Authentication Protocol for Wearable Devices Using Mobile Authentication Proxy

Doo-Hee Hwang Jin-Myeong Shin Yoon-Ho Choi
School of Computer Science and Engineering, Pusan National University, Busan, 26241, Republic of Korea
dooheeh@pusan.ac.kr sinryang@icloud.com yhchoi@pusan.ac.kr

***Abstract*—The data transmitted from the wearable device commonly includes sensitive data. So, application service using the data collected from the *unauthorized* wearable devices can cause serious problems. Also, it is important to authenticate any wearable device and then, protect the transmitted data between the wearable devices and the application server. In this paper, we propose an authentication protocol, which is designed by using the Transport Layer Security (TLS) handshake protocol combined with a mobile authentication proxy. By using the proposed authentication protocol, we can authenticate the wearable device. And we can secure data transmission since session key is shared between the wearable device and the application server. In addition, the proposed authentication protocol is secure even when the mobile authentication proxy is *unreliable*.**

***Index Terms*—authentication, wearable device, proxy, handshake protocol.**

I. Introduction

Wearable devices are used in various fields such as healthcare, fitness, infotainment, and industrial [1], [2]. The current wearable devices overcome the limitation of communication distance by connecting personal devices such as mobile and tablet PC as intermediate device [3]. However, this communication structure overlooks wearable device authentication and does not provide good-enough security between wearable device and intermediate device. So, the data may be exposed to an external attacker. Also, application server may receive incorrect data from an unauthorized wearable device. If the received data itself is manipulated, the application server will operate abnormally. To solve these problem, an application service using the wearable device should operate with a secure authentication protocol [4].

Currently, many studies have been done to solve security problems of the wearable device. However, most of these studies focus on authenticating users of wearable devices which uses a password, biometric data, and so on as authentication methods[5], [6]. Also, the existing method is limited to a scenario in which the wearable device can communicate directly with the application server. But, the existing method is not practical since wearable devices use short range communication such as NFC or Bluetooth.

In this paper, we propose the wearable device authentication protocol using the mobile authentication proxy instead of the existing intermediate device only for data forwarding. In the proposed authentication protocol, we can authenticate the wearable device using the TLS handshake protocol which is verified protocol since the mobile authentication proxy performs the TLS handshake protocol instead of the wearable device. Since the proposed authentication protocol establishes the session key after the authentication process is completed, the wearable device can securely communicate with the application server. And the application server can receive reliable data from the authenticated wearable device.

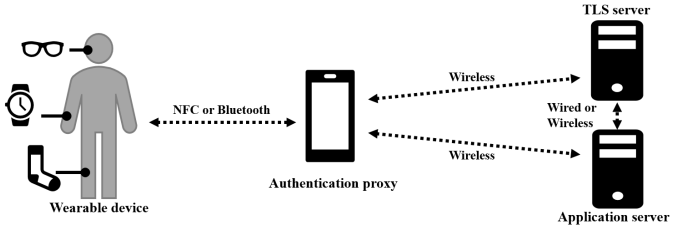

Fig. 1. Communication structure using the mobile authentication proxy

This paper consists of as follows. In section II, we overview the current authentication methods for the wearable device. In section III, we describe the proposed protocol in details. In section IV, we discuss security of the proposed protocol. We also describe prototype implementation of the proposed protocol. Finally, we summarize this paper in section V.

II. Related Work

The authentication method proposed in [8] authenticates the wearable device only to the intermediate device. That is, the wearable device cannot be considered authenticated to the remote server. Therefore, this authentication method requires the assumption that the intermediate device is always reliable. Otherwise, it may be a problem with the data transfer process.

In [9], an authentication protocol is proposed in which a wearable device can perform mutual authentication with another wearable device, an intermediate device, and a remote server. This authentication protocol does not specify the security channel setting process after authentication. Therefore, this method has a risk that data is exposed to the outside during communication. In addition, the wearable device performs the public key algorithm directly in the authentication process of this protocol. These operations are a considerable burden on the wearable device.

III. Proposed Authentication Protocol

In this section, we describe the operation and message structure of the proposed authentication protocol in details. To solve

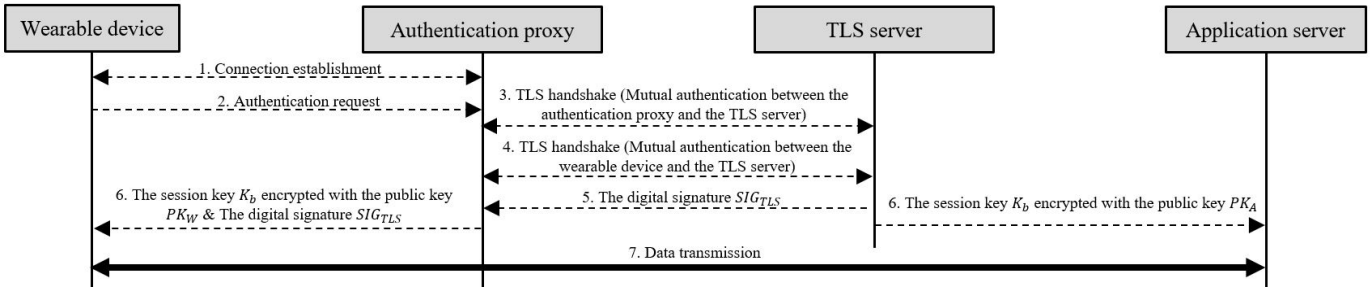

Fig. 2. The proposed authentication protocol

problem of the existing wearable device authentication, end-to-end authentication must be performed using the authentication protocol such as TLS [10]. However, wearable devices are not suitable for performing TLS handshake protocol directly because of limited performance. But, the proposed authentication protocol can authenticate the wearable device using the TLS handshake protocol since the mobile authentication proxy performs the TLS handshake protocol instead of the wearable device. Also, the proposed authentication protocol provides a secure communication channel by sharing the session key.

