Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 34 (2014) 323 – 329

Procedia Computer Science

The 11th International Conference on Mobile Systems and Pervasive Computing

(MobiSPC-2014)

Mobile Authentication Secure Against Man-In-The-Middle Attacks

Kemal Bicakci[a,] , Devrim Unal[b], Nadir Ascioglu[c,*], Oktay Adalier[c]

[a] *Computer Engineering Department TOBB University of Economics and Technology, Sogutozu, Ankara 06560, Turkey*
[b] *Systems Engineering Department TUBITAK-BILGEM, Gebze, Kocaeli 41470, Turkey*
[c] *eID Department TUBITAK-BILGEM, Gebze, Kocaeli 41470, Turkey*

Abstract

Current mobile authentication solutions put a cognitive burden on users to detect and avoid Man-In-The-Middle attacks. In this paper, we present a mobile authentication protocol named Mobile-ID which prevents Man-In-The-Middle attacks without relying on a human in the loop. With Mobile-ID, the message signed by the secure element on the mobile device incorporates the context information of the connected service provider. Hence, upon receiving the signed message the Mobile-ID server could easily identify the existence of an on-going attack and notify the genuine service provider.

*Keywords:*authentication; phishing; Man-In-The-Middle attack; secure element; mobile signature; security protocol

1. Introduction

Traditional password based user authentication has well known security problems. In particular, even an unsophisticated and remotely-executed phishing attack can be effective for a simple password theft. Due to rise of phishing and related attacks, sensitive applications such as online banking increasingly prefer more secure authentication alternatives. Their solutions usually fall into a group called two-factor authentication. By having a secondary factor, user authentication not only depends on something-you-know but also on something-you-have. As a result, setting up a fake site for collecting user passwords is no longer sufficient to conduct a successful attack.

There is a subtle but important difference between a phishing attack and a Man-In-The-Middle (MITM) attack. While a phishing site is static and passive, a MITM attack is an active attack conducted in real-time. Two-factor

[*] Corresponding author Tel.: +90-262-6481298 ; fax: +90-262-6481100
E-mail address: nadir.ascioglu@tubitak.gov.tr

doi:10.1016/j.procs.2014.07.031

authentication by itself does not provide protection against MITM attacks[†] [1]. There are some solutions (detailed in section 5) proposed against MITM attacks, but in practice they are either not effective or not widely deployed due to usability or other factors.

One trend in recent years that changes the landscape of authentication is the rise of mobile devices. Last year, it was predicted that by the end of 2013 smart phones would overtake PCs as the most common Web access device worldwide [2]. Using the mobile device as the something-you-have factor in two-factor authentication is found to be more usable than carrying something else by many users. However current mobile solutions (e.g., OTP over SMS and mobile signatures) share the drawback of failing to address MITM attacks.

One key difference between mobile and fixed platforms is that a secure element is already available in most of the mobile devices. For authentication purposes, we could use this secure-element (e.g., a SIM card) which is a tamper resistant hardware that is capable of storing credentials such as private keys and perform cryptographic operations securely without keys having to leave the card.

Our Contribution in a Nutshell. By exploiting the signing capability with a secure element on a mobile device, we propose a mobile authentication protocol secure against MITM attacks. Up to our best knowledge, this is the first such effort in the literature.

Organization of the Paper. Section 2 briefly overviews mobile signature solutions deployed in many countries as a secure two-factor user authentication method. Section 3 presents our new protocol named Mobile-ID. Section 4 provides a security analysis and a comparison. Section 5 summarizes the related work and Section 6 concludes.

2. Available mobile signature solutions

Currently implemented mobile signature solutions for user authentication rely on qualified digital certificates. Qualified certificates are defined by the ETSI standards and issued by an authorized Certificate Authority. Recently ETSI has made a draft standard available for public review, which provides the framework for further standardization for the creation and validation of advanced electronic signatures (AdES) in mobile environments [13].

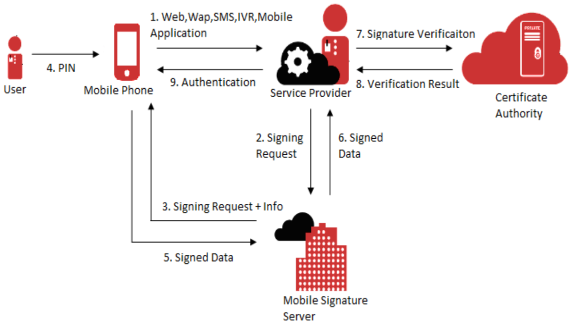

Figure 1. The operation of currently available mobile signature solutions for user authentication.

