# Single Sign-On Webservice with Adaptable Multi Factor Authentication

Sandra Kübler, Christian Rinjes, Anton Wiens and Michael Massoth
Department of Computer Science
Hochschule Darmstadt – University of Applied Sciences
Darmstadt, Germany
e-mail: {sandra.kuebler | christian.rinjes | anton.wiens | michael.massoth}@h-da.de

***Abstract*—Cybercrime activities have led to a global cost of 445 billion USD in 2014. Potential and attractive targets of cybercriminals are identity and access management systems. These are especially used by enterprises to better organize their employees' credentials and privileges. Part of such a system can be a single sign-on service to reduce the number of different accounts/credentials of a user. To enhance security, multi factor authentication is slowly becoming more present in identity and access management systems and single sign-on services. In this paper, we will present a new approach to multi factor authentication in a web-based single sign-on service called SecureAID. This service is thought to be extensible and easy to implement for service providers, who are able to define their own (minimum) security levels. A security level defines which factors are required for a login to the service of a service provider. For a user, it is possible to define their own order in which factors are used, thus further improving usability. Additionally, a user is able to use an arbitrary number and type of factors, as long as the minimum security level defined by a service provider is met. This paper concludes with an evaluation of our approach.**

***Keywords-web-based single sign-on; multi factor authentication; digital identity; security levels.***

## I. INTRODUCTION

Through the help of identity and access management (IAM) systems, enterprises are trying to meet the demands of user account, privilege and password management, as well as single sign-on (SSO) services. With the growing digitalization, the need for secure cyber systems is bigger than ever, as cybercrime activities are growing as well. According to the internet security company *McAfee*, the global cost of cybercrime in 2014 has been estimated to be 445 billion USD [1]. Identity theft is one of the threats companies and their employees, as well as a person in private have to face. To enhance security of IAM systems, the use of multi factor or at least two factor authentication is growing. On the one hand, multi factor authentication combined with a SSO service in an IAM system must meet basic and additionally defined security aspects. On the other hand, these IAM systems must provide ease of use for a company and its employees, for users in general (private) and have to be practicable.

In this paper, we will present a new approach to combine an arbitrary number of authentication methods with distinct strengths to one identity. A service provider is given the possibility to support a large number of factors without having to implement them in their own platform. Only the minimum requirements concerning the required security level in total have to be defined by a service provider. A *factor* is an *identity* the user already possesses, e.g., a social login (Facebook, Twitter, etc.) of a third-party supplier or his own palm which can be scanned. A service provider using the system cannot derive the user's *distinct identities* from the single identity given by the system.

The paper is structured as follows. Subsequent to the introduction, a definition of terms is given. Section III shows related work, as in products and papers concerning multi factor authentication in an identity and access management system and single sign-on service. Following this, Section IV introduces attack vectors in general on web-based services. Our approach, SecureAID, is presented in Section V. How big of a threat the described attack vectors of Section IV are to our service is shown in Section VI. Section VII ends this paper with a conclusion and future work.

## II. DEFINITIONS

In this section, we will introduce definitions of terms helpful for understanding this paper.

**Security (own definition):** The term security can be defined in many different ways, depending on the actual context. As our goal is to provide a web-based single sign-on system with multi factor authentication, our definition is as follows: The system is viewed as *secure*, if no other individual (or robot, artificial intelligence, etc.) can impersonate the actual user, who wants to log into the service. It has to be pointed out that such a system consists of several possibly safety-critical components. This further bears the question whether one or more compromised components can lead to an insecure system as a whole.

**Identity and Access Management (IAM) System:** An IAM system is defined to be able to combine user account, privilege and password management, as well as single sign-on (see definition below) across all platforms and for all application types. It is a so-called *multiproduct approach*. [2]

**Web-based single sign-on (WebSSO) service**: A web-based SSO service is "used to move the authentication and authorization of users out of individual web applications, to a shared platform" [3]. Typically, when a user wants to sign in to a website or web application using a WebSSO service, the service first checks if the user is already authenticated. If this is not the case, the user is able to sign in using the required methods (e.g., username + password or a smartcard) at an authentication server [3].

