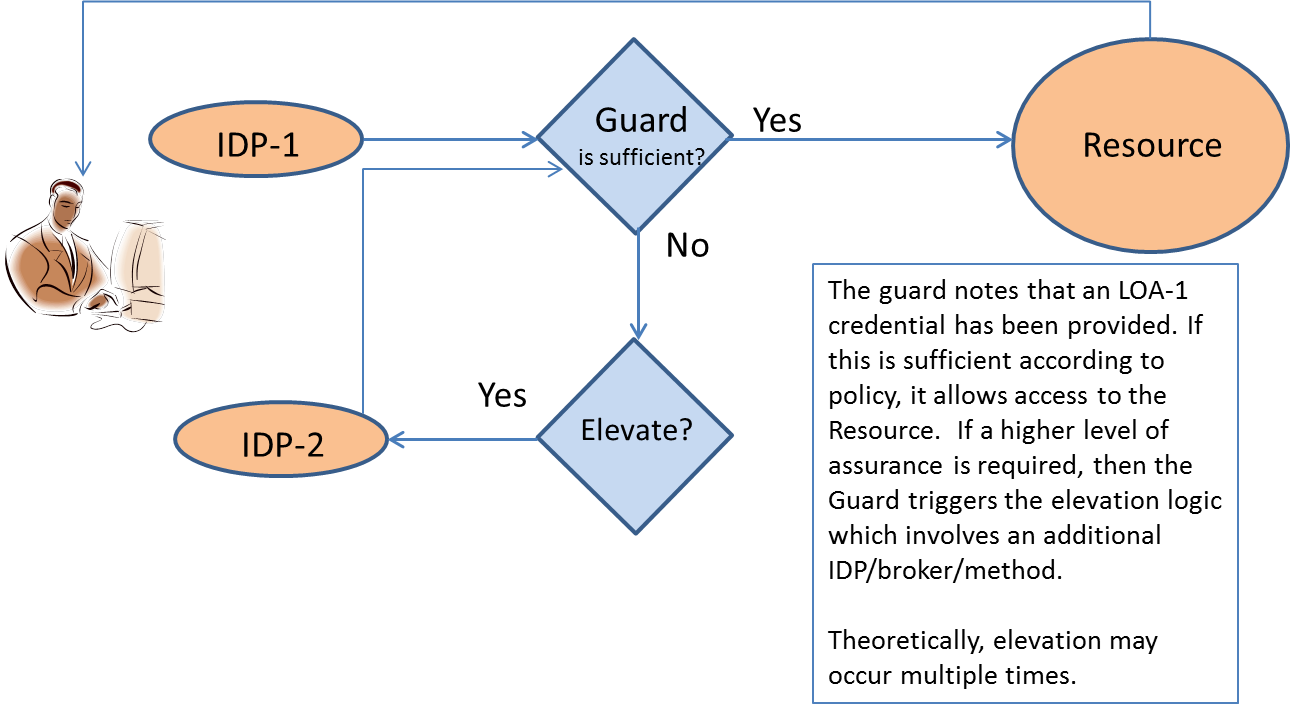


Figure 1 Trust Elevation Concept

The TC’s second deliverable is an analysis of the methods identified in the first deliverable to determine each one's ability to provide a service provider with assurance of the submitter's identity sufficient for elevation between each pair of assurance levels, to transact business where material amounts of economic value or personally identifiable data are involved. After analyzing which specific threats are countered by a specific trust-elevation method, we will be able to determine which methods, when combined will reduce overall threats and which methods when combined leave a service provide vulnerable to this same threats. Thus the vulnerabilities list is a key part of the analysis.

## Analysis Approach

We began by constructing a list of known vulnerabilities and formatting them into an analysis template. We referenced the threats identified by ITU-T x.1254 [3] as a starting point. We then used the template as a basis for analyzing several of the trust-elevation methods from the various categories. Upon reviewing these analyses, we determined that for our purposes, the vulnerabilities in our analysis template needed to be much more granular in order for our approach to be effective and therefore added more detail to the vulnerabilities list.

### Vulnerabilities are a moving target

There is an ongoing arms race in the area of computer security and identity management. As countermeasures are developed and widely deployed, for example adding simple knowledge based authentication (KBA) questions to a user interaction to mitigate the threat of a stolen credential, attackers look for ways to thwart the mitigation (e.g., finding your mother’s maiden name on a social networking site.) Thus best practices in vulnerability definition and threat mitigation are constantly evolving. When documenting a vulnerability (threat) mitigation, we assume that current best practices for that mitigation method are being used.

### Analysis Template (Vulnerabilities list)

The final Analysis Template is show below in Table 1.

|  |  |
| --- | --- |
| *Trust Elevation Method:* |  |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. |  |
| How does the method improve trust? |  |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? |  |
| Use of Latency examination? |  |
| Use of One Time Pads? |  |
| Use of Application layer encryption for transport (SSL/TLS)? |  |
| Use of dedicated transport layer? |  |
| Other? | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? |  |
| Use of Complex credential? |  |
| Use of a second factor for privileged access? |  |
| Use of Incremental delay for each failed attempt? |  |
| Use of session token? |  |
| Other? | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? |  |
| Use of One Time Password (OTP)? |  |
| Use of Nonce with Message Authentication Code? |  |
| Use of Timestamp? |  |
| Other? | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile |  |
| Use of PKI? |  |
| Use of Strong mutual authentication (PKI)? |  |
| Use of Latency examination? |  |
| Use of Second (secure) channel verification? |  |
| Use of One-Time Pads? |  |
| Use of Carry-Forward Verification? |  |
| Other? | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? |  |
| Use of Verified Session Tokens? |  |
| Use of dedicated transport layer? |  |
| Other? | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
|  | |
| How does the method address the threat of theft? | |
| Use of Password? |  |
| Use of Biometrics? |  |
| Use of nonce? |  |
| Use of second factor? |  |
| Other? | |
| How does the method address the threat of phishing? *Spear Phishing, Clone Phishing, Whaling.* | |
| Use of Out of Band verification? |  |
| Other? | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? |  |
| Other? | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01])   Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? |  |
| Use of Session Key? |  |
| Use of Dynamic Cookies (values change for each)? |  |
| Use of ARP Handler Inspection? |  |
| Use of Session Analyzers? |  |
| Other? | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? |  |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? |  |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

When completed, the analysis temple highlights which threats a method mitigates and which mitigation techniques it uses so that once a group of methods have be analyzed using this common template, it will be possible to assess the extent to which a set of methods are complementary. In other words, the extent to which the methods in the set mitigate different risks, so that overall risk is reduced when the methods are used together.

This study categorizes the methods according to the four generally-accepted identity management factors: what you are; what you know; what you have; what you typically do; plus a fifth category of context.

Analysis

The following are the completed analysis templates for the trust elevation methods identified in phase 1. The analyses were intentionally performed at the level of each individual method, rather than on combinations of methods.

## Method Category – What You Are

### Physical Biometrics

|  |  |
| --- | --- |
| *Trust Elevation Method:* | Physical Biometrics is the automated recognition of individuals based on their biological characteristics [ISO/IEC 2382-37].  Examples include:   * Physical: Fingerprint/palmprint, facial recognition, iris recognition, hand geometry. * There are 2 processes involved in the use of biometrics:   + Enrollment (registration) – where the biometric data is captured, processed, and securely stored.   + Verification (authentication) – where a newly captured biometric sample is processed and compared against the previously enrolled sample (based on an assertion of identity) to determine if the biometric sample originated from the same source (human being).   [Note: There is a 3rd process known as 1:N (identification); however, this is seldom used within an authentication context.]  Biometric matching is a statistical (probabilistic rather than deterministic) operation. Determination of a match is based on comparing the comparison score against a threshold value. The strength of function is proportional to this value, though increasing the threshold, while reducing the false match rate, increases the false non-match rate.  The primary advantage of biometric methods are that they link the authentication event to an individual human being (rather to something they know or have which can be shared), thus providing a potentially higher level of non-repudiation.  Threat models against biometric systems are dependent upon the architecture implemented, primarily the location where the storage and matching occur (i.e., within a central server, local workstation, device, or physical token). In addition, presentation attacks (sensor spoofing) are a threat unique to biometrics. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Architecture dependent – usually the identity provider. |
| How does the method improve trust? | Trust is improved by tying the authentication event to a particular human being. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? Answer is architecture dependent:  No – if the biometric data is transmitted across the network (i.e., in the clear).  Yes – if the biometric matching is performed locally. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | Guessing a biometric is more difficult than guessing a password, not because of its intrinsic entropy but because the generation of biometric samples to utilize in the attack is more difficult, as is their injection. |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? See above. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | Is compatible with this technique. |
| Use of Timestamp? | No |
| Other? This method does not directly address the threat of replay attacks unless used in conjunction with authenticity/integrity techniques such as ACBio (ISO/IEC 24761) or nonces. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? This method does not directly address the threat of man in the middle attacks unless used in conjunction with authenticity/integrity techniques such as ACBio (ISO/IEC 24761). | |
| How does the method address the threat of spoofing and masquerading? | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? This method is vulnerable to spoofing unless liveness detection techniques (countermeasures) are employed. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| Trust can be improved through the use of:  - Multifactor/multi-method authentication  - Cryptographic protection of the biometric data in transit and at rest  - Storage of the biometrics within secure hardware  - On card biometric comparison  - Authenticity/integrity protections (such as ACBio).  - Liveness detection/anti-spoofing techniques.  - Choice of algorithms independently tested to achieve high matching accuracy (low FMR).  [Note: These are not requirements, per se, but potential approaches to improving trust.] | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | Yes |
| Use of nonce? | No |
| Use of second factor? | No |
| Other? Biometrics can provide an additional factor. | |
| How does the method address the threat of phishing? *Spear Phishing, Clone Phishing, Whaling.* | |
| Use of Out of Band verification? | No |
| Other? | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? Not applicable. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Biometric data is generally considered PII and appropriate confidentiality protections (e.g., encryption) need to be applied. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | In general, convenience is high for biometrics; however, for a particular biometric modality, a small subset of the population may experience difficulty enrolling (finite failure to enroll rate).  Usability varies by biometric modality and specific implementation.  Exception handling provisions are necessary for those who are unable to enroll. Multi-biometric approaches are one method of addressing this issue. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