In the available solutions the service is implemented on EAL4+ certified SIM cards. The certificate is activated over the air when the user has subscribed to the service. Public and private keys are generated for each certificate. The private key is stored on the SIM card. The public key is published in a directory. When the user requests access to a service, to authenticate him/her the service automatically pops up on the user's phone and requests a signature. When the user enters his PIN, the signature is sent to the service provider over the mobile operator. Then, the service provider checks its validity and grants access to the service. Figure 1 presents a step-by-step operation of mobile signatures for user authentication.

[†] MITM attacks have many types. In the literature, one type is referred as real-time phishing attack in which a fake web site collecting user credentials attempts to use these in real-time on the genuine web site.

3. Mobil-ID specification

3.1. Assumptions

1. Mobile-ID is assumed to be functioning on the mobile medium similar to an identity card. Mobile-ID is utilized without any external card reader. A secure element (such as a SIM card, secure SD card, etc.) is targeted as the storage for credentials and processing unit for secure operations. Our main aim is to provide users a more usable and secure alternative to password-based as well as earlier mobile signature-based systems for mobile identity.

2. We assume that the operating environment of Mobile-ID i.e., the mobile terminal is secure. The presence of mobile malware is an increasing threat to the mobile environment. There are a number of studies which could improve the security of authentication in the presence of mobile malware. For instance, studies based on TEE (Trusted Execution Environment) [3] and the similar work have the potential to establish a secure mobile environment. We assume that such a secure environment is already established.

3. As a more special assumption regarding the security of mobile device, we assume that the digital signature feature is accessible only by the Mobile-ID application that could be implemented as a mobile browser plug-in. No other application is allowed by the mobile operating system to interfere with the communication between the secure element and the Mobile-ID application.

4. We do not address privacy aspects within this study.

5. While the presented protocol could easily be tailored as a more traditional authentication method not involving an authentication server, representing our expectation to a shift to the authentication-as-a-service paradigm we assume the existence of a Mobile-ID authentication server; thus, our solution is a three-party authentication protocol that can operate as a single sign-on solution.

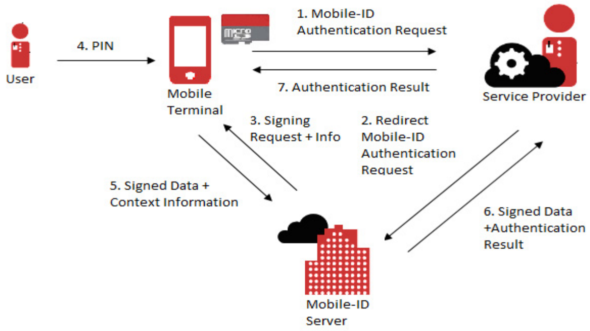
User
4. PIN
Mobile Terminal
1. Mobile-ID Authentication Request
7. Authentication Result
Service Provider
3. Signing Request + Info
2. Redirect Mobile-ID Authentication Request
5. Signed Data + Context Information
6. Signed Data +Authentication Result
Mobile-ID Server

Figure 2. The operation of the Mobile-ID protocol secure against MITM attacks.

3.2. Parties that take part in the Mobile-ID Protocol

The following parties take part directly or indirectly within the proposed Mobile ID protocol: (i) User (U), (ii) Mobile phone or mobile terminal (M), (iii) Secure Element (SE), (iv) Service Provider (SP), (v) Mobile-ID Server (MIS).

3.3. Preparation

The following steps need to be followed before the execution of the protocol:

1. U obtains a secure element (SE) with digital signature capability and inserts it into the mobile terminal (M). The public-private key pair within the SE is securely personalized and certified (which could be performed in way similar to the available mobile signature solutions explained in section 2).

2. U has installed a mobile browser plug-in (or a specialized mobile browser application) which allows for M to communicate with the MIS as well as with the SE. This plug-in implements the mobile terminal part of the protocol.

3.4. Protocol steps

The protocol steps outlined in Figure 2 are detailed below:

1. U determines the SP that he/she wants to connect and enters the URL of the SP into the browser. A connection[‡] is established from U's browser to SP. U clicks on the appropriate link within the web site of SP which allows him/her to login using Mobile-ID credentials. By this way, U chooses a MIS to log in.

2. SP establishes a secure (encrypted and authenticated) connection to the MIS. For this purpose, SSL/TLS protocol could be used. SP redirects U to MIS over the secure connection.

3. MIS sends to M a nonce value, which is to be signed after the required information which follows it.

4. The signed nonce will be used to authenticate U. In order to generate a digital signature, U is required to present the correct PIN value. The PIN is entered by U to M.

