Web Authentication Using a Secure Personal Device

Sharareh Monfared and João Nuno Silva
INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
Email: {sharareh.monfared, joao.n.silva}@inesc-id.pt

Abstract—The authentication of users in legacy websites via mobile devices is still a challenging problem. Users are required to provide passwords, introducing several vulnerabilities: since strong passwords are hard to memorize, users often use weak passwords that are easy to break, and passwords can be intercepted by malware and stolen. Some users tend to use password managers in order to restore credentials more easily, hence using password managers has its own vulnerabilities: master password is needed to authenticate into password managers, and users tend to use vulnerable password. Besides that most passwords managers use clipboard as a mechanism for transferring credentials that can be compromised by malicious application.

In this work we propose a novel system, named BioALeg, to support secure biometric authentication on legacy websites. Our approach leverages the potential of a Secured Personal Device (SPD), a hardware addon for mobile phones that is being developed in the context of the PCAS European project (www.pcas-project.org). The device offers biometric authentication and secure storage services. BioALeg uses this infrastructure, and a companion web site plugin, to support biometric authentication in legacy web sites that currently use username/password authentication. Besides that BioALeg offers a secure copy/paste mechanisms for transferring credentials from SPD to legacy websites’ forms.

When accessing a legacy web site the user is presented a regular login form. The BioALeg middleware verifies that the server is also running the middleware and asks him a OTP. This OTP is generated on the server and transferred to the browser using the SPD. This transfer is only executed if the owner of the SPD is correctly biometric authenticated. To increase the security of stored password (for sites not implementing BioALeg) we also developed and Password manager using the secure storage feature of the SPD. To access a password the user must be biometrically authenticated. Besides that BioALeg system avoiding using clipboard by offering a secure copy/paste mechanisms, our approach takes advantage of SoftKeyboard architecture and paste the credentials through that.

These solutions increase the security of authentication on mobile web sites, requiring only installation of the middleware on the mobile-phone and minimal changes on the server infrastructure.

Index Terms—Web authentication, Legacy web services; Biometric authentication; Password manager; One-Time-Passwords;

I. INTRODUCTION

Based on the current researches, it is well known that mobile phones becomes more popular and the average user has around 25 online accounts and check these online accounts on their smart phones: users usually have various e-mail account, online-shopping, social networks, finance, banking or any other online system that needs credentials to verify the identity of a legitimate user[1].

These accounts still requires ways of authentication, one of the most commonly used system is password based authentication. Since memorizing passwords and typing complex words on mobile phones are not easy for most of the users, they prefer to pick easy passwords, that can be targeted by various attacks, such as dictionary attacks, or simply brute-force [2][3][4].

Additionally, users often share passwords over multiple accounts, which makes these attacks more profitable[1]. As a consequence, if someone credentials being phished, every associated account is compromised. Phishing attacks aim to steal private information, such as usernames, passwords, and credit card details, by way of impersonating a legitimate entity. So if user does not respect to security tips and use the same credentials for all owned accounts, he gets more profitable for hackers.

Since users often choose simple passwords with a low entropy allowing for efficient brute-force or dictionary attacks, different password policies try to address this issue by defining minimum password requirements. Examples are policies, such as "minimum-password-length", "different password for each account", "usage of special characters" and similar[5]. Increasing the password complexity could lead in usability issues since users tend to forget long and hard to guess passwords [6], they will store them on their smart-phone unencrypted[7]. Furthermore, those users, who do use a strong password, might think themselves safe and thus use it on multiple sites.

To avoid the burden of typing and memorizing password, in recent years, various password managers have been developed and are currently available for various mobile operating systems. The use of password managers makes easier for the users to select stronger passwords when performing web site authentication. Unfortunately, password managers also have some severe vulnerabilities. The credentials used for web authentication need to be stored in a local file or in the cloud. The master password used to access this file can be attacked by sociotechnical attacks such as shoulder surfing or it can be compromised if the user selects a weak password [8]. Also, the cryptographic primitives used to encrypt the file and/or the communication with the cloud may not be fully secure. Finally, several of these solutions use the clipboard for exchanging information among the web forms and password managers. This communication channel is open, so every application running simultaneously on an Android device can read the items stored in the clipboard at any time. This can be used by malicious applications to harvest passwords as they are passed through the clipboard [9][10].

Our work aims at solving the main vulnerabilities of password use in mobile devices: insecure storage of the passwords, insecure communication with the browsers and
mobile-devices, and leakage of the used passwords by avoiding copy/paste mechanism (clipboard), adding biometric authentication and encrypting communication to password managers.