### Behavioral Biometric

#### Behavioral Biometric

|  |  |
| --- | --- |
| *Trust Elevation Method:* | Biometric behaviors are those based on the person’s physical behavioral activity patterns. These include:  • Keyboard signature (can be used for continuous trust elevation. Keyboard signature techniques may be adaptive);  • Voice (is both physical and behavioral). |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The Relying Party. For keyboard signature, software must be installed on the computer. Generally there is no other party involved. This method depends on information that the RP has gained via previous accesses to its own site. |
| How does the method improve trust? | The Relying Party collects data on the user’s keyboard signature, sometimes the signature used to type a password. The system compares current behavior to prior behavior patterns. This method provides an additional opportunity to detect impersonations. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via eavesdropping, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | Guessing a biometric is more difficult than guessing a password, not because of its intrinsic entropy but because the generation of biometric samples to utilize in the attack is more difficult, as is their injection. |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? If a person succeeds in fraudulently gaining access via online guessing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | Is compatible with this method. |
| Other? If a person succeeds in fraudulently gaining access via a replay attacking, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. However a sophisticated attacker may also be able to replay enough of the user’s behavior to elude the browsing patterns analytics engine. The browsing pattern analytics engine should check the dates on the browsing pattern data. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? If a person succeeds in fraudulently gaining access via a man in the middle attack, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via spoofing and masquerading this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method does not work on the first visit to a site. It works better after many visits. It may work less well on relatively simple sites, as there is less scope for possible variations. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | Yes |
| Use of nonce? | No |
| Use of second factor? | No |
| Other? This method works even if all credentials are stolen because it does not rely on any credential. | |
| How does the method address the threat of phishing? *Spear Phishing, Clone Phishing, Whaling.* | |
| Use of Out of Band verification? | No |
| Other? If a person succeeds in fraudulently gaining access via phishing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional verifications can be performed. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method could potentially detect session hijacking if the hijacker engages in different behaviors than the actual user, as long as the session hijacking doesn’t include replay of browsing patterns. Attacks using replay methods could be thwarted by checking timestamps. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | If the RP’s site was hacked, the biometric signature information could be captured. This information would typically not include any PII, but might be linked to PII on the RP system. Though it can require software to be installed on the user’s computer, keyboard signature, this is a method that can be active without the user knowing it. So it could remove the ability to visit a site pseudonymously. While credentials may use aliases, it may not be possible to have pseudonymous signature in some implementations. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method may not require any action on the part of the user for keyboard signature so its usability is high. Voice signature does require an extra step. If an anomaly is detected, then the user may be challenged to further identify them self. If the algorithm is tuned to be too sensitive there could be too many unnecessary challenges. This is often an issue with voice signature, as background noise can be a problem. If the user is sick or tired or has a cold or their fingers are cold or injured, etc., there can also be false negatives. This method does add system overhead. Keyboard signature can be performed one time or continuously. The analysis engine will require many exits. There may be many false negatives, requiring additional user action. There is a potential for multiple false negatives requiring multiple additional user actions in a single session. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

## Method Category – What You Know

### Password

A password can be requested as an additional factor.

|  |  |
| --- | --- |
| *Trust Elevation Method:* | Password: Access to network resources can be controlled through the use of a User Name / Password combination. A network, an application, a website, a service or an operating system can use password for granting permissions and user rights based on a user account that associated with the resource. This technique fits into providing something a user know (i.e. the password). A major problem with depending only on passwords for authentication is that more than one party can know the password. Sometimes, this is intentional, however, most of the time it’s not. A system administrator cannot be sure on who access to the password besides the intended owner. After a password is compromised, its original owner isn’t the only person who can access the system with it. As such passwords are one of the weakest links in the information security.  In general, the account administrator assigns a password to a user account at the time of account creation. Typically, the user is prompted to change the password after the first login.  Passwords can be stored in a database locally or at network authentication server. The database contains a list of user accounts and the corresponding passwords that are stored in a hashed from (generally salted and hashed).  Password fits in the category of something you know. A user or a system cannot tell if a hacker has knowledge of the password. There are multiple methods for a hacker to obtain access to a user password. Methods include:   1. Exploitation of weak passwords    1. A hacker that knows something about the user may be able to guess the password    2. Dictionary methods can also be used    3. Brute Force methods can be used 2. Exploitation of user behavior    1. Users may write passwords on their system if passwords are difficult to remember    2. Users may share the password across systems and social networks    3. Users may use patterns to help them remember password    4. Passwords could be guessed, seldom changed, forgotten, shared and written down and subsequently lost or stolen 3. Credentials theft    1. Hackers may be able to capture passwords when they are sent in the clear or hashed across the network    2. Passwords are vulnerable to man in the middle, replay, session hijacking and man in the browser attacks    3. Hackers can exploit weak password recovery implementations    4. Hackers can impersonate users and get password from help desk 4. Two general classifications of password vulnerabilities:    1. Organizational including end-user vulnerabilities       1. Some organizations lack good passwords policies and internal standards.       2. Organizations may lack password awareness and best practices for the part of end users.    2. Technical vulnerabilities Include:       1. The use of weak encryption methods and salting and hashing for password storage       2. The use of insecure storage of passwords on computer systems.       3. End-user applications that display passwords on the screen while typing. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Relying Party/ Identity Service Provider/ Network |
| How does the method improve trust? | In the context of trust elevation, presentation of a userID/password pair functions as a secondary “what you know” factor which may be added to the initial authentication event. The usefulness of the userID/password pair is dependent on the policies and practices under which the userID/password pair is issued and managed. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? N0. Method is vulnerable to eavesdropping. Method requires the use of TLS, VPN or out of band measures to mediate the risk | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | Yes |
| Use of Complex credential? | Yes |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | Yes |
| Use of session token? | No |
| Other? This method is vulnerable to online guessing. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? No. This method is vulnerable to replay attacks. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? No. This method is vulnerable to man in the middle attacks | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? No. This method is vulnerable to spoofing and masquerading | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| In order to elevate trust and provide counter measures, organizations need a multi-faceted defense against password vulnerabilities including:   * Mandating the use of secure passwords   + Don’t use words that are in the dictionary   + Mix upper and lower case alphabetic characters, numbers, and symbols in the password.   + Prevent password re-use * Mandating that passwords be changed regularly   + Don’t use the same password or the same two or three passwords over and over when it’s time to change passwords.   + Always hash passwords. Use a hashing algorithm designed for passwords, with cryptographically random salt.   + Do not allow or support password retrieval. Implement a secure password reset process supported by a policy.   + Do not limit password length. Longer passwords increase entropy and such security   + Send hash of passwords using encrypted network links   + Always keep up with system patches to reduce vulnerabilities | |
| How does the method address the threat of theft? | |
| Use of Password? | Yes |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | No |
| Other? | |
| How does the method address the threat of phishing? *Spear Phishing, Clone Phishing, Whaling.* | |
| Use of Out of Band verification? | No |
| Other? No. This method is very vulnerable to phishing. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? No. Passwords are easy to duplicate, once their contents are known. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? No. This method is vulnerable to session hijacking. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | In general there are no privacy issues, unless the user crates a password that contains PII. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | Users are very familiar with using passwords.  People may forget passwords or write them down. More complex passwords are more difficult to generate and remember. The more passwords one has, the more difficult it is to manage these passwords. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? | This is a single factor regardless of the password strength. When performed with a previously registered phone that is subscribed to by the user, it can constitute an acceptable second factor. |