**Multi factor authentication:** Using two or more independent factors for an authentication is seen as multi factor authentication. Generally, the more factors are used, the more secure it is. A hacker has more "obstacles" to overcome, meaning the hacker has to be able to gain access to all of the factors used to impersonate a person.

## III. RELATED WORK

There exist several products offering multi factor authentication in identity and access management systems and single sign-on services. PalmSecure truedentity from Fujitsu [4] is a product offering a mutual authentication of services and users, while the users' identities are kept in their possession. The authenticity of both parties is verified by a server. Several different factors, such as smartcards or biometric factors, can be used, but their palm vein scanner, PalmSecure, has to be used as one of the factors. IBM offers identity and access management systems and has several products in their product family. One of those is the IBM Security Access Management. IBM provides an integrated platform for web, mobile and cloud, offering "multiple strong authentication schemes", including one-time passwords (OTP) and SMS verification codes [5]. CA Technologies offers "CA Strong Authentication", which provides "multi-factor authentication for web applications, portals and mobile apps" [6]. Several factors are supported and security levels are introduced. These are dependent on the application the user wants to gain access to and it is not possible to choose custom factors within a category. As for the factors, they do not include external identity providers and all user data is saved on their own repositories (users and 2F credentials, as well as audit data).

Aloul et al. propose a method, which uses mobile phones for two factor authentication [7]. A mobile phone is used for generating an OTP being composed by using factors unique to the user and the mobile phone. This OTP is only valid for a user-defined period of time. They use a SMS-based mechanism for back-up and synchronization purposes.

Bhargav-Spantzel et al. use a two-factor biometric authentication in the first phase for a two-phase authentication mechanism for federated identity management systems [8]. Other authentication factors are combined with the biometric factor in the second phase. Their focus lies on the generation of a biometric key using vector-space modeling.

In [9], a modular framework for multi factor authentication and key exchange is proposed and a tag-based method is used.

In all solutions or approaches presented in related work, external identity providers were not included. In some cases, multi factor authentication is used as a synonym of two-factor authentication. Proprietary solutions often demand the use of at least one factor which has to be used, for instance PalmSecure in combination with truedentity. Our approach is designed to include external providers to be more extensible and to grant more flexibility. Furthermore, the possibility to use more than two independent factors is granted.

## IV. ATTACK VECTORS

Multiple possibilities to perform attacks on web-based systems do exist. In this work, we will focus on rudimentary attack vectors possible on such systems and will not delve into details.

We distinguish between attacks on our service/platform, the used interfaces (third-party provider) and the user. The purpose of such attacks is to gain access to user accounts and, therefore, compromise them. Depending on the system, a hacker can gain access to additional services. Concerning our approach, the hacker could have the intent to gain the digital identity of a certain user.

Several methods can be used to perform attacks. The act of *social engineering* consists of using a social disguise, a cultural ploy or a trick, mostly psychological towards a computer user in order to gain illegal access to, e.g., a computer or a network [10]. *Brute force* is the most basic method to perform an attack. An attacker uses "brute force" to gain access to a system, mostly by sequentially trying all possible variations of a password concerning the order of numbers, letters, etc. The method of performing *Man-in-the-Middle attacks* (user- and third-party supplier side) is defined as a situation, in which "an adversarial computer between two computers [is] pretending to one to be the other" [11]. Concerning attacks against a database (gaining access), be it one at a third-party supplier side or of the service provider, there exist several possibilities to do so. Having gained access to a database can potentially, e.g., enable an attacker to impersonate a customer/user, to alter existing data (changing administration controls), to add new content (giving the attacker full access to the system) or to simply delete or extract existing data.

## V. SECUREAID

Our approach, *SecureAID* (Secure N-Factor-Authentication and IDentity Management), is to combine several factors into one single identity of a user while securing the user's (data) privacy and identity. It is an independent web-based single sign-on service enabling multi factor authentication and is thought to be extensible and easy to implement for service providers. A web interface enables the user interaction, providing the possibility to register, login and configure an account. Service providers can integrate SecureAID as a service and define, which factors a user is required to have and use for the login to the provider's service. Users are able to register themselves to SecureAID using existing external factors, such as a login to a social network, etc., so that they can easily login via SecureAID, given the service provider they want to login to has integrated our service.