We address these problems with the help of a Secured Personal Device (SPD), a hardware add-on for mobile phones that is being developed in the context of the PCAS European project[11]. The device offers biometric authentication and secure storage services, that we use to support biometric authentication into legacy web sites and as a secure storage for credentials into new-designed password manager. To support user authentication in this manner, web site providers just need to implement an OTP generator and a simple web service on the server the generation and transfer of the OTP to the SPD/web browser.

BioALeg is composed of a set of components running on the web site and on the mobile phone which consists of two different parts:

- Combination of SPD with a new-designed password manager that we called this part of the BioALegPM. In BioALegPM, the transfer of the credentials from SPD to password manager and vice versa requires the user to authenticate using the biometric sensors of the SPD. On mobile phone, the BioALeg is responsible for the inserting , deleting and restoring credentials in SPD. Further more BioALeg is capable of restoring credentials to the BioALegPM by getting the request from BioALegPM, receives it and fills the authentication form automatically without usage of clipboard.

All these communication phases are passed through SPD by using biometric authentication, so by fulfilling all these steps, legitimate user will be identified accurately. BioALeg allows the use of combination of BioALegSoft-Keyboard and BioALegAccessibilityService to replace with clipboard copy/paste mechanism. By usage of this system, compromising of credentials and being hacked got decreased dramatically.

- Combination of SPD with One Time Passwords (OTP) that we called this part of the BioALegOTP. In BioALegOTP, the transfer of the secret from the server to the client and back to the server requires the user to authenticate using the biometric sensors of the SPD. On the web site, the BioALeg is responsible for the generation of OTP and for transferring it to the SPD. On the smartphone, the BioALeg first request the generation of the OTP on the web site, receives it and transfers it to the authentication form on the browser. Since all communication steps are authorized in the SPD by means of biometric authentication, the completion of these steps guarantees the validation of the user identity. BioALeg allows the use of strong OTPs by supporting different algorithms implemented only in the server. This makes this system more resilient against risks and hacks.

The rest of the paper is structured as follows. Section II presents the related work, followed by a description of the proposed architecture in Section III and Section IV. An evaluation of BioALeg is presented in Section VI, and Section VII concludes the paper.

II. RELATED WORK

In this section, we briefly survey related work on traditional authentication, mobile device authentication vulnerabilities, and multi-factor authentication. We also describe the main features of the SPD device.

A. Usernames and Passwords

Nowadays, most online services authenticate users using the traditional method based on a username and a password: the username identifies which account the client wishes to access, while the password is used to validate the identity of the client[12]. On the server side, passwords are stored using a cryptographic hash function and thus, unless other vulnerabilities are introduced in the authentication work-flow, an attacker is forced to guess the password to attack the system.

Unfortunately, guessing a password may be relatively easy. In fact, users often choose simple passwords, with low entropy, allowing for efficient brute-force or dictionary attacks[13][14]. Other attacks, such as phishing[15] also allow to obtain passwords from users. Furthermore, since users tend to share the same password for multiple services, once a password gets stolen, every associated account is compromised. As a result, obtaining passwords by illegitimate means has become a lucrative business.

B. Vulnerabilities of Mobile Device Authentication

When users rely on mobile devices to authenticate, new vulnerabilities are introduced. This happens because mobile devices inherit most vulnerabilities of desktop computers, amplified by the higher risk of loss or theft, and its usage patterns add new vulnerabilities[8], as listed below:

- Insecure data storage: Users may rely on password managers to save passwords that are hard to memorize. However, these managers may store passwords in the memory of the mobile device where it may be read by other applications or use insecure communication when backing up the database to the cloud[16]. This causes the exposure of sensitive information.

- Infected Device: Users tend to install applications, such as games, from untrusted sources on their mobile devices. Also, some devices quickly become obsolete and no longer support required software updates. This makes easier for mobile devices to become infected by malware.

- Shoulder Surfing attacks: Mobile devices tend to be used in public areas, where it is easy for an attacker to observe or capture the user’s movements to get the password[17].

- Dictionary Attacks: Because long passwords are harder to type on mobile devices, users tend to pick simpler passwords that are more vulnerable to dictionary attacks[17].

- Unauthorised use: Mobile devices can easily get lost or be stolen. Once the attacker has physical access to
the device he/she can generally get access to all the information stored on the device [18].

- **Man in the middle attack**: Mobile devices typically access the internet using untrusted networks (for instance, wifi networks), making easier to setup man in the middle attacks: a form of message tampering in which an attacker intercepts the very first message in an exchange of encryption keys to establish a secure channel [19].