5. If the PIN value is correct, before the generation of the digital signature, context information is appended to the nonce value. The context information (CI) specifies the SP M is attempting to connect. The context information could contain the URL of the SP and additional components like IP address, hash of SSL/TLS certificate, etc… See [4] for the discussion of pros and cons of different types of context information[§]. The value which is signed and returned to the MIS is as follows:

$$\text{Signature}_{\text{U}}(\text{nonce} \parallel \text{CI}), \text{CI} \tag{1}$$

6. MIS performs signature verification. Additionally MIS verifies that the context information is correct. For this purpose MIS compares the information of SP which has performed a redirection in step 2 and the SP which U wishes to connect and sends the context information accordingly[**]. If the comparison is successful and the signature is verified, then MIS informs the successful authentication to the SP by a signed message. Additionally, MIS redirects C to SP. The signed message contains the identity information of U (ID_U), identity information of SP (ID_SP) and time information (TI) for protection against replay attacks. The message is signed by MIS. The signed message format is as follows:

$$\text{Signature}_{\text{MIS}}(\text{ID}_\text{U} \parallel \text{ID}_\text{SP} \parallel \text{TI}), \text{ID}_\text{U}, \text{ID}_\text{SP}, \text{TI} \tag{2}$$

7. SP verifies the signed message sent by MIS for the confirmation of successful authentication of M.

4. Security analysis and discussion

In this section we present an informal security analysis of both earlier signature based solution and the proposed protocol against MITM attacks.

Suppose that M has attempted a connection to a web site SP' which impersonates SP. SP' is a man in the middle i.e., SP' connects to SP as well as M.

[‡] This connection is established over an encrypted channel. However we do not assume the existence of an authenticated channel.

[§] For instance using URL as the only context information is quite sufficient to avoid a real-time phishing attack. On the other hand, a pharming attack which poisons the DNS cache could not be avoided by the URL information.

[**] As mentioned previously, our proposed solution also works if a trusted third party server does not exist. If the protocol does not involve a separate MIS and U is directly authenticated by the SP, the context information could be verified by SP instead of MIS.

1. With the mobile signature solution, since the SP' connects to SP the information appended to the signature request (step 3 in Figure 1) contains the identity information of SP (e.g., URL information). This information is displayed to U who is expected to check whether it is as same as the connected party – SP' - and if not decides not to enter the PIN. However it was shown in earlier work [5] that this type of user involvement is not reliable. If the user fails to notice the mismatch between SP and SP' and enters the PIN, with the digital signature SP' could impersonate itself as the legitimate user to SP. Keeping the user outside the security loop and preventing MITM attacks without any expectation for correct user behavior is a more desired approach from usable security point of view.

2. With the Mobile-ID protocol, the information obtained by M and forwarded to MIS contains the context information of SP'. The context information obtained by MIS when SP connects to MIS would be different than the information of SP'. In step 6 of the proposed protocol, MIS detects this difference and therefore the attack is stopped. SP is warned which breaks the connection with SP', thus preventing the MITM attack without any user involvement.

For a better understanding of the key difference between the two protocols, we provide the following example:

Suppose Alice receives a phishing email that pretends to be her bank *gbank.com* but contains a link to the bogus site *ggank.com*. With her mobile device, she clicks on the link and opens *ggank.com* while thinking she is visiting *gbank.com*. The bogus site immediately opens a session on the real bank website *gbank.com* to impersonate Alice and commit a fraud. Since Alice chooses to use Mobile-ID, the bogus site has no choice but to mimic it on the real site. Then, the real bank web site establishes a secure connection to MIS to authenticate Alice. MIS obtains a signed message from Alice's machine which contains the information that Alice is connected to *ggank.com*. MIS could easily see the mismatch between the real and bogus web sites and informs *gbank.com* for this incident. As a result, a real-time phishing attack is prevented.

4.1. Communicating with Secure Element

Regarding accessing the signing capability of the secure element (SE) on mobile devices, we identify two use cases:

Remote communication with SE: In the mobile signature solutions described in section 2, the mobile operator establishes a communication with the secure element (SIM card) remotely (using encrypted SMS messages) bypassing the mobile terminal. Regarding the content of the signed message this approach has the advantage of being immune to a malware on the mobile terminal. However, it rejects any opportunity to append any context information prior to signing i.e., the context information of the connected service provider is available only locally.

Local communication with SE: On the other hand, as we have seen, accessing the SE from a local application offers the advantage of realizing a protocol secure against MITM attacks. The downside is that now security of the mobile environment becomes an issue of concern.

We note that the former approach does not eliminate the need for a secure client platform. For instance, a malicious software running in the background on the client machine could generate a fake transaction after the authentication step is completed [10].

A disadvantage with the later approach is that Mobile-ID protocol could only work if the user accesses the service using his/her mobile device. The existing solutions allow users to access services using other devices such as a PC and the use of mobile device could be only for the digital signature generation.

We believe the trade-off between local versus remote communication with SE is interesting and requires further investigation. We finish this section with a short discussion on the use of a Mobile-ID server in the Mobile-ID protocol.