In this section, we will first provide an overview of SecureAID's architecture, followed by registration and authentication processes. Afterwards, possible factors for a multi factor authentication are organized in types of factors. These types are then further categorized into security levels, before a definition of the term "digital identity" is given. This section ends with an example of a login process, as well as advantages and disadvantages concerning the usability of SecureAID.

### *A. Architecture overview*

An overview of the architecture of SecureAID is shown in Fig. 1.

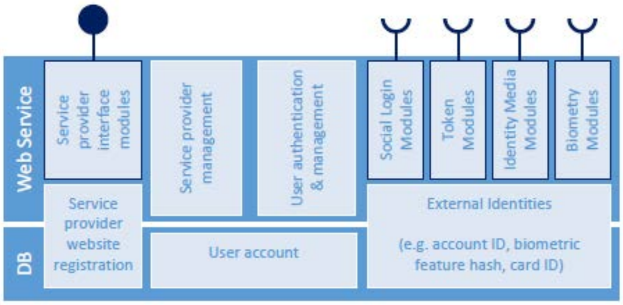
Web Service
DB
Service provider interface modules
Service provider management
User authentication & management
Social Login Modules
Token Modules
Identity Media Modules
Biometry Modules
Service provider website registration
External Identities
User account
(e.g. account ID, biometric feature hash, card ID)

Figure 1. Overview of the architecture of SecureAID.

SecureAID is a modular web-based single sign-on service. It provides modules for factors to retrieve or verify user identities from social login services, identity media, biometric devices and security tokens (external identities). Other modules exist to allow service provider websites to request authentication and retrieve a unique user identity. A database stores the identities and no credentials, as well as an ordered list of all identities used for authentication with service provider websites. A web interface allows users to create a new account and add, remove and rearrange the order of factors for the platform and service providers. It also allows for registration and configuration of new and existing service providers under the current user's account.

### *B. Registration and authentication processes*

In order to use SecureAID and before an authentication can take place, users have to register themselves to the service. In the following, we distinguish between user and service provider, the latter being a specialization of the former, as every user can potentially be a service provider.

#### *1) User registration process*

Users trying to sign in with an identity not present in the database are given the option to create a new account using the third-party identity provided. This new account can then immediately be used for authentication and be further extended with additional factors and ordered login lists for service providers.

#### *2) User authentication process*

For service provider authentication, a website redirects a user to the SecureAID platform to provide their credentials, which are then matched against the registered information. To modify their SecureAID account, a user simply opens the platform's website. The user then selects the first module he chose for the given service to start the authentication process and, in order, verifies and confirms all factors. After the requirements for the login list have been met, the user is then signed in to SecureAID or, if the required security level is met, authorizes the service provider website to retrieve their specific user ID. Furthermore, a user is able to define multiple paths for authentication. Each path is an order in which factors are used. With multiple possible paths for authentication, a user is provided with alternatives for the authentication process, for instance, if the user forgets one of the factors or is not able to use one. The alternative login processes have to meet the service provider's requirements concerning the security level.

#### *3) Service provider registration/authentication process*

Any user authenticated to SecureAID is able to register a new service provider website by specifying a display name, minimum security level and further authentication method specific credentials. In the case of OAuth2 (delegation protocol), a valid redirect URI must be specified and the user/service provider will be provided with a client ID and secret.

A sequence diagram of an authentication process of a user is shown in Fig. 2. In this example, a social login is used for authentication. Preliminary to the shown authentication process, the service provider a user wants to login to has registered its service to SecureAID. Upon starting the authentication process, the user is able to see SecureAIDs web interface and do further actions to authenticate.

Upon choosing, in this example, "login with social login X", a user gets redirected to the interface of the social login X (SLX), together with a client ID, redirect URI and the requested scopes belonging to the ID. The SLX first checks whether client ID and redirect URI are known. Upon a match, the user has to login to the social login. Given the credentials of the user are correct and the user wants to login for the first time with SLX using SecureAID, the user is asked if SecureAID is allowed to have access to the scopes of the ID. The confirmation is saved into a database to allow the access to the scopes for longer range, given the user allowed it. If the confirmation has been already given, the next step is for SLX to generate an authentication code which is, using the redirect URI, redirected to SecureAID. At last, SecureAID is given an ID token from SLX in order to request user data.