- **Cached Passwords**: Since passwords are inconvenient to type in a mobile device, there is a strong motivation to cache them in the browser password manager. This creates a window of opportunity for an attacker to retrieve passwords from this insecure password storage.

### C. Multi-Factor Authentication

Multi-Factor Authentication is an authentication strategy that requires the users to provide multiple pieces of evidence. Typically, these evidences are the combination of a password (something the user knows) with an additional token such as something that the user owns or something that captures who the user is (such as a biometric verification). Multi-Factor Authentication aims at strengthening the security of password-based login authentication by requiring an attacker to compromise multiple mechanisms before successfully breaking into the target [20]. The implementation of multi-factor authentication on mobile devices is simplified by the fact that most devices support multiple interfaces and multiple communication channels/protocols (infra-red, Bluetooth, 3G, and WLAN connectivity).

1) **One-Time Passwords**: As the name implies, One-Time Passwords (OTP) are passwords that can be used only once to authenticate a user with a service. Thus, even if the password is stolen, it cannot be used again [21].

One-time passwords can be generated offline and stored for later use, but such approach shares many of the vulnerabilities enumerated before: if the storage is compromised, the attacker will be able to use all the passwords that have not been used before. Another approach relies on the generation of OTPs only when needed, such that it does not need to be stored persistently.

OTPs can be generated by the server and sent to the user by an secondary channel, such as via a SMS service. In this case, the user needs to access web service, use a traditional password to request the generation of the OTP and own a mobile phone to receive the SMS.

Another current OTP implementation relies on the use of an smartphone application to generate the one-time password on demand. In this case, the users needs to know a password to activate the generation an OTP on the device. Google Authenticator [22] is a software based solution that generates one-time passwords on mobile devices, that implements this solution.

2) **Time-based One-time Password (TOTP)**: A Time-based One-time Password Algorithm (TOTP) is an algorithm that computes a one-time password from a shared secret key and the current time [23]. The TOTP combines a secret key with the current time-stamp using a cryptographic hash function to generate a one-time password. TOTPs make easier to enforce a limited time validity on the generated token.

3) **Mobile OTP implementations**: Several system that make use of OTPs or TOTPs have been developed in the past. We briefly present some of the most relevant approaches in the next paragraphs.

   a) **SMS OTP**: One way of deploying an OTP-based authentication scheme, using a channel that is available to wide range of users, is to use the Short-Message Service (SMS) provided by mobile telecom operators as a secondary channel. Typically, the server generates the one-time code using the OTP algorithm, and sends this to the user’s smartphone using the SMS. Authentication happens when the server recognizes that the user typed in the correct code during login.

   One of the benefits of SMS OTP is that the user does not have to carry an additional hardware device, only her smartphone that it is assumed to carry anyway. Systems that use this approach generally produce four to six digit OTPs that are easy for users to copy and type for authentication.

   A limitation of the approach is that, if the mobile phone is stolen, the attacker will able to receive the SMS that was intended to the user [24]. Also, a powerful attacker may intercept the communications with the smartphone and capture the SMS content in transit. Furthermore, since many delays may occur in the process of delivering a SMS (even up to more than a minute [16]), the validity of the generated token needs to be large to avoid it being invalidated before the user uses it; unfortunately this also widens the window for an attacker to steal and use the token. The use of smartphones allows the installation of mali-ware that intercepts the SMSs.

   b) **Google Authenticator**: Google Authenticator [22] uses an offline variant of TOTP, where the user’s device generates the one-time codes and the server verifies it (this works because both can use the same algorithm to create the same OTP). Since secret keys have been previously shared between the client and the server, the server is able to validate the TOTP being presented for authentication. Because both participants have generated the same one-time code on the given time window, the server can verify the code sent from the client.

   However, if the mobile device is stolen or compromised (virus), the user credentials can be compromised, allowing the request of OTPs by the attacker. If the smartphone storage is compromised the secret shared key between client and server can be discovered and used by the attacker to generate TOTPs.

   c) **YubiKey OTP**: A YubiKey [25][26] is a small hardware device developed by Yubico that supports two-factor authentication. If a user registers with a service or site that supports this kind of authentication, each time user logs in, the service will request proof that user has his/her YubiKey in addition to the static password. The YubiKey is capable of generating standard OAUTH (Open Authorization Protocol) tokens as well as its own Yubikey-OTP, and implements challenge-response operations. YubiKey is vulnerable to theft: if someone steals the device, data can be compromised, since
there is no mechanisms to prevent unauthorized users from using it.