4.2. Mobile-ID Server

The use of a Mobile-ID server allows users to consolidate their digital identities. Users may register with their preferred server and then use it for authenticating themselves to any web site which accepts the Mobile-ID protocol. Actually, the described Mobile-ID protocol resembles to the OpenID standard [11]. The main difference is that while OpenID standard specifies an authentication-neutral protocol but in practice usually depends on passwords shared between users and the server, the Mobile-ID protocol represents a public-key based authentication protocol in which the users and the server do not share a long-term secret (the password).

5. Related Work

In the literature, there are many proposals to replace traditional password based authentication. Bonneau et al. [6] performed a comprehensive and comparative evaluation of 35 of these according to 25 criteria grouped under security, usability and deployability properties. Interestingly, in their work prevention of MITM attacks was not one of the security criteria. We attribute this choice to the difficulty of taking an effective precaution against MITM attacks.

Traditionally, MITM attacks are considered not to be avoided by the design of a user authentication protocol but by authenticating the server. In theory, server authentication is straightforward with the SSL/TLS protocol. The underlying assumption of SSL/TLS protocol is that the server public key is certified by a trusted third party. If such a certification does not exist, the common approach implemented in many browsers is to warn users against a possible security breach. If the warning message is ignored by the user (and we know many of them do so [7]), then a MITM could easily be conducted.

An overview of mobile authentication solutions and approaches in European countries could be found in [8]. None of these solutions have addressed MITM attacks.

A recent work by Ben-David et al. [4] addresses MITM attacks by contextual one-time passwords (XOTPs). In their solution, a XOTP device communicates with the browser over Bluetooth. A shared key is used to generate one-time passwords cryptographically entangled with the session context. Our Mobile-ID protocol is similar to XOTP in the use of context information to avoid MITM attacks but eliminates the need to set up a communication between an external device and the browser.

Security is usually an arms race. Given the resistance of two-factor authentication against standard phishing attacks, attackers start exploiting MITM attacks. See [9] for an example.

6. Conclusion

User authentication could be broken by malicious software running on the client machine and by attacks performed remotely from the network. If trusted platforms for mobile devices could provide the trust environment eliminating the attacks exploiting the insecurity of the client machine, we should expect the change of the threat picture and the rise of even more sophisticated network attacks such as MITM attacks.

Many solutions for MITM attacks rely on users' awareness. However since human error is the main source of security failures, a solution performing its function automatically without user involvement is desired. In this work, we propose Mobile-ID, a protocol which prevents MITM attacks while keeping the human outside the security loop [12]. The proposed protocol carries the context information of the man in the middle from the mobile client to the Mobile-ID server which then compares this information with the information belonging to the intended service provider and stops the protocol by notifying the mismatch. We are currently working on implementing the Mobile-ID protocol and integrating it with the OpenID standard.

References

1. B. Schneier, "Two-Factor Authentication: Too Little, Too Late," Comm. ACM, vol. 48, no. 4, Apr. 2005, p. 136.
2. http://www.gartner.com/newsroom/id/2209615, Last access: 04/04/2013.
3. http://www.globalplatform.org/mediaguidetee.asp, Last access: 04/04/2014.
4. A. Ben-David, O. Berkman, Y. Matias, S. Patel, C. Paya, M. Yung: Contextual OTP: Mitigating Emerging Man-in-the-Middle Attacks with Wireless Hardware Tokens. ACNS 2012: 30-47
5. M. Alzomai, B. AlFayyadh, A. Josang, A. McCullag, An Experimental Investigation of the Usability of Transaction Authorizationin Online Bank Security Systems, Proc. of Australasian Security Conf, Jan 2008.
6. J. Bonneau, C. Herley, P.C. van Oorschot, F. Stajano. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. IEEE S & P, May 2012.

7. J. Sunshine, S. Egelman, H. Almuhimedi, N. Atriand L. F. Cranor. Crying Wolf: An Empirical Study of SSL Warning Effectiveness. Usenix Security, 2009.
8. Mobile eID, Stork Work Item 3.3.6, https://www.eid-stork.eu/, Last access: 04/04/2014.
9. http://nakedsecurity.sophos.com/2013/04/19/anatomy-of-a-phish-how-to-spot-a-man-in-the-middle/, Last access: 04/04/2014.
10. C. Jackson, D. Boneh, and J. Mitchell. Transaction generators: Root kits for web. In 2nd USENIX Workshop on Hot Topics in Security (HotSec'07), pages 1–4. USENIX Association, 2007.
11. OpenID standard, http://openid.net/specs/openid-authentication-2_0.html, Last access: 04/04/2014.
12. L.F. Cranor, A framework for reasoning about the human in the loop, In Proceedings of the 1st Conference on Usability, Psychology and Security, USENIX Association, 2008.
13. ETSI, Rationalised Framework of Standards for Advanced Electronic Signatures in Mobile Environment, Draft SR 019 020 V0.0.4, 2013.