### KBA

Note that this analysis covers both dynamic and static KBA issues.

|  |  |
| --- | --- |
| Trust Elevation Method: | Knowledge based authentication (KBA) is an authentication scheme that asks a user one or more secret questions in order to confirm the users identity. This type of authentication is often used as a component in multifactor authentication (MFA). KBA schemes use static or dynamic information in order to help compose the “secret” question for the users.  KBA is widely used in self-service password retrieval requests. KBA may also be used as part of the identity proofing process performed on a user before an authentication credential is issued to the user to improve trust in the user’s identity. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The end user answers the KBA quiz. The KBA quiz is presented by the relying party company and may leverage some of the following KBA data sources to generate the quiz.   * Third party non-public data sources such as professional society proprietary data or government entity proprietary data; * Data aggregators. The quality of the data from data aggregators varies greatly. Those employing competent methods for scoring the accuracy of the data they resell, and whose high-scoring data sets provide a greater assurance of accuracy, provide greater trust; * Multiple third party knowledge-bases that have been cross-checked for accuracy; * User relationship transactions (local prior history) that are not generally available to public databases or social networking sites; * User-data procured at enrollment |
| How does the method improve trust? | This method provides an additional opportunity to detect impersonations and confirm the identity of the user. The trustworthiness of this method generally relies on two factors, the accuracy of the answers stored by the KBA service provider and the strength of security practices applied to protect the databases storing the answers. These questions may be based on information from public, private and proprietary data sources.  Using KBA as an authentication mechanism will only provide up to NIST LOA 2 (as referenced in NIST SP 800-63).  Use of a dynamic KBA with multiple questions which all must be answered correctly within a specified time period has the potential to elevate trust. Limiting the number to tries that a user can take the KBA will also elevate trust. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? This method is subject to eavesdropping but this threat can be mitigated by using a dynamic KBA whose questions are not repeated within a specified time period (e.g., 30 minutes) and more than one dynamic question is asked (e.g., 5 questions) which all must be answered correctly. Dynamic questions should change on each try. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | Yes |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | Yes |
| Use of session token? | No |
| Other? Dynamic KBA is safer than static KBA as static KBA is subject to successive guessing attacks. Limiting the number to tries to answer questions correctly, asking non-wallet questions (e.g., of the following names which have you not lived with) and asking multiple questions (e.g., 5 questions) which all have to be answered correctly, will also limit guessing attacks. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? Static KBA is subject to replay attacks if the question does not change for each session or each time the question is answered incorrectly.  Dynamic KBA is less likely to be subject to replay attacks if the dynamic questions are not repeated within a specified time period (e.g. 30 minutes) and more than one dynamic question is asked (e.g. 5 questions) which all must be answered correctly. If multiple tries are allowed (e.g. 2 tries) the dynamic questions should change on each try to mitigate this threat. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? If a person tries to fraudulently gaining access via a man in the middle attack, this method may be used to verify the identity of the user. A dynamic KBA is less likely to allow a person using a man-in-middle attack to gain access if parameters are set for the KBA such as requiring 5 questions to be answered correctly. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? If a person tries to fraudulently gain access via a spoofing or masquerading attack, this method may be used to verify the identity of the user. A dynamic KBA is less likely to allow a fraudulent person to gain access if parameters are set for the KBA such as requiring 5 questions to be answered correctly. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| A static KBA relies on questions and answers that do not change and which have been collected during previous contact with the user. This method becomes increasingly insecure over time.  A dynamic KBA uses knowledge questions to verify a user’s identity, but requires no previous contact. This is because the questions are generated on the fly and based on information in a user’s personal aggregated data file (using public, private, and proprietary records), complied marketing data, or credit report. The dynamic KBA is a better option for improving trust.  Static or dynamic KBAs should be subject to implementation requirements to mitigate risks. For Static KBAs   * Limit time for the entry of a response * If the user fails to answer the question correctly the question should not be repeated in the same session * To further increase the level of assurance of the user, ask more than one question. * For dynamic KBA * Limit time for entry of responses * Questions should not are not repeated within a specified time period (e.g. 30 minutes) * More than one question should be asked (e.g., 5 questions with multiple choice answers). Set it up like a quiz present 5 questions on a page. * All questions must be answered correctly or set a limit (e.g., 4 out of 5 questions must be answered correctly) * If multiple tries are allow (e.g., 2 tries) the questions and multiple choices, if these is used, should change on each try to mitigate this threat. * The KBA matching process should be capable of objectively scoring the information provided, so that the potential absence of certain information does not necessarily result in the inability to authenticate a particular person. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes |
| Other? Static KBAs could be subject to threat of theft. Leveraging a dynamic KBA using more than one dynamic question (e.g., 5 questions) which all must be answered correctly will mitigate this threat. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? Static KBAs may be subject to a phishing attack. Leveraging a dynamic KBA using more than one dynamic question (e.g., 5 questions) which all must be answered correctly will mitigate the phishing threat. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? KBA works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method could potentially detect session hijacking, as long as the session hijacking doesn’t include replay of KBA, which can be addressed with a dynamic KBA. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | There are privacy and confidentiality issues.  For companies that collect the knowledge-based questions/answers from the user during registration, consent should be obtained before these question/answers are obtained. Companies must strike the right balance to avoid prying questions that ruffle the feathers of privacy advocates and that could potentially turn away customers.  KBA questions obtained through a third party service provider may contain private data and companies and other users of the technology must use this data responsibly or risk facing a loss of trust with their customers.  Note: Outside the US, Third Party Service Providers do not offer a KBA service, as there is not enough access to private and public data sources to create a KBA quiz due to privacy laws. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | There are usability issues with KBA.   * How many questions should be used? * How many questions does the user have to be answered correctly? * Are prepared questions used at registration, or can the user make up their own questions? * If users make up their own questions how you do stop offensive questions being entered? * Can the actual person answer correctly? * How many times will you allow the user to try to answer the questions? * How do you ask the right questions without offending the user? * Dynamic or Static KBA? * KBA based on public information is not available outside US. * The system must provide an alternate path for authorized users who are not able to pass the KBA screen. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

## Method Category – What You Have

### End Point Identity

End Point Identity is an umbrella term that describes any of a number of sub-methods used to identify the device by which the user is accessing the service provider. Transactions are considered less risky when a known device is used. If an unknown device or a compromised device is used, then additional methods may be required to mitigate risk. Gathering information about an end point device is sometimes called device fingerprinting. End point identity attributes may include IP address, router number, mobile phone number and/or SIM and/or OS, provider, cookie, browser, chip. The end point identity chain of trust can also include software tokens/digital certificates that provide information about other layers of the data stack.