d) 1password: 1password is one of the password managers that can be used for android devices as well as windows and IOS. 1Password creates strong, unique passwords for every site, remembers them all for users, and logs in user with a single tap. It is one of the ways to stay secure and the fastest way to use the web[27]. Following features are some of the abilities of this application:

- 1Password remembers all users’ passwords so user does not have to.
- Easily and quickly recall time-based, one-time passwords when users need them.
- Sync items with 1Password for PC, Mac, iPhone, or iPad
- Take all users’ website logins, credit card info, secure wallet items, and more on the go.
- Log in to websites with a single tap, using the built-in browser or 1Password Keyboard.
- Protect users’ vault with a single Master Password.
- Secure with authenticated AES-256 encryption.
- Auto-lock keeps user’s data protected even if his/her device is lost or stolen.

D. SPD - PCAS

PCAS [11] is an European project whose goal is to provide a solution to allow users to store confidential information and to perform web site authentication in a secure manner. Central to the PCAS architecture is a special hardware device, the Secured Personal Device (SPD), that offers secure storage, secure communication, and biometric user authentication. The SPD operates as a sleeve that can be attached to a mobile phone and that is able to provide storage to trusted applications running on the phone and allows establishing end-to-end secure connections with external web sites using the communication capabilities of the attached mobile phone.

Figure 1 provides an overview of the PCAS architecture. The architecture defined by the PCAS project includes a number of components, as described below:

- The Secure Personal Device (SPD) is a smartphone add-on that stores information securely, authenticates the user, and mediates a safe communication with service provider.
- Secured Trusted Applications (STA) are applications that executes on the smartphone and that is authorized to communicate with the SPD to read and store data. The communication between the SPD and a STA is mediated by a dedicated library and service.
- The Secure Trusted Gateways (STG) is the PCAS component that implements the PCAS PKI and manages all identities for SPD, service providers, and users. The key management done in the context of the STG guarantees the identity of entities communicating, namely: i) between the SPD and the service provider and, ii) the authenticated user and the service provider.
- A Proxy for Secured Trusted Applications (PSA) is the PCAS component that allows the integration with existing web sites. This proxy translates the authenticated calls generated in the SPD into application level web site calls. This proxy also allows matching SPD identities (validated by the cryptographic algorithms used) into user identities (in the service provider domain).

Fig. 1. Architecture for secure web site access using SPD

III. BioALEGOTP

In this paper, we propose a method to use the SPD for authenticating users in mobile devices and to use to such authentication to login on legacy web sites. Since the biometric authentication is local to the SPD, we resort to a token produced in the server that is transferred to the browser on the mobile device via the SPD to validate the user identity.

In order to authenticate to a legacy web site, the user will be asked to fill a regular login form. The user will not use a fixed password, but will use an OTP. This OTP will be transferred to the smartphone through the SPD, an action that only occurs after the user biometric authentication is performed successfully. After the biometric authentication is performed by the SPD, the OTP will be delivered to the web browser and, on submission, transferred back to the server. Since the PCAS infrastructure provides identity management to validate SPDs, web site, and users, if the server receives back the produced OTP, it will know that the owner of the SPD was correctly authenticated.

Using the SPD as a secure communication channel, the OTP can be generated by the server using strong algorithms, securely transferred to a valid SPD, and only accessed by the correct user.

This proposed solution resemble TOTPs, but differs in some aspects. TOTPs require the execution of compatible code in the server and mobile device. Since the SPD can not run custom code (developed by external programmers) it is impossible for a TOTP to be generated on the device. The generation and secure transfer of the OTP will overcome this limitation, with added security. Furthermore the algorithm to generate the OTP can change without any modifications on the smartphone code.

A. Architecture

When a user attempts to access remote web sites or services, BioALEGOTP replaces password typing with biometric authentication, relying on the following components:
• The SPD device, which can only be accessed by her owner.
• A Secure Trusted Gateway (STG) controls the registered SPDs, mediates the establishment of communication between users and the service providers, and validates user’s identities.
• Applications installed in the mobile phone, called Secured Trusted Applications (STAs), that provide interfaces for secure data and web services access. These applications are implemented by the service providers and will able to securely communicate with the SPD (for data access) and the service provider (for invocation of web site). Each Service provider should implement and sign its own set of STA.
• Security algorithms and protocols protect both communication and data. The public key infrastructure and protocols guarantee the identity of all the entities (SPD, user, services) in the system.