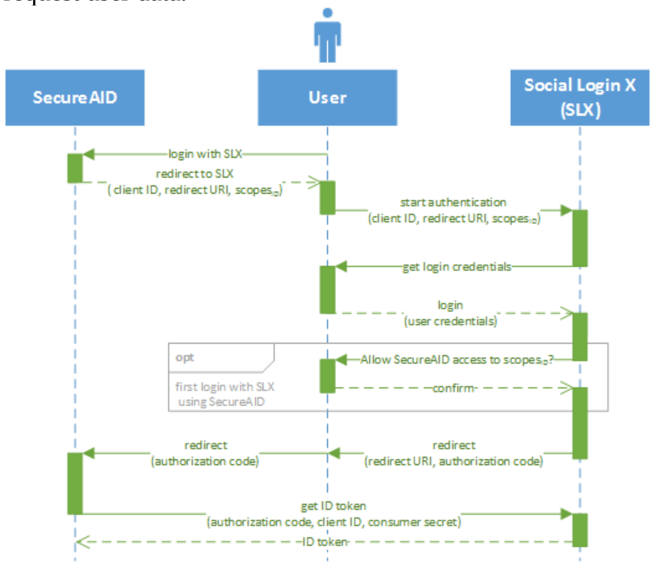
SecureAID
User
Social Login X (SLX)
login with SLX
redirect to SLX
( client ID, redirect URI, scopes$_{ID}$)
start authentication
(client ID, redirect URI, scopes$_{ID}$)
get login credentials
login
(user credentials)
opt
first login with SLX using SecureAID
Allow SecureAID access to scopes$_{ID}$?
confirm
redirect
(authorization code)
redirect
(redirect URI, authorization code)
get ID token
(authorization code, client ID, consumer secret)
ID token

Figure 2. The authentication process of a user using a social login X.

### *C. Factors for multi factor authentication*

There can be several factors for a multi factor authentication. In the following, the authors provide a list of factors grouped into different types.

#### *1) Social logins*

Already existing user accounts from external providers are defined as social logins. For example, Facebook, Google or Twitter accounts count as such. Usually, the login process is based on knowledge of the user (username + password). Therefore, it is a factor the user knows.

#### *2) Token*

A token is something users have in their possession and a factor from an external device. One example for tokens are time-based one-time passcode generators (software), such as Google Authenticator [12] or even a SMS with a passcode send to a mobile device. These passcodes can be used for a two-factor authentication by combining a conventional login of a user with a username and password with the generated passcode as a second factor. The FIDO (Fast IDentity Online) Alliance’s hosted Universal Second Factor (U2F) protocol can be used for a strong two-factor authentication, too, for instance with a FIDO U2F device [13] via USB with a button or NFC (hardware token).

#### *3) Identity media*

A user has an identity medium. This possession is often combined with knowledge of a PIN or password. Such a medium can be a smartcard or electronic identity (eID) card. Further, it is possible to have other features saved on an identity medium, such as a fingerprint or a vein scan.

#### *4) Biometry*

A biometric factor is something a user is. The most prominent biometric factors are fingerprints, iris scans or even vein scans of, e.g., a palm (see Fig. 3).

Figure 3. Example of a (palm) vein scanner: Fujitsu’s PalmSecure ID Match [14].

### *D. Security level*

As a means to further improve security of the service, *security levels* are introduced. A service provider has the possibility to define a minimum security level, which all of its users shall meet using a set of pre-defined *types of factors*. The combination of the chosen types of factors results in the minimum security level.

For instance, a security level is set to only allow authentications that include the two types of factors “identity medium” and “unforgeable biometric” (e.g., palm scan). A user can now only authenticate to this service provider when using at least these two types of factors.

Fig. 4 shows a pyramid of possible types of factors (see Section V.C). The higher its position in the pyramid, the more secure the type is. This ranking does not exclude the possibility to combine different types of factors, resulting in a higher security level. For instance, it is still possible for a service provider to demand a social login, as well as vein scan (unforgeable biometric factor) of a user for the login process.