#### Cookie, OS, Browser

|  |  |
| --- | --- |
| Trust Elevation Method: | Cookie: A cookie (HTTP cookie or a browser cookie) is a short piece of data sent by a website and stored in a client’s web browser when a user browses a website. Data stored in the cookie can be retrieved by the website on subsequent revisits. Cookies are used as a means of storing state information of the website than may include activities the user had performed in previous interactions.  In general cookies cannot carry viruses or install malware on the host compute. However tracking cookies including third party tracking cookies are commonly used as ways to compile long-term records of individuals' browsing histories are a major privacy concern. Keys or security information stored in a cookie also present security risks.  A client can easily modify, add or delete information from a Cookie, as such; web sites should treat data from Cookies as un-trusted user’s input. Cookies are prone to sniffing in an HTTP request. Malicious internet services can intercept or modify cookies. The use of SSL should mitigate this risk. The relying party checks whether it has previously set a cookie on the end point computer accessing it. If it finds the cookie, it knows that it has authorized that computer access to its service(s) before. This obviates concerns over different IP addresses. If it doesn’t find the cookie, it can set one for the next time should circumstances warrant.  OS, Browser: The relying party checks to see what OS and browser the end point computer is using. This can reveal many things, such as vulnerability to malicious code which may threaten the RP system. It also serves as a check to see if the same person on the same computer was using the same OS and browser as previously, thus providing added incremental confidence that the customer is the person his/her logon claims to be. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The relying party. A website server set a cookie on a client machine when the browser connects for the first time.  The browser returns a copy of the cookie to the server each time it connects. |
| How does the method improve trust? | A cookie confirms that the same computer with the same OS and browser and with the same asserted user was authorized access to the relying party previously with no adverse effects.  This method enable a server or site owner to provide better service to the user by remembering information about the user such as preference, shopping habits etc. On the other hand Cookies can be used for more controversial purposes, for example. Each access a browser makes to a Web site leaves some information about the client behind, creating a gossamer trail across the Internet. Personal data such as device name and IP address of your computer, the brand of browser, the operating and the URL of the Web page being accessed. This may cause privacy issues to some users.  Cookies can have security implications. Many sites use cookies to implement access control schemes that may require the use of a user name and password. The site may store this login information in a cookie and might pass a cookie back to the client browser on re-visits. Such methods are vulnerable to various means of cookie sniffing, stealing and modification and are not very secure and can be hijacked. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | Yes |
| Use of dedicated transport layer? | No |
| Other? If the cookie is marked secure, then only the issuing domain should be able to access it, however, vulnerabilities remain. This method is subject to eavesdropping but this threat can be mitigated by using SSL or TLS in addition to encrypting the cookies. Also, some enterprise policies erase cookies with some regularity. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | The cookie, OS or Browser act as an additional credential. |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? This method does not contain information that can be guessed. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? N/A | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? N/A | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? Vulnerable: Cookies are simple text files that can be manually installed on a computer; OS and browser are more or less likely to change over the lifetime of the computer and so differences in them do not automatically raise alarm. There is no direct connection between the identity of the end user and the end point device. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| None. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | No |
| Other? Will detect theft of credential other than platform. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? This can help as the cookie, OS, browser can operate as an additional factor. However it is still vulnerable: Cookies are simple text files that can be manually installed on a computer; OS and browser are more or less likely to change over the lifetime of the computer and so differences in them do not automatically raise alarm. There is no direct connection between the identity of the end user and the end point device. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This works if other credentials are duplicated because it does not rely on any credential other than those that are part of this method. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | Yes for cookies |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? No. This does not help with session hijacking once the session is established. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | A cookie is a small text file that is written to the end point computer by the relying party. The application should request permission before writing anything to a customer’s computer.  OS, Browser: N/A |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method will not work if browsers are configured to not support cookies. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? | Verifying machine fingerprint including IP address along with cookies will mitigate threat. |

#### IP Address, router

|  |  |
| --- | --- |
| Trust Elevation Method: | IP address: Each computer connected to the Internet has either a permanent or a temporarily-assigned Internet Protocol (IP) address. This address allows the Internet middleware to connect the computer to other devices connected to the Internet. Because at this level of abstraction, all connections are point-to-point and these points are defined by IP addresses, it is a straightforward task for software to discover the IP address of another computer. This allows a relying party to identify the computer that is attempting to connect to its service(s); if it has seen this IP address before, it has incrementally greater confidence that the end user employing the computer is the person his/her credential claims him/her to be.  Router: In order to extend the pool of available IP addresses in the Internet universe, dynamic assignment of IP addresses is a commonly-used technique. In this scenario, an end point computer has no permanent IP address, but one is dynamically assigned to it when it attempts to connect to the internet. Dynamic assignment is performed by an intermediate device, usually a router. The router needs to have a permanently-assigned IP address. Should a relying party application see dynamically-assigned IP addresses for an end point computer used by a single individual that come from a common range of numbers, it can look to see if all the addresses have been assigned by the same router, it can reasonably assume that the same user is employing the same computer. This method is admittedly one step removed from direct association with the end point computer but has the benefit of expressing a permanent IP. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The relying party. |
| How does the method improve trust? | This method looks past the inputs of the human customer to the device that is being used on his/her end. The method determines the kind of device being used and one or more components (hardware and software) to determine if the application service provider has ever seen that unique device in the past associated with successful login by that customer. If a previous correlation is found, the method raises the confidence that the customer is who his/her credential asserts he/she is combined with the device identity as the second factor. If there is no previous correlation with that customer and device, and there is correlation with many failed login attempts, then the confidence is significantly reduced. If the device is not recognized, then confidence is reduced and additional methods may be required to raise confidence back to acceptable levels.  IP address, router: In general, end user applications such as web browsers do not allow users access either to the computer’s IP address or to the router’s IP address. This separates the user from the transport service so that he/she cannot claim to be operating from someone else’s computer. He/she can still claim to be someone else, of course, and can do that from the other party’s computer, but that is an inherent weakness in this elevation mechanism. More generally, this method is not able to make a strong connection between the computer with IP address and the actual operator at the keyboard. Also, software is widely available that allows a knowledgeable user to spoof an IP address. The router makes an even weaker claim to connecting operator from device, though it is incrementally harder for an end user to spoof an infrastructure device, in part because a responsible network operator will have intrusion detection services enabled and monitored. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? N/A | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | The IP address, router act as a weak additional credential. |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? This method does not contain information that can be guessed.  IP address, router: Since IP address, router numbers are easy to look up on and off the Internet, this method is vulnerable to informed guessing. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? Last login is tracked in cloud and the device identity is augmented with the credential associated with last login. May also help with replay attacks if the device identity includes the time on the device clock.  IP address, router: N/A | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? N/A  IP address, router: Vulnerable, see below. N/A | |
| How does the method address the threat of spoofing and masquerading?  • SMS spoofing  • IP address spoofing  • CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? TBD  IP address, router: Vulnerable, because intrusion devices/software can spoof IP addresses, even of network devices and masquerade as a router or end point computer. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method assumes that there is a persistent one-to-one relationship between the login data and the device used at the customer end. Moreover, there is the possibility of aliased device ID for each user addressing shared devices challenge. Besides per user device ID, a per service/app device ID to segregate the device identity provisioned for a financial institution from a social network or content provider. Prior registration of that relationship could improve trust, since that information could serve also as a shared secret. Agreement by the user not to share the device could improve trust, perhaps, by providing some confidence that no other user's login credentials were affiliated with that device. Because users do sometimes change devices and devices are not always exclusive, this method can result in too many false negatives, unless as we have noted above, there is a unique binding of the user to a distinct device identity and service. In many cases, such as use of a family computer by many members of a household or as a public kiosk, this would be impractical.  IP address, router: The usefulness of this method is substantially enhanced when the network infrastructure on both ends of the transaction has enabled a full suite of anti-spoofing, anti-intrusion, cyber protection tools and security practices. As previously noted, these implementations cannot protect end points from a determined and sophisticated attack, but they can substantially mitigate risk for the general user population. Improved binding of the operator to the end point computer may be done through high-assurance authentication of operator to device, for example, requiring two-factor authentication for a user to log on to his/her computer as is the evolving practice in the USG. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | No |
| Other? Will detect theft of credential other than platform. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? N/A | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This works if other credentials are duplicated because it does not rely on any credential other than those that are part of this method. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? Session hijacking can be prevented to a certain extent by verifying if a persistent cookie is dropped for a particular IP address or range. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Since the method does not engage the customer in the exchange of information, he or she may not be aware that the device attributes are even being sent. Mobile devices particularly can expose PII. Explicit consent for release of device data should be a prerequisite of application device query. If the device identity of each service is aliased and is unique, privacy risk is significantly reduced  IP address, router: For IP address, see above; for router identification, N/A unless the network hosting the router is a private one. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method is particularly well-suited to customer usability as it requires little or no user interaction for the typical session. This method is typically used in conjunction with an additional method such as relationship-based KBA.  IP address, router: N/A |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