BioALegOTP extends the SPD/PCAS architecture with the following components:
• A web browser plugin that allows the secure use of this new authentication mechanism;
• Secured trusted applications are installed on the client’s mobile phone and request the generation of a new OTP; these requests are forwarded to the service provider through the SPD;
• A server-side software, named the Service provider OTP Generator, required to generate OTPs and transfer them to the SPD.

Figure 2 shows how the user authenticates on a web site. During this procedure the user is not required to type any password; such procedure is replaced by the biometric authentication on the SPD.

The browser plugin is a generic component, that can be used by multiple web site providers. This plugin will automatically fill the forms with the username and the OTP provided by the server, if correctly authorized by the biometric authentication procedure implemented by the SPD. Each web site provider should implement the OTP generator, install it on its server, and configure the client side application to run on the mobile device as a Secured Trusted Application (i.e., an application that is able to interact with the SPD). These two components (both the web site and the STA) should be previously registered into the PCAS/SPD infrastructure. This is required because the interaction between these two components will be mediated by a Secure Trusted Gateway that will check the identity of both peers.

Next we describe in detail the steps involved in the authentication process when using BioALegOTP:

1) The user accesses a web site through the mobile browser;
2) The mobile browser connects to the web site and presents the login form;
3) The user requests the BioALegOTP to start the biometric authentication;
4) The BioALegOTP initiates a web site call to the OTP generator. BioALegOTP does not contact the web site directly, but uses instead the SPD as a proxy;
5) Before forwarding the request to the web site, the SPD initiates the user biometric authentication and the user performs the requested authentication (e.g hand recognition);
6) The SPD validates the data read from the sensors with the internally stored user biometric template;
7) If the user trying to authenticate is the owner of the SPD, the OTP request is forwarded to the web site;
8) The OTP generator creates a new OTP, stores it on a persistent storage, and returns the value to BioALegOTP;
9) BioALegOTP fills the web form. The SPD sends an authentication acknowledge to the service-provider;
10) The user submits the authentication form (containing the received OTP), and the web site compares the OTP sent by the user with the one generated previously.

Although some of the steps (3 and 9) may be performed by means of insecure communication channels, the more sensitive information is guaranteed by the SPD/PCAS system to be encrypted and valid. The PCAS key infrastructure and the cryptographic algorithms guarantee that if the web site receives a call from a certain SPD (step 7) the owner of the SPD was correctly authenticated. If, on step 6, the SPD fails to validate the user identity, the request will not be forwarded to the server and the OTP will not be generated.

B. Implementation

The proposed solution was implemented and deployed on mobile phones using the Android operating system. On the smartphone the BioALegOTP is composed of:
• A browser add-in to generate strong TOTPs;
• BioALegSoftKeyboard to detect the second factor authentication form;
• A browser plugin to allow the automatic filling of the authentication forms;
• Accessibility service for detection of different forms parts need to be filled;
• the PCAS library, common to all application that use the SPD;
• A browser plugin to allow the automatic filling of the authentication forms;
• One or more STAs, for communication between the smartphone and the service provider.

Android allows the implementation and use of different software keyboards. The browser plugin is implemented by means of this Android extension. This software keyboard allows the retrieval of information from the forms being filled, and allows the filling of text in a secure manner, without resorting to the clipboard.

Since the SPD/PCAS environment requires the registration (and assignment to a specific service provider) of the STAs, each service provider willing to use BioALegOTP should implement its own STA. The users should install on the smartphone these Service provider specific STAs. Depending on the web site being accessed, this application receives from the browser plugin a request to authenticate user, and contacts the web site.

Legacy services must be extended with an TOTP generator that were developed in this project, that will be accessible by the SPD/smartphone trough a website.

Whenever a user wants to authenticate through these legacy web services, BioALegSoftKeyboard can detect which forms are requested to be filled. Based on that BioALeg system can request BioALegPM or BioALegOTP for automatically filling first authentication factor or second authentication factor forms.

This automatic filling is a plug in that is compatible with most of the browsers that could support accessibility services. If BioALegOTP system has been requested, TOTP will be generated and stored on the database of the website. The result of this request will be retrieved as a xml file and being transferred to the SPD, if the owner of the phone and SPD is correctly authenticated.

The content of the xml file will be transferred to the SPD and will be filled the form by secure copy/paste mechanism that took advantage of using BioALegSoftKeyboard.

Since the SPD/PCAS environment requires the registration (and assignment to a specific service provider) of the STAs, each service provider willing to use BioALegOTP should implement its own STA. The users should install these Service provider specific STAs on the smartphone.