*Social logins* are seen by the authors as the least secure of the shown types of factors. The credentials of a user are typically stored in databases by external providers of a social login. A database itself is a likely target by a hacker. Besides the database, there exist varying levels of personal security concerning the chosen passwords, which lead to them being regularly compromised. Additionally, social networking accounts are a common target for social engineering attacks. The next factor, *tokens*, are relatively easy to obtain, e.g., hardware tokens can be easily stolen, lost or even broken. They require access to another device or even physical access.

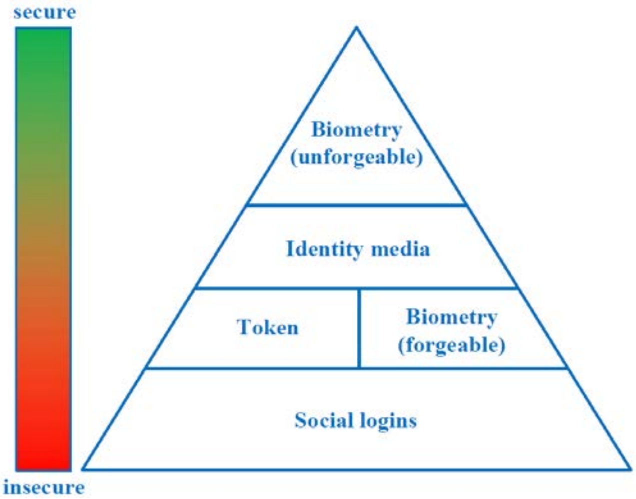

Figure 4. Pyramid of possible types of factors, in which each layer corresponds to a higher grade of security.

Biometric factors have been divided into *forgeable* and *unforgeable*. *Forgeable biometry*, for instance, a fingerprint can be copied by another person using a tape or a high resolution picture. Using an iris scan as a factor is viewed as forgeable as well, as a high resolution picture can suffice to tamper with an access gain system, too. On the other hand, a fingerprint or a picture of an iris is still needed beforehand in order to copy and use the factor, which is per se more difficult than obtaining credentials of a social login. *Identity media* are, in total, ranked higher than forgeable biometry and tokens. It is possible to save biometric factors, for instance, a fingerprint or a vein scan, which makes the identity medium itself less possible to be forged. In most cases, a PIN is required as well when utilizing an identity medium. *Unforgeable biometry* is at the top of the pyramid and, therefore, seen as the safest out of the mentioned factor types. For instance, using a vein scan as a biometric factor to gain access to a system is currently seen as unforgeable. A vein scan, be it of a palm or an iris, requires a flow of blood, meaning, a person wanting to gain access to a system has to be alive and therefore making it more difficult to forge.

### *E. Digital identity*

Each user has their own *digital identity*. After using the service to authenticate themselves with an arbitrary number of factors, a service provider is given a hash value – the digital identity. Besides the number and types of factors, a user is able to choose the order in which the factors are used. Due to the digital identity being a hash value, a service provider is not able to derive any factors and the order in which these factors were used by a user during the authentication process. Additionally, no two service providers receive the same digital identity for the same user.

### *F. Example of a login process*

Fig. 5 shows the login process with SecureAID using a social login and another factor from the second layer upwards of the pyramid (see Fig. 4) for the authentication.

A user from a service provider website wants to login using the service SecureAID (1). Using a social login and getting the approval of the service (2), the user can now use a second factor to authenticate. The second factor is used to verify that the user trying to login is truly the user. After the verification using the second factor is approved (3), SecureAID communicates the approval to the website of the service provider (4).

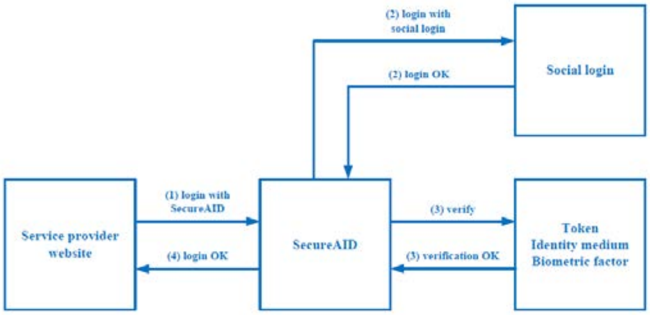

Figure 5. Example of a login process with SecureAID.