#### Phone Number

|  |  |
| --- | --- |
| Trust Elevation Method: | Phone Number: This attribute is the telephone number and/or account number assigned to an individual subscriber by a provider of direct-wired telephone services. Where the account is assigned to a business or other non-individual entity, its potential role in individual trust elevation is extremely remote. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The relying party. |
| How does the method improve trust? | This method looks past the inputs of the human customer to the device that is being used on his/her end. The method determines the kind of device being used and one or more components (hardware and software) to determine if the application service provider has ever seen that unique device in the past associated with successful login by that customer. If a previous correlation is found, the method raises the confidence that the customer is who his/her credential asserts he/she is combined with the device identity as the second factor. If there is no previous correlation with that customer and device, and there is correlation with many failed login attempts, then the confidence is significantly reduced. If the device is not recognized, then confidence is reduced and additional methods may be required to raise confidence back to acceptable levels.  Phone number: For use during a transaction. The RP asks the customer for his or her home phone number, historically a landline, and then looks up that number in a directory maintained by the Telco. If the name on the customer’s credential (assuming there is one) or the name the customer asserts to the RP at login and the name on the phone number and/or account match, then the RP can assume that the customer requesting authorization knows the phone number and/or account number and the likelihood that the customer is the entity he or she asserts he/she is increases. Since a landline has a street address associated with it as well, it carries intrinsically a second attribute. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | Yes in VoIP connection over SSL |
| Use of dedicated transport layer? | Yes. Landline would be using dedicated connection; cell phone would be using dedicated cellular tower connection (most mobile connections are encrypted); |
| Other? N/A  Landline: N/A | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | The IP address of the router acts as a weak additional credential. |
| Use of a second factor for privileged access? | No |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? This method does not contain information that can be guessed.  Landline: Since landline numbers are easy to look up on and off the Internet, this method is vulnerable to informed guessing. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? Last login is tracked in cloud and the device identity is augmented with the credential associated with last login. May also help with replay attacks if the device identity includes the time on the device clock.  Landline: N/A | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? N/A  Landline: Vulnerable. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? TBD  Landline: Vulnerable. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method assumes that there is a persistent one-to-one relationship between the login data and the device used at the customer end. Moreover, there is the possibility of aliased device ID for each user addressing shared devices challenge. Besides per user device ID, a per service/app device ID to segregate the device identity provisioned for a financial institution from a social network or content provider. Prior registration of that relationship could improve trust, since that information could serve also as a shared secret. Agreement by the user not to share the device could improve trust, perhaps, by providing some confidence that no other user's login credentials were affiliated with that device. Because users do sometimes change devices and devices are not always exclusive, this method can result in too many false negatives, unless as we have noted above, there is a unique binding of the user to a distinct device identity and service. In many cases, such as use of a family computer by many members of a household or as a public kiosk, this would be impractical.  Since landline number, and to a lesser extent account number, is easily discovered, and since it is easy for an individual to impersonate another on the phone, this attribute has little standalone value. It has value as a way of calling the customer at the time of login and using person-to-person KBA methods, voice recognition methods or other biometric methods of elevating trust. In this way it is an effective enabler of more reliable trust elevation methods. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes |
| Other? Will detect theft of credential/objects other than phone line. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? N/A | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This works if other credentials are duplicated because it does not rely on any credential other than the method used. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | Yes, in VoIP |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? N/A | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Since the method does not engage the customer in the exchange of information, he or she may not be aware that the device attributes are even being sent. Mobile devices particularly can expose PII. Explicit consent for release of device data should be a prerequisite of application device query. If the device identity of each service is aliased and is unique, privacy risk is significantly reduced  In a majority of cases, landline numbers are public knowledge. Mobile numbers are not generally public knowledge. Phone numbers have been categorized as PII in some jurisdictions (California.) |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method is particularly well-suited to customer usability as it requires little or no user interaction for the typical session. This method is typically used in conjunction with an additional method such as relationship-based KBA. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? | In smart phones, finger scan on the phone, and or fingerprint of the phone would help. |

### Hard Token

A user may be requested to provide a hard token as an additional factor. Frequently the hard token is used to generate an OTP. See also the section on OTP.

|  |  |
| --- | --- |
| Trust Elevation Method: | Hardware tokens include:  • Proprietary tokens;  • USB tokens;  • Smart Cards;  • Mobile phone and or SIM.  These HW tokens are often associated with a discrete physical device and used to prove one's identity electronically (as in the case of a customer trying to access their bank account). The token is used in addition to or in place of a password to prove that the customer is who they claim to be. The token acts like an electronic key to access something and service.  When combined with some sort of Crypto key generated by the device (i.e. device identity or service specific device identity), biometric and geo location data and a method of physical input (keyboard, RF, voice, video including gesture) can become very useful.  The biggest problem with such discrete keys/tokens is loss of the token generator (USB key or Smart card). Though more integrated class of keys such as SIM cards reduce this risk until an ISIM that is provisionable and can be partitioned is available, service keys (lockbox concept) is more employed. Besides loss or leaving them at home, the TCO and ease of use are amongst other challenges for large deployment of this class of solutions to mass of users. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Relying Party/ Identity Service Provider/ Network. |
| How does the method improve trust? | This method can improve trust if done properly. Trust is improved through the use of HW token |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | Yes for smart cards |
| Use of Latency examination? | Yes for RSA SecurID tokens |
| Use of One Time Pads? | Yes for RSA SecurID tokens |
| Use of Application layer encryption for transport (SSL/TLS)? | Yes for smart cards & mobile phones |
| Use of dedicated transport layer? | No |
| Other? N0. Method is vulnerable to eavesdropping. However, 2-3 factor authentication (2-3FA) can make the eavesdropping moot. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | Yes |
| Use of Complex credential? | Yes |
| Use of a second factor for privileged access? | Yes, with the use of a password |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? If part of a 2 or 3 factor solution it elevates the trust of platform and session | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | Yes for RSA SecurID tokens |
| Use of Nonce with Message Authentication Code? | Yes for RSA SecurID tokens |
| Use of Timestamp? | Yes for RSA SecurID tokens |
| Other? | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? This method is vulnerable to man in the middle attacks for mobile devices especially if the generated token is used all by itself. One solution that can reduce the risk significantly is to generate a session hash of the token generated. The hash can be tied to the device identity and user or even contextual info including Geo Location, time, user habits etc. (any element of 2FA-3FA ingredient) and the service specific (social website vs. bank) request. Such 2-3FA can significantly elevate the trust (dynamic token chain). | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | Yes for RSA SecurID tokens |
| Use of dedicated transport layer? | No |
| Other? This method can be vulnerable to spoofing and masquerading for mobile phones unless methods to improve trust are employed such as a session hash that is generated which is tied to the device and the service requested (some sort of smart dynamic token chain). | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| Elevating Trust and counter measures available  A number of multi-faceted defenses against token vulnerabilities exist:   * Use 2FA-3FA (Device ID and location, time, geo location, etc.) * Dynamic per service hashing of the token * SSL * Loss of HW token device is a big issue which is to be augmented by an integrated solution (Use of biometric and device ID, and secured storage/vault). | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | Yes for RSA SecurID tokens |
| Use of second factor? | Yes |
| Other? No. This method is vulnerable to theft. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | Yes for RSA SecurID tokens |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? No. This method is vulnerable. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Yes, mapping the token to an individual is assumed. Hence user consent is required.  Solution is to use 2FA-3FA and make the token service specific to contain the risk (e.g., token given to your bank vs. your social site). |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | Loss, forgetting to bring the token with you, requires drivers and access port on computer or other device. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

### Soft Token

A user may be requested to provide a soft token as an additional factor. Frequently the soft token is used to generate an OTP. See also the section on OTP.

#### Digital Certificates

|  |  |
| --- | --- |
| Trust Elevation Method: | Digital Certificates (X.509 PKI). Although digital certificates are most often used as primary credentials for authentication, encryption and/or signing, there is no intrinsic reason they cannot be used as secondary assertions of identity. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | A trusted Certification Authority issues the public and private keys, which being software, reside on a computer platform. The subscriber is responsible for use of the credential during transactions. |
| How does the method improve trust? | A secondary credential can reinforce the identity assertion(s) of the primary credential or assertion. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | Yes |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | Yes. Encryption digital certificates can be used to establish the TLS session.  Also, it is recommended that when digital certificates, or certain information related to those certificates, are transmitted across a network, that the channel be encrypted (e.g., with TLS). |
| Use of dedicated transport layer? | No |
| Other? The main reason that digital certificates (as an authentication method) address the eavesdropping threat is that the protocol uses cryptographic information in the cert within the authentication protocol (i.e., to verify proof of possession) without exposing that information over the network (where it could be eavesdropped). So the main strength here is lack of exposure, followed by difficulty of deciphering the message. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | Yes, The cryptographic technology of the digital certificate minimizes the threat of online guessing, though a lightweight key pair with obsolete hash algorithm is vulnerable to cracking. |
| Use of a second factor for privileged access? | Yes, Use of a second factor of what you are or what you have, if complex device identification is used rather than simple device identification which can be duplicated. |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? Not Applicable | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | Yes, can serve as session tokens |
| Use of One Time Password (OTP)? | Yes, can utilize digital certificate |
| Use of Nonce with Message Authentication Code? | Yes |
| Use of Timestamp? | Yes, strengthens time stamping substantially |
| Other? | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | Yes |
| Use of Strong mutual authentication (PKI)? | Yes with PKI, SecretKeys and Passwords |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | Yes in mutual SSL handshake |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? Digital certificate technology is a credential-based method and does not address these. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| The trustworthiness of cryptographic technology is very sensitive to implementation practices. Rather than describe implementation requirements, refer to the documentation of the Federal PKI Policy Authority through www.idmanagement.gov . | |
| How does the method address the threat of theft? | |
| Use of Password? | Yes, possible |
| Use of Biometrics? | Yes, possible |
| Use of nonce? | Yes, possible |
| Use of second factor? | Yes, possible |
| Other? The private key is protected by a password, biometric or nonce. Some tokens make the private key non-exportable and a second factor protection (password or PIN) of the token itself is an added layer of security. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? Digital certificate technology is a credential-based method and does not address these threats | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? Yes, Each certificate contains one or more unique public keys based on proven cryptographic technologies. Only the paired private key, which is stored separately, can “unlock” the distributable public key. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? Digital certificate technology is a credential-based method and does not address these threats. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | The concerns are the same as for any other credential-based alternative. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | Software applications are only now reliably enabled to use PKI for authentication and signing. Operating systems are now PKI-aware and there are evolving implementation options for the basic technology that make it more user-friendly. That said, the process of using a digital certificate for any of its functions should require validation, which requires infrastructure to discover the path to the issuer’s directory service and software to deliver the answer in a way the application can use. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? | Digital Certificate with finger print of the hardware where it is contained will help mitigate threat |