Depending on the web site being accessed, this application receives from the browser plugin a request to authenticate user, and contacts the website.

C. User Authentication

Before the first use of the SPD, it is necessary to register that SPD and assign it to a specific user. This is done after the purchase of the device and in front of an administrative officer (i.e., authorized personal). The whole system guarantees that all authenticated accesses performed from that SPD are done by its registered owner.

Before being allowed to deploy services on the SPD/PCAS infrastructure, service providers also have to register the identity of the web site and STA applications. The public keys stored in the gateway (STG presented in Figure 1) guarantee the authenticity and identity of each application and service provider.

The SPD/PCAS infrastructure guarantees that all accesses from a certain SPD were performed after the owner of that device was correctly authentication using biometric.

IV. BioALegPM

BioALegPM is a password managers that can be used for android devices. BioALegPM stores credentials on SPD, communicate securely with web services and use a safe copy/paste mechanism for authentication instead of using clipboard.

Most of the password managers suffer from different vulnerabilities include. Using master password can be compromised by easy to guess passwords, insecure data storage that is caused by using mobile devices as a storage, unencrypted credentials and insecure communication which is caused by using clipboard are some of the main issues exist in password managers.

Before these transactions has been started the SPD needs to be authenticated by biometric factors. By implementing this steps we can make sure that the user is the real legitimated user, so if the SPD has been stolen the attacker could not compromised data since he does not have the biometrics of the legitimated user to authenticate through SPD.

In order to authenticate to a legacy web site, the user will be asked to fill a regular login form. The user will not fill the form by his own anymore, meaning that BioALegPM will help the user through that and fills the forms authentically. BioALegPM will transfer credentials to legacy websites through the SPD, an action that only occurs after the user biometric authentication is performed successfully. After the biometric authentication is performed by the SPD, the credentials will be delivered to the web browser through SPD, BioALegPM and BioALegSoftKeyboard. Since the biometric authentication is local to the SPD, we resort to a token produced in the server that is transferred to the browser on the mobile device via the SPD to validate the user identity. Since this transactions implemented to go through secure trusted application (STA) and secure trusted gateway (STG) to restore credentials from SPD to BioALegPM and filling authentication forms in legacy websites, the security has been guaranteed.

Using the SPD as a secure communication channel, the credentials stored on SPD can be transferred to the server using secure copy/paste mechanism, securely transferred through BioALegPM to legacy websites, and only accessed by the correct user.

By all these new techniques the recommended solutions will turn to the best password manager among all the others.

This system is secure and consists of four parts:
- BioALegPM software installed on the client’s mobile phone.
- SPD connected to the mobile-phone.

A. Architecture

When a user attempts to fill the forms in remote web sites or services, BioALegPM replaces password typing with biometric authentication, relying on the following components:

- The SPD device, which can only be accessed by his/her owner.
- A Secure Trusted Gateway (STG) controls the registered SPDs, mediates the establishing of communication between users and the service providers, and validates user’s identities.
- Applications installed in the mobile phone, called Secured Trusted Applications (STAs), that provide interfaces for secure data and web services access. These applications are implemented by the service providers and will able to securely communicate with the SPD (for data access) and the service provider (for invocation of website). Each Service provider should implement and sign its own set of STA.

BioALegPM extends the SPD/PCAS architecture with the following components:

- A web browser plugin that allows the secure use of this new authentication mechanism;
- Secured trusted applications are installed on the client’s mobile phone; these requests are forwarded to the service provider through the SPD;

The BioALegAccessibilityService and BioALegSoftKeyboard are the generic components, that can be used by BioALegPM. These components will automatically fill the forms with the username and password that stored on the SPD, if correctly authorized by the biometric authentication procedure implemented by the SPD. Each legacy web site credentials should be stored on SPD at the first steps. BioALegPM implement the credential storing step besides further steps for restoring and filling the forms automatically. BioALegPM will be installed on the mobile phones, and configure the client side application to run on the mobile device as a Secured Trusted Application (i.e., an application that is able to interact with the SPD). These two components (both the web site and the STA) should be previously registered into the PCAS/SPD infrastructure. This is required because the interaction between these two components will be mediated by a Secure Trusted Gateway that will check the identity of both peers.