### *G. Usability*

In the following, possible advantages and disadvantages of SecureAID concerning usability are presented.

#### *1) Advantages*

Both a service provider and a user can profit from the customizable security level. For instance, if the required authentication security level of another service enabling the use of multi factor authentication is regarded as not high enough, a user can choose the use of SecureAID. Another point is the configuration through the web interface of the system. As already mentioned in Section V.A, it is possible to add, remove and rearrange the order of factors. This is done via drag and drop, which is more intuitively and comfortable for a user. The registration of new and existing service providers can also be done via web interface. A service provider only needs to indicate a redirect URI and is able to choose the types of factors for the minimum security level via checkboxes.

#### *2) Disadvantages*

More factors and diversity for authentication may lead to a decrease in usability. For instance, a service provider chose more than two types of factors – e.g., palm vein scan (biometric), eID (identity medium) and Facebook login (social login). Logging in with all three factors can lead to a user to be on edge, as the most common form of authentication is to type in a username and password or, especially concerning employees, by using an identity medium, e.g., a smartcard, which is less time consuming.

## VI. ATTACK VECTORS AGAINST SECUREAID

In Section IV, an overview of attack vectors against web-based systems has been given. In this section, we will elaborate theoretically how big a threat these attack vectors are to our approach and whether it is possible to impersonate a user or not.

### *A. Social engineering*

This attack vector – or rather method to perform an attack – strongly depends on the user. It is possible for a hacker to trick users into revealing all of their factors. A next step would be to get all login credentials. Now, it strongly depends on the used factors and the security level defined by the service. For instance, if only a social login and an OTP were required as factors, the possibility of a hacker being able to impersonate a user after using social engineering would be very high. If an unforgeable biometric factor like a palm vein scan were to be required, a hacker would not be able to impersonate the user, as this factor is bound to a “sign of life” of the user.

### *B. Brute force*

Using the method of brute force to perform an attack on SecureAID requires the knowledge of SecureAID as a service. If a hacker knows about a user’s profile in our service, the hacker could possibly acquire as much pieces of information as when he has access to the database. From here on, hackers can extend their “research” on other services, such as social logins, and eventually acquire the login credentials and, therefore, possibly impersonate the user.

### *C. Man-in-the-Middle attack*

A Man-in-the-Middle attack can be distinguished between user-side and third-party supplier side. On the user’s side, an attacker would be able to acquire the login credentials of a user of different services (e.g., social logins like Facebook or Twitter). Using *SecureAID* would not change the fact that acquiring such credentials is still possible for an attacker. The attacker would still have to gain the knowledge that a service like *SecureAID* exists and that there are additional pieces of information and data to be collected.

On the side of a third-party supplier, the digital identity of a user would be exposed for an attacker. The digital identity per se is not usable for a hacker, as it is only a hash value. However, the attacker could be able to monitor the user and possibly back trace the user's actions. This would lead to the attacker knowing about the service SecureAID and that the user is using this service. But the attacker would not know about the service provider the user wants to login to using SecureAID. Nevertheless, without any additional data, the attacker would not be able to fully impersonate the user, as the attacker does not have the *whole identity* and would still have to bypass the other factors used by the user.

### *D. Attack against database (gaining access)*

At this point, we are not going into detail about defense strategies in various aspects of a database holder, but we are assuming the fact that a hacker actually *has gained access* to a database. Potential targets of an attack could be a database of a third-party supplier or SecureAID's database.

Having access to a database of a third-party supplier is giving a hacker as much information as a Man-in-the-Middle attack. The attacker gets the third-party supplier's digital identity of a user and gains knowledge about SecureAID, but still has to do further "research".

Access to the database of SecureAID is providing a hacker with more pieces of information. All user's digital identities (user IDs) and corresponding third-party suppliers are stored in a database. With this data, a hacker is able to recreate a user's profile, but still has to get login credentials of a user and **is not able to impersonate** a user. Hacking the database of SecureAID does not come with a loss of a user's login credentials.

The success of a hacker is strongly dependent on the type of the factors used and, therefore, the defined security level of a service provider. For instance, if only social logins are used, a hacker could easily impersonate a user.