### Out of Band

|  |  |
| --- | --- |
| Trust Elevation Method: | Out of band refers to the use of a second, separate communications channel for the relying party to acquire a second factor for authentication from the end user or a previously-identified third party. Out of band may even function as a primary channel if the initial primary channel has been compromised. This method is only as secure as the second channel is secure. Variations include all manner of communications channels such as the following. Each variant has its own risk profile.   * User calls service provider from a registered phone; * Response to a phone call from the service provider; * Response to an email from the service provider; * Response to an SMS message from the service provider; * Response to a mobile application transaction initiated by the service provider; * Response to a post card; * Response to a letter, registered or otherwise. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Relying Party/ Identity Service Provider/ Network |
| How does the method improve trust? | This method can improve trust if done properly. Trust is improved through the use of a second independent channel. Since the reliability is wholly dependent on the reliability of the second channel, the effectiveness of the varying sub methods varies widely. Many of these methods explicitly include a second factor such as what you have (cell phone or access to a specific landline or mailbox locked in a specific building). |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? |  |
| Use of dedicated transport layer? | Yes in case of landlines |
| Other? This method addresses most eavesdropping. The email variant is vulnerable to eavesdropping on the email password. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No. |
| Use of a second factor for privileged access? | Yes, guessing when it involves an out-of-band approach that includes what you have. |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? This method addresses online guessing when it involves an out-of-band approach that includes what you have. When used with an insecure email address, it may not reduce this threat. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | Yes |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? This method solves the problem of replay attacks when a live person needs to be involved. Replay attacks could still be successful when a compromised email or mobile app approach is used. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | Yes |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? This method makes it much more difficult to conduct a man-in-the-middle attack. Multiple channels would need to be intercepted in a time coordinated fashion. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? This method makes it much more difficult to spoof. Multiple channels would need to be intercepted in a time coordinated fashion. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| It is important that the out-of-band method be as easy to use as practical to avoid user drop off. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes |
| Other? This method is only as secure as the second factor is secure. Cell phones can be stolen, email can be hacked, and physical mail can be intercepted. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | Yes |
| Other? | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method is only as secure as the second facto is secure. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? No. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | This method generally requires the user to share additional information (home address cell phone) that constitutes or can be readily linked to PII. On the other hand, home address, cell phone and email address is often already readily available. The various methods have various confidentiality risks.   * Information on post cards can be read * Letters can be stolen from some unlocked mailboxes without detection. * Information sent via email is only as secure as the email is secure (some emails are very insecure.) |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method is becoming increasingly common. It requires the user to perform additional steps. Many variants require access to specific devices or locations. It creates delays of varying length (minutes to days). Some users are reluctant to go to the extra effort unless they have previously experienced identify fraud or wish to perform a high value transaction. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

### One Time Password (OTP)

A user may be requested to provide an OTP. OTP’s are usually generated by a hard or soft token. See also hard and soft tokens.

|  |  |
| --- | --- |
| Trust Elevation Method: | One time password (OTP):  Access to network resources can be controlled through the use of a User Name / Password with the possible combination of One Time Password.  A one-time password (OTP) is a password that is valid for only one login session or transaction. OTP requires additional technology in order to work.  OTP are dynamic and avoid many of the shortcomings that are associated with static passwords.  In theory OTPs are not vulnerable to replay attacks.  There are many approaches for the generation of OTPs.  1. Based on time-synchronization between the authentication server and the client providing the password (OTPs are valid only for a short period of time)  • Time-dependent authenticators base their OTP generation on time intervals. A password is valid for a given amount of time.  • Can be implemented with small key cards that are equipped with digital displays. The card displays a unique numeric combination that has been randomly generated by the hardware token.  • To be authenticated, users enter their personal PIN numbers, followed by the current OTP displayed on their key card.  2. Using a mathematical algorithm to generate a new password based on the previous password (OTPs are effectively a chain and must be used in a predefined order).  • Generate of OTPs randomly. Some token generators allow the users to generate the OTP by entering a PIN.  3. Using a mathematical algorithm where the new password is based on a challenge (e.g., a random number chosen by the authentication server or transaction details) and/or a counter.  • Smart Devices that a user carries can be used to generate OTP. The token can be generated using software that runs on the smart device.  • OTP can be sent out through SMS to users or other out of band methods  • OTP over text messaging may be encrypted with variable vulnerabilities  SMS based OTP have vulnerabilities  • There are threats from hackers, phishing and stealing.  • The mobile phone operator becomes part of the trust chain.  • hackers may mount a MITM attack.  • OTPs can be printed on paper that the user is required to carry  Some providers offer web based methods for delivering one time passwords without the need for tokens.  • Example includes the use of pre-chosen categories from a randomly-generated grid of pictures. Each picture in the grid has a randomly generated alphanumeric character overlaid on it.  • User enters alphanumeric characters associated with the preset category to form the OTP.  OTP helps to eliminate a number of system vulnerabilities, such as password cracking, password sniffing, brute force attacks, and social engineering. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Relying Party/ Identity Service Provider/ Network |
| How does the method improve trust? | This method can improve trust if done properly. Trust is improved through the use of dynamic passwords. This is a single factor regardless of the OTP strength |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? No. Method is vulnerable to eavesdropping. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | Yes, guessing when it involves an OTP that includes what you have. |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? When implemented correctly, this method not vulnerable to online guessing. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | Yes |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? This method solves the problem of replay attacks. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | Yes, if the OTP is done using a secure channel |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? This method makes it more difficult to conduct a man-in-the-middle attack. Multiple channels would need to be intercepted in a time coordinated fashion. This method is somehow vulnerable to man in the middle attacks. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | Yes |
| Use of dedicated transport layer? | No |
| Other? This method is vulnerable to spoofing and masquerading. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| Organizations need a multi-faceted defense against password vulnerabilities   * Use a secure OTP method | |
| How does the method address the threat of theft? | |
| Use of Password? | Yes, dynamic password |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes |
| Other? This method is only as secure as the second factor is secure. Cell phones can be stolen, email can be hacked, and physical mail can be intercepted. When performed with a previously registered phone that is subscribed to by the user, can constitute a second factor hard token. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | Yes |
| Other? This method is vulnerable to phishing. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method is only as secure as the second factor is secure. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method is vulnerable to session hijacking. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? |  |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method is very common with users. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

## Method Category – What You Typically Do

What you typically do consists of attributes related to an individual’s repeated behaviors or behavioral habits. We differentiate this category from biometric behaviors that have a physical component such as key signature and voice. It includes attributes related to correlation or statistical analysis of time of access (time of day, day of week, day of month, time of year, duration of access.) This is a type of active or continuous authentication.

### Time of Access

|  |  |
| --- | --- |
| Trust Elevation Method: | A service monitors access time and compares it to the history of a user’s access time to establish a statistical correlation between this behavior and the identity of the user. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The Relying Party. Generally there is no other party involved. This method depends on information that the RP has gained about the user during prior accesses to its own site. |
| How does the method improve trust? | The Relying Party collects data on the user’s access history at the site and compares current behavior to prior behavior patterns. This method provides an additional opportunity to detect impersonations. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via eavesdropping, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | Yes, this could be considered an additional factor |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? If a person succeeds in fraudulently gaining access via online guessing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? If a person succeeds in fraudulently gaining access via a replay attacking, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. However a sophisticated attacker may also be able to replay the user’s behavior at an appropriate time to elude the analytics engine. The analytics engine should check a full date and time stamp. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? This method does not help with man-in-the-middle attacks. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via spoofing and masquerading this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method does not work on the first visit to a site. It works better after many visits and for persons with stable access patterns. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes, this could be considered an additional factor |
| Other? This method works even if all credentials are stolen because it does not rely on any credential. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? If a person succeeds in fraudulently gaining access via phishing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional verifications can be performed. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method could potentially detect session hijacking if the system checks full date and timestamps. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | This information would typically not include any PII, but might be linked to PII on the RP system. This is a method that can be active without the user knowing it. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method does not require any action on the part of the user so its usability is high. If an anomaly is detected, then the user may be challenged to further identify them self. If the algorithm is tuned to be too sensitive there could be too many unnecessary challenges. The analysis engine will require many exits. There may be many false negatives, requiring additional user action. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