Next we describe in detail the steps involved in the authentication process when using BioALegPM:

1) The user accesses a web site through the mobile browser;
2) The mobile browser connects to the web site and presents the login form;
3) The user requests the BioALegPM to start the biometric authentication;
4) The BioALegPM initiates a web site call to restore credentials. BioALegPM does not contact the web site directly, but uses instead the SPD as a proxy;

5) Before forwarding the request to the web site, the SPD initiates the user biometric authentication and the user performs the requested authentication (e.g. hand recognition);
6) The SPD validates the data read from the sensors with the internally stored user biometric template;
7) If the user trying to authenticate is the owner of the SPD, the restoring credentials steps is forwarded to the SPD and SPD send credentials to the BioALegPM via BioALegSoftKeyboard;
8) Credentials will be filled to the forms by BioALegAccessibilityService in the mobile browser and using BioALegSoftKeyboard;
9) The user submits the authentication form and the web site compares credentials sent by the user with the one stored on its side;

Although some of the steps (3 and 8) may be performed by means of insecure communication channels, the more sensitive information is guaranteed by the SPD/PCAS system to be encrypted and valid. The PCAS key infrastructure and the cryptographic algorithms guarantee that if the web site receives a call from a certain SPD (step 7) the owner of the SPD was correctly authenticated. If, on step 6, the SPD fails to validate the user identity, the request will not be forwarded to the server and the OTP will not be generated.

B. Implementation

The proposed solution was implemented and deployed on mobile phones using the Android operating system. On the smartphone the BioALegPM is composed of:

- the PCAS library, common to all application that use the SPD;
- a browser plugin to allow the automatic filling of the authentication forms;
• one or more STAs, for communication between the smartphone and the service provider.

Since the SPD/PCAS environment requires the registration (and assignment to a specific service provider) of the STAs, each service provider willing to use BioALegPM should implement its own STA. The users should install these Service provider specific STAs on the smartphone.

Depending on the web site being accessed, this application receives from the browser plugin a request to authenticate user, and contacts the web site.

V. Secure Password Filling

Secure Password Filling is a method of entering credentials at login prompts without the need to type them. This prevents shoulder surfing as well and also thwarts key loggers. This method does not use clipboard since using clipboard for copying a password from the application can compromise credentials. The best practice of this work is to keep passwords encrypted during transaction of credentials between SPD, smart phone and web service.

Since using clipboard can cause malware compromise credentials and typing passwords can lead to shoulder surfing attack and decrease usability, this work also implemented a nice automated form filling feature. Since this system can prevent the actual typing of a password and avoiding use of clipboard, any potential keyloggers that could be maliciously installed on the computer are rendered useless.

We propose a new SoftKeyboard with two more buttons for pasting username and password/OTP. In this method we use the SPD for authenticating users in mobile devices and to use to such authentication to transfer credentials from SPD to BioALegPM and then to legacy websites. Since this transactions implemented to go trough secure trusted application (STA) and without using clipboard (clipboard is accessible by all other applications on mobile-phones it can be a good target for malicious application [10] which steals the passwords and will likely stay in the clipboard for a long period of time)secure, the secure transaction has been guaranteed.

At the first step for authentication a user into a legacy web site, the user will be asked to fill a regular login form. The BioALegSoftKeyboard is a softkeyboad that can be used for android devices. BioALegSoftKeyboard restore credentials from SPD, communicate securely with BioALegPM and use a safe copy/paste mechanism for authentication instead of using clipboard. By all these new techniques the recommended solutions will turn to the best secure transaction among all the other password managers.

This system is secure and consists of four parts:

• BioALegSoftKeyboard installed on the client’s mobile phone by the first installation of BioALeg system.
• BioALegSoftKeyboard enabled on the client’s mobile phone.
• BioALegAccessibilityService enabled on the client’s mobile phone.
• SPD connected to the mobile phone.

A. Architecture

When a user attempts to login into a legacy web site, BioALegSoftKeyboard replaces the normal keyboard and pops up, relying on the following components:

• The SPD device, which can only be accessed by his/her owner.
• A Secure Trusted Gateway (STG) controls the registered SPDs, mediates the establishing of communication between users and the service providers, and validates user’s identities.
• Applications installed in the mobile phone, called Secured Trusted Applications (STAs), that provide interfaces for secure data and web services access. These applications are implemented by the service providers and will able to securely communicate with the SPD (for data access) and the service provider (for invocation of website). Each Service provider should implement and sign its own set of STA.
• BioALegAccessibilityService enabled after installing BioALeg system, which can communicate to the user on the application’s behalf, by detecting text fields types when a user is hovering on an important area of the screen.