## VII. Conclusion and Future Work

In this paper, we presented a new approach for multi factor authentication using an arbitrary number of independent factors as a web-based SSO service SecureAID with a customizable level of security.

The ***unique characteristics and features*** of our approach are the following: Service providers can define their own minimum security level by choosing which types of factors have to be at least used. A user is able to freely choose the order and amount in which the factors are used. The choice of factors is only limited to the ones supported by SecureAID and the security level defined by a service provider using SecureAID. Another strength is, given a hacker has gained access to the digital identity (hash value) of a user, having the digital identity alone is not sufficient to impersonate a user. For instance, if one social login of a user is compromised, the hacker knows that the hacked user is using our system. This knowledge alone is not sufficient enough to impersonate the user, as the hacker still has to bypass all other factors/systems. Potential shortcomings can possibly lie in the usability and user-friendliness, as those can decrease with the number of used factors.

Concerning ***future work***, the list of possible attack vectors can be extended. Additionally, several and extensive tests concerning our approach's defense against these attack vectors have to be conducted.

## References

[1] McAfee, Net Losses: Estimating the Global Cost of Cybercrime, Economic impact of cybercrime II. Center for Strategic and International Studies. June 2014. Available from: http://www.mcafee.com/us/resources/reports/rp-economic-impact-cybercrime2.pdf, 2016.05.29.

[2] R. Witty, A. Allan, J. Enck, and R. Wagner, "Identity and Access Management Defined" Research Note, 04 November 2003, Note Number SPA-21-3430. Available from: http://www.bus.umich.edu/KresgePublic/Journals/Gartner/research/118200/118281/118281.pdf, 2016.05.11.

[3] Hitachi ID Systems, Inc., Web Single Sign-on. [Online] Available from: http://hitachi-id.com/resource/concepts/web-sso.html, 2016.05.30.

[4] Fujitsu, PalmSecure truedentity. White paper. [Online] Available from: http://sp.ts.fujitsu.com/dmsp/Publications/public/wp-palmsecure-truedentity-de.pdf, 2016.05.30.

[5] IBM, IBM Security Access Manager, data sheet. [Online] Available from: http://www.ibm.com/common/ssi/cgi-bin/ssialias?subtype=SP&infotype=PM&htmlfid=SED03160USEN&attachment=SED03160USEN.PDF, 2016.05.31.

[6] CA Technologies, CA Strong Authentication, data sheet [Online] Available from: http://www.ca.com/content/dam/ca/us/files/data-sheet/ca-strong-authentication.pdf, 2016.05.31.

[7] F. Aloul, S. Zahidi, and W. El-Hajj, "Multi Factor Authentication Using Mobile Phones" In International Journal of Mathematics and Computer Science, vol. 4, no. 2, pp. 65-80, 2009.

[8] A. Bhargav-Spantzel, A. C. Squicciarini, S. Modi, M. Young, E. Bertino, and S. J. Elliot, "Privacy Preserving Multi-Factor Authentication with Biometrics" In Journal of Computer Security, vol. 15, no. 5, pp. 529-560, 2007.

[9] N. Fleischhacker, M. Manulis, and A. Azodi, "A Modular Framework for Multi-Factor Authentication and Key Exchange" In Security Standardisation Research: First International Conference, Proceedings, Springer International Publishing, pp. 190-214, 2014.

[10] S. Abraham and I. Chengular-Smith, "An overview of social engineering malware: Trends, tactics, and implications" In Technology in Society 32, pp. 183-196, 2010.

[11] Yvo Desmedt, Man-in-the-Middle Attack. Reference work entry in Encyclopedia of Cryptography and Security. 2011, Springer US, pp. 759-759. ISBN 978-1-4419-5906-5, doi:10./1007/978 -1-4419-5906-5_324.

[12] Google Authenticator OpenSource, project side. [Online] Available from: https://github.com/google/google-authenticator, 2016.05.19.

[13] FIDO Alliance official homepage, Specifications Overview. [Online] Available from: https://fidoalliance.org/specifications/overview/, 2016.05.19.

[14] Fujitsu PalmSecure ID Match, product description. [Online] Available from: http://www.fujitsu.com/de/products/computing/peripheral/accessories/security/palmsecure-id-match/, 2016.06.08.