### Browsing Patterns

|  |  |
| --- | --- |
| Trust Elevation Method: | This type of active or continuous authentication relies on an external entity monitoring the user’s browsing behaviors and cataloging unique patterns of visits. By its nature it is highly intrusive. Note that we also differentiating between this method and mouse patterns. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The Relying Party. Generally there is no other party involved. This method depends on information that the RP has gained via previous accesses to its own site. |
| How does the method improve trust? | The Relying Party collects data on the user’s browser history at the site and compares current behavior to prior behavior patterns. This method provides an additional opportunity to detect impersonations. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via eavesdropping, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | Yes, this could be considered an additional factor |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? If a person succeeds in fraudulently gaining access via online guessing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? If a person succeeds in fraudulently gaining access via a replay attacking, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. However a sophisticated attacker may also be able to replay enough of the user’s behavior to elude the browsing patterns analytics engine. The browsing pattern analytics engine should check the dates on the browsing pattern data. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? If a person succeeds in fraudulently gaining access via a man in the middle attack, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via spoofing and masquerading this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method does not work on the first visit to a site. It works better after many visits. It may work less well on relatively simple sites, as there is less scope for possible variations. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes, this could be considered an additional factor |
| Other? This method works even if all credentials are stolen because it does not rely on any credential. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? If a person succeeds in fraudulently gaining access via phishing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional verifications can be performed. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method could potentially detect session hijacking if the hijacker engages in different behaviors than the actual user, as long as the session hijacking doesn’t include replay of browsing patterns. Attacks using replay methods could be thwarted by checking timestamps | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | If data is collected that is not purely referential to the site than this could potentially violate privacy and be offensive. Since there is a large market for personal information, over time there will be an incentive to design systems that are more intrusive. Even if the method only collects data that is purely referential, there could still be confidentiality issues if these signatures became sophisticated enough to differentiate amongst users at a level similar to fingerprints. If the RP’s site was hacked, the browsing pattern information could be captured. This information would typically not include any PII, but might be linked to PII on the RP system. This is a method that can be active without the user knowing it. So it could remove the ability to visit a site pseudonymously. While credentials may use aliases, it may not be possible to have pseudonymous browsing patterns in some implementations. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method does not require any action on the part of the user so its usability is high. If an anomaly is detected, then the user may be challenged to further identify them self. If the algorithm is tuned to be too sensitive there could be too many unnecessary challenges. Ideally the analysis algorithms should be designed so that if the RP makes standard changes to their site, browsing behavior that was previously collected can still be used. Otherwise this method would stop working for a period, each time the website was changed, and create an unacceptably high level of false negatives. If independent of the website, there may be incentive to resell the data While not requiring any action from the user in the base case, this method does add system overhead.. The analysis engine will require many exits. There may be many false negatives, requiring additional user action. There is a potential for multiple false negatives requiring multiple additional user actions in a single session. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

### Mouse patterns

|  |  |
| --- | --- |
| Trust Elevation Method: | This method captures the user’s patterns of mouse movement within an online site, as opposed to collecting data on browser patterns. Where the mouse points, where mouse moves across the screen are the patterns of this method. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | The Relying Party. Generally there is no other party involved. This method depends on information that the RP has gained via previous accesses to its own site. |
| How does the method improve trust? | The Relying Party collects data on the user’s mouse patterns when accessing the site and compares current behavior to prior behavior patterns. This method provides an additional opportunity to detect impersonations. Comparisons of mouse characteristics gathered from frequent behaviors are more stable. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via eavesdropping, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | Yes, this could be considered an additional factor |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? If a person succeeds in fraudulently gaining access via online guessing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? If a person succeeds in fraudulently gaining access via a replay attacking, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. However a sophisticated attacker may also be able to replay enough of the user’s behavior to elude the mouse patterns analytics engine. The analytics engine should check the dates on the mouse pattern data. | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? If a person succeeds in fraudulently gaining access via a man in the middle attack, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? If a person succeeds in fraudulently gaining access via spoofing and masquerading this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional actions can be taken. | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| This method does not work on the first visit to a site. It works better after many visits. It works better if the user tends to perform the same or similar tasks as a portion of every visit | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes, this could be considered an additional factor |
| Other? This method works even if all credentials are stolen because it does not rely on any credential. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? If a person succeeds in fraudulently gaining access via phishing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional verifications can be performed. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method could potentially detect session hijacking, as long as the session hijacking doesn’t include replay of mouse patterns. Attacks using replay methods could be thwarted by checking timestamps. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Even though the method only collects data that is purely referential (e.g., how the user mouses over a picture link), there could still be confidentiality issues if these signatures became sophisticated enough to differentiate amongst users at a level similar to fingerprints. If the RP’s site was hacked, the mouse pattern information could be captured. This information would typically not include any PII, but might be linked to PII on the RP system. This is a method that can be active without the user knowing it. So it could eventually remove the ability to visit a site pseudonymously. While credentials may use aliases, it may not be possible to have pseudonymous mouse patterns. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method does not require any action on the part of the user so its usability is high. If an anomaly is detected, then the user may be challenged to further identify them self. If the algorithm is tuned to be too sensitive there could be too many unnecessary challenges. Ideally the analysis algorithms should be designed so that if the RP makes standard changes to their site, mouse patterns that were previously collected can still be used. Otherwise this method would stop working for a period, each time the website was changed, and create an unacceptably high level of false negatives. While not requiring any action from the user in the base case, this method does add system overhead. The analysis engine will require many exits. There may be many false negatives, requiring additional user action. There is a potential for multiple false negatives requiring multiple additional user actions in a single session. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

## Method Category – Context

This category and method has not been recognized as a traditional factor, however, evolving technologies and user behaviors have made it a significant element in the overall use of Internet-based services.