BioALegSoftKeyboard enables the SPD/PCAS architecture with the following components:

• A web browser plugin that allows the secure use of this new authentication mechanism;
• Secured trusted applications are installed on the client’s mobile phone; these requests are forwarded to the service provider through the SPD;

The BioALegAccessibilityService and BioALegSoftKeyboard are the generic components in our BioALeg system. These components will automatically fill the forms with the username and password that stored on the SPD, if correctly authorized by the biometric authentication procedure implemented by the SPD. BioALegPM will be installed on the mobile phones, and configure the client side application to
run on the mobile device as a Secured Trusted Application (i.e., an application that is able to interact with the SPD). These two components (both the web site and the STA) should be previously registered into the PCAS/SPD infrastructure. This is required because the interaction between these two components will be mediated by a Secure Trusted Gateway that will check the identity of both peers.

B. Implementation

The proposed solution was implemented and deployed on mobile phones using the Android operating system. On the smartphone the BioALegSoftKeyboard is composed of:

- The PCAS library, common to all application that use the SPD;
- Accessibility service for detection of different forms parts need to be filled;
- BioALegSoftKeyboard to allow the automatic filling of the authentication forms;
- One or more STAs, for communication between the smartphone and the service provider.

Android allows the implementation and use of different soft-keyboards. The BioALegAccessibilityService is implemented by means of receiving callbacks from Android background. The BioALegSoftKeyboard allows the retrieval of information from SPD, and allows the filling of text in a secure manner, without resorting to the clipboard.

Since the SPD/PCAS environment requires the registration (and assignment to a specific service provider) of the STAs, each service provider willing to use BioALegSoftKeyboard should implement its own STA. The users should install these Service provider specific STAs on the smartphone.

VI. EVALUATION

BioALeg allows the seamless integration of biometric authentication into legacy web sites. The avoidance of requiring users to memorize or type password reduces the risks and attacks, and increased the usability of the available authentication methods available on mobile devices.

A. Security

Table I summarizes the differences among the proposed solution and other solutions based on the various risks and attacks.

As it is depicted in Table I, the classic authentication method (based on username/password) is the most unsafe method, as it can be compromised by several attacks (most of them related to user selecting weak passwords). Since most of the authentication methods resort to tokens, they become vulnerable to attacks (infected device or shoulder surfing). The unauthorized use of the device leads to other attacks: the user can request a SMS based OTP that will be received in the same smartphone.

BioALeg is resilient of the various attacks due to its characteristics:

- use of SPD - avoiding the use of insecure storage, or cached passwords, voiding of compromise of credentials even on infected devices and avoiding insecure communication channel between OTPGenerator and SPD.
- use of BioALegPM - voiding of use of dictionaries to find the passwords, voiding of memorize complex passwords, avoiding the share of passwords over multiple accounts and avoiding clipboard for copy/paste mechanism.
- use of OTP - avoiding the use of insecure storage, or cached passwords, voiding the use of dictionaries to find the passwords;
- use of strong OTP generators - even on infected devices the discovery of the OTP generator function will be complex;
- biometric user authentication - shoulder-surfing attack will be impossible;
- no exchanged secrets - since no secrets are exchanged and shared between the smartphone and the service provider, the MiM attack reders no relevant information.

B. Usability

Table II compares how the different authentication mechanism handle several usability concerns.

C. Compatibility

Our proposed solution is compatible with all recent android versions (from Kitkat onward), and browsers (Chrome, Fire-
VII. CONCLUSION

With the increasing use of mobile devices, the threat of compromising credentials grows: weakly-chosen static passwords, reuse of password in multiple services, password manager implementing weak security on the storage, and communication of the passwords greatly increase risks in mobile devices.

In this paper, we propose the integration of a novel smartphone add-on, the SPD, that supports user biometric authentication to implement a more secure mobile user authentication.

Thanks to the SPD/PCAS infrastructure, it is possible to integrate biometric authentication into legacy web sites, with increase of mobile authentication security and usability.

On the android phone, BioALegPM implements a copy/paste mechanism by using BioALegAccessibilityServices and BioALegSoftKeyboard that communicates with SPD and websites. The BioALegPM fills the forms automatically without using clipboard. The access to the SPD and automatic form filling will only completes after the successful user biometric authentication on the SPD.

On the other hand, service providers only have to implement a server plugin that generates OTPs and delivers them to the SPD using a web site. The OTP generation algorithm only runs on the server, allowing the implementation of custom, stronger algorithms. On the android phone, the BioALegOTP is composed of a plugin to fill web forms and a application to interact with the service provider. The access to the service will trigger a OTP generation that only achieves after the user biometric authentication on the SPD will be done successfully.

REFERENCES