### Context

|  |  |
| --- | --- |
| Trust Elevation Method: | Context consists of any additional attributes relevant to the user or situation. Some of these are specific to other trust elevation methods and some are generic. Context-based trust elevation can adjust dynamically to the circumstances surrounding a transaction based on the risk mitigation needs of the relying party application, and, it only needs to be invoked when needed. Context includes, but is not limited to:   * Location; * Time of access; * Frequency of access; * Party; * Prior relationship ; * Social relationship; * Source and endpoint identity attributes such as   + Date of last virus scan   + IP address   + Subscriber identity module (SIM)   + Device basic input/output system (BIOS)   + Virus scan software version   + CallerID   + Cookie (presence and/ or contents); * Multi-channel combination; * Credential lifecycle attributes; * Certificate binding and or other chain of trust attributes; * Secure device with user specific disk allocation. |
| Questions: |  |
| Which party is performing the method? Include details of multiple parties and attestation where appropriate. | Relying party. |
| How does the method improve trust? | This method looks past the inputs of the human user to the attributes associated with the transaction. Attributes considered for trust elevation from one level to another is based on the assurance required for each level. Each assurance level determines the types and combination of attributes that will be needed to verify before elevating trust of the user. A human user is trusted more when the transaction initiates and occurs within a familiar location and during a familiar time frame of user activity. If the same user were to initiate a transaction from an unfamiliar location and during unexpected time frame, then trust in the individual’s credential and identity would be significantly reduced and the customer might need to present supplemental credentials to elevate trust. |
| How does the method address the threat of eavesdropping?  NOTE: 800-63 identifies these countermeasures for eavesdropping:   * Use tokens with dynamic authenticators where knowledge of one authenticator does not assist in deriving a subsequent authenticator. * Use tokens that generate authenticators based on a token input value. * Establish tokens through a separate channel.   And  Eavesdropping resistance – An authentication process is resistant to eavesdropping attacks if an eavesdropper who records all the messages passing between a Claimant and a Verifier finds it impractical to learn the Claimant’s token secret or to otherwise obtain information that would allow the eavesdropper to impersonate the Subscriber in a future authentication session. Eavesdropping-resistant protocols make it impractical for an Attacker to carry out an off-line attack where he or she records an authentication protocol run and then analyzes it on his or her own system for an extended period to determine the token secret or possible token authenticators. For example, an Attacker who captures the messages of a password-based authentication protocol run may try to crack the password by systematically trying every password in a large dictionary, and comparing it with the protocol run data. Protected session protocols, such as TLS, provide eavesdropping resistance.  NOTE2 – As eavesdropping is a passive attack, merely collecting the data, the active attacks in which that data is used (replay, MiM, etc.) are separate attacks and addressed elsewhere. (We have to be careful that we don't mix them up and address more than we should here.) | |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of One Time Pads? | No |
| Use of Application layer encryption for transport (SSL/TLS)? | No |
| Use of dedicated transport layer? | No |
| Other? Attributes collected and used outside of a transaction session could be used to indicate that he person is unlikely to be whom he or she says he/she is so that additional actions can be taken. Attributes collected during a session can be altered if they are not collected using a secure connection. | |
| How does the method address the threat of online guessing? | |
| Limit number of attempts within a time frame? | No |
| Use of Complex credential? | No |
| Use of a second factor for privileged access? | Yes, this could be considered an additional factor |
| Use of Incremental delay for each failed attempt? | No |
| Use of session token? | No |
| Other? This method does contain information that can be guessed, however, a requirement for a combination of multiple attributes for a particular assurance level would help to address the threat of online guessing. A risk algorithm could then combine and compare several content attribute values to baseline attributes to determine a “trust score”. This “trust score”, could then be compared to a predetermined trust threshold, which varies by assurance level. | |
| How does the method address the threat of replay attack?  COMMENTS: The usage of CIA mechanisms as TLSv1 throughout the whole application (e.g., the whole website) will avoid replay attacks. Note that in web service communications, where the usage of TLSv1 should be avoided [KLEINER01], WS-Security must be in place, with nonces, timestamps (e.g., in the SecureConversation token and in the message). | |
| Use of session tokens? | No |
| Use of One Time Password (OTP)? | No |
| Use of Nonce with Message Authentication Code? | No |
| Use of Timestamp? | No |
| Other? This method by itself will not address the threat of replay attack | |
| How does the method address the threat of man in the middle?  • Man in the browser  • Boy in the browser  • Man in the mobile | |
| Use of PKI? | No |
| Use of Strong mutual authentication (PKI)? | No |
| Use of Latency examination? | No |
| Use of Second (secure) channel verification? | No |
| Use of One-Time Pads? | No |
| Use of Carry-Forward Verification? | No |
| Other? A combination of multiple attributes - both session and non- session based (knowledge based) - would address the threat of man in the middle to a certain extent, but not completely. | |
| How does the method address the threat of spoofing and masquerading?   * SMS spoofing * IP address spoofing * CallerID spoofing | |
| Use of Mutual Authentication? | No |
| Use of Verified Session Tokens? | No |
| Use of dedicated transport layer? | No |
| Other? A combination of multiple attributes - both session and non-session based (knowledge based) would address the threat of man in the middle to a certain extent, but not completely | |
| Are there implementation requirements for improving trust? If so, what are they and why are they necessary? | |
| * The relying party needs to establish baseline attribute data before enforcing this method. This could consist of information that is collected during a registration process as well as permission to collect supplemental session and non-session attribute data. * A human who is just introduced to the relying party system may not be trusted as much as a human who has been interacting with the system for years. * The system may need some time to learn about the user before enforcing this method. | |
| How does the method address the threat of theft? | |
| Use of Password? | No |
| Use of Biometrics? | No |
| Use of nonce? | No |
| Use of second factor? | Yes, this could be considered an additional factor |
| Other? This method works even if all credentials are stolen because it does not rely on any credential. | |
| How does the method address the threat of phishing? Spear Phishing, Clone Phishing, Whaling. | |
| Use of Out of Band verification? | No |
| Other? If a person succeeds in fraudulently gaining access via phishing, this method may indicate that the person is unlikely to be whom he or she says he/she is so that additional verifications can be performed. | |
| How does the method address the threat of credential duplication? | |
| Use of asymmetric key cryptography? | No |
| Other? This method works even if all credentials are duplicated because it does not rely on any credential. | |
| How does the method address the threat of session hijacking?  Commons attacks to sessions are related to the   * Session key stealing (e.g., the attacker obtains the session key and impersonate the user) * Session key guessing (e.g., the attacker guesses the session key [GUTTERMAN01]) * Session token manipulation (e.g., in the case of SOAP Message Rewrite Attacks [GORDON01]) | |
| Use of Application Layer encryption? | No |
| Use of Session Key? | No |
| Use of Dynamic Cookies (values change for each)? | No |
| Use of ARP Handler Inspection? | No |
| Use of Session Analyzers? | No |
| Other? This method has the potential to detect session hijacking as long as the session hijacking doesn’t include replay of context information. Attacks using replay methods could be thwarted by checking timestamps. | |
| Are there privacy and/or confidentiality issues engaged when using the method, such as user consent for attribute release/exchange? Are there reasonable solutions for potential privacy impacts? | Yes. The relying party may collect sensitive information on the human user and the relying party needs to get consent from the user before collecting and using such data. Use of access controls and cryptography may reduce threat of loss of personal identifiable information, however, that does not mitigate the risk of collecting unauthorized data on an individual. Consent is required. Greater protection is required for PII than for general data. |
| What are the usability issues when using the method? Are there reasonable solutions for potential usability impacts? | This method is particularly well-suited to customer usability as it requires little or no user interaction for the typical session. However, the user needs to go through a one-time registration process to establish baseline attributes and the user needs to provide some kind of identity authentication to the system for each session before the method can be enforced. |
| Are there any other factors in the environment that might mitigate or exacerbate the threat? |  |

1. Acknowledgments

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

**Chairs:**

Abbie Barber, Bank of America

Don Thibeau, OIX

**Editors:**

Peter Alterman, SAFE-BioPharma Association

Shaheen Abdul Jabbar, JPMorgan Chase Bank, N.A.

Jaap Kuipers

Thomas Hardjono, M.I.T.

Mary Ruddy, Identity Commons

**Document Contributors:**

**Peter Alterman, SAFE-BioPharma Association**

Abbie Barber, Bank of America

Shaheen Abdul Jabbar, JPMorgan Chase Bank, N.A.

Mary Ruddy, Identity Commons

Shahrokh Shahidzadeh, Intel

Cathy Tilton, Daon

**Technical Committee Member Participants:**

Abbie Barbir, Bank of America

Anil Saldhana, Red Hat

Antonio Campanile, Bank of America

Anthony Bass, Lockheed Martin

Brendan Peter, CA Technologies

Brian Spector, CertiVox

Cathy Tilton, Daon

Carl Mattocks, Bofa

Colin Wallis, New Zealand Government

Dale Rickards, Verizon Business

David Brossard, Axiomatics

David Staggs, Jericho Systems

Dazza Greenwood, MIT

Don Thibeau, Open Identity Exchange

Doron Cohen, SafeNet

Ed Coyne, Dept Veterans Affairs

Gershon Jansen

Jaap Kuipers, Amsterdam

John Bradley, OIX

John "Mike" Davis, Veteran's Affairs

John Walsh, Sypris Electronics

Kevin Mangold, NIST

Lucy Lynch, ISOC

Marty Schleiff, The Boeing Company

Mary Ruddy, Identity Commons

Massimiliano Masi, Tiani "Spirit" GmbH

Mohammad Jafari, VA

Peter Alterman, SAFE-BioPharma Association

Ranier Hoerbe,

Rebecca Nielsen, Booz Allen Hamilton

Scott Fitch Lockeed Martin

Shaheen Abdul Jabbar, JPMorgan Chase Bank, N.A.

Shahrokh Shahidzadeh, Intel Corp

Suzanne Gonzales-Webb, VA

Tony Rutkowski,

Thomas Hardjono, M.I.T.

William Barnhill, Booz Allen Hamilton

1. References

[1] Survey of Methods of Trust Elevation Version 1.0. 24 September 2012. OASIS. <https://www.oasis-open.org/apps/org/workgroup/trust-el/download.php/46987/trust-el-survey-v1.0-wd01.doc>

[2] FFIEC Authentication Guidance. <http://www.ffiec.gov/pdf/authentication_guidance.pdf>

[3] ITU-T X.1254 (ISO/IEC JTC 1/SC 27 N10558)

[4] Wikipedia Man-in-the-middle attack.

[KLEINER01] Kleiner, E., Roscoe, A.W.: On the relationship between web services security and traditional  protocols. In: Mathematical Foundations of Programming Semantics, MFPS XXI (2005)

[GORDON01] Karthikeyan Bhargavan, Ricardo Corin, Cédric Fournet, Andrew D. Gordon: Secure sessions for web services. SWS 2004: 56-66

[GUTTERMAN01] Zvi Gutterman and Dahlia Malkhi, Hold Your Sessions: An Attack on Java Session-Id Generation

[LOUIS01] Jerry Louis, Detection of Session Hijacking, Master Thesis, University of Bedfordshire, 2011

1. Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Editor** | **Changes Made** |
| 0.1 | 2 Jan, 2013 | Mary Ruddy | Initial Draft |
| 0.2 | 20 Jan, 2013 | Mary Ruddy | Added revised, more detailed analysis template and revised analyses using new template. |
| 0.4 | 10 Feb, 2013 | Mary Ruddy | Incorporated Peter Alterman’s edits and adjusted formatting |
| 0.5 | 20-Feb, 2013 | Mary Ruddy | Incorporated Shaheen Abdul Jabbar’s comments. Adjusted formatting and wording and acknowledgments. |
| 0.6 | 7-Apr, 2013 | Mary Ruddy | Corrected two transpositions in committee member names. |