Fig. 1 shows the communication structure using the mobile authentication proxy. The mobile authentication proxy stores and manages the certificates of wearable devices and performs the TLS handshake protocol instead of the wearable device in the mutual authentication process between the wearable device and the TLS server. Since the mobile authentication proxy has better processing power than the wearable device, TLS handshake protocol does not have much load to the mobile authentication proxy.

A. Authentication Scenario

Since the intermediate device generally is the user's personal device, authentication between a wearable device and an intermediate device is skipped. However, the mobile authentication proxy is not always a trusted device. In the proposed protocol, we do not assume that the mobile authentication proxy is trusted. The proposed authentication protocol is secure, even if the mobile authentication proxy is unreliable. The certificate of the wearable device is sent to these mobile authentication proxy and stored. Even if the untrusted mobile authentication proxy obtains the certificate of wearable device, the information cannot be acquired other than the public key.

In the proposed authentication protocol, we assume that the TLS server knows the trusted public key PK_A of the application server and the wearable device knows the trusted public key PK_{TLS} of the TLS server. In Table I, we summarize terms and their notation used in this paper. The operation process of the proposed authentication protocol is shown in Fig. 2. And the description of each step is described in detail below.

1) An NFC or Bluetooth connection is established between the wearable device and the mobile authentication proxy.

TABLE I
TERMS AND NOTATION

Term	Notation
$CERT$	Certificate
PK	Public key
SK	Session key generated by the TLS handshake protocol
K	Private key
ID	Identifier
TS_i	Timestamp
$Lifetime_i$	Valid lifetime of a signature or session key
SIG_{TLS}	Digital signature of the TLS server
$E(X,[Y])$	Encrypt message content Y using key X

2) The wearable device requests the authentication to the mobile authentication proxy. This message consists of ID_{WD}, ID_{AP}, and TS_1.

3) The authentication proxy performs the TLS handshake protocol with the TLS server. At this time, the host certificate of the TLS handshake protocol means the certificate $CERT_{TLS}$ of the TLS server. And the client certificate means the certificate $CERT_P$ of the mobile authentication proxy. If the mobile authentication proxy is an unauthenticated device, the TLS handshake protocol fails mutual authentication. When the TLS handshake protocol is completed, the mobile authentication proxy and the TLS server are mutually authenticated, and a session key SK_A is generated.

4) In this step, the mobile authentication proxy, that has completed the mutual authentication with the TLS server and stores the certificate $CERT_W$ of the wearable device, performs the handshake protocol instead of the wearable device. That is, the mobile authentication proxy performs the TLS handshake protocol using the certificate $CERT_W$ of the wearable device. When the TLS handshake protocol is successfully completed, the wearable device and the TLS server are mutually authenticated, and a session key SK_B is generated.

5) The digital signature SIG_{TLS} generated by K_{TLS} is transmitted to the authenticated authentication proxy. The $Lifetime_2$ included in this message means the period in which SIG_{TLS} is valid.

TLS server→Proxy: $SIG_{TLS}||ID_{TLS}$

$SIG_{TLS} = E(K_{TLS}, [ID_{WD}||ID_{AP}||TS_2||Lifetime_2])$

6) To provide secure end-to-end secure channel, the mobile authentication proxy encrypts SK_B and sends it to the wearable

device along with the digital signature SIG_{TLS} of the TLS server. Also the TLS server encrypts SK_B and sends it to the application server. The wearable device decrypts the encrypted message with its own private key K_W to obtain the SK_B. Also, by using the SIG_{TLS} and the PK_{TLS}, it can be confirmed that the mobile authentication proxy is authenticated. Therefore, the wearable device can trust the transmitted the session key SK_B. Also the application server can decrypt the encrypted message with its own private key K_A to obtain it.
Proxy→Wearable: $E(PK_W, [K_B||ID_{AS}||TS_3||Lifetime_3])$
TLS server→App: $E(PK_A, [K_B||ID_{WD}||TS_3||Lifetime_3])$

7) Since the wearable device and the application server encrypt and transmit the data with the session key SK_B, it is possible to prevent the transmission data from being exposed to the outside.

IV. Discussion and Prototype Implementation

In this section, we discuss security of the proposed authentication protocol and implement the prototype of the proposed authentication protocol. The proposed authentication protocol works based on the TLS handshake protocol which is verified protocol. In this paper, we omit discussion for security of the TLS handshake protocol.

The intermediate device is not always a reliable device. Therefore, it should be considered that the proposed authentication protocol is safe even if the mobile authentication proxy used as the intermediate device is unreliable. In the followings, we describe the possible three attacks by an untrusted intermediate device and strengths of the proposed protocol against them.

1) The malicious mobile authentication proxy obtains the $CERT_W$: Even if the malicious mobile authentication proxy obtains the $CERT_W$, the information cannot be acquired other than the PK_W.

2) The malicious mobile authentication proxy obtains the $CERT_W$ and attempts mutual authentication with TLS server : The malicious mobile authentication proxy is not authenticated to the TLS server. So, the malicious mobile authentication proxy cannot perform the next step.

3) The malicious mobile authentication proxy generates the untrusted session key and transmits it to the wearable device : The malicious mobile authentication proxy cannot provide a digital signature SIG_{TLS} of the TLS server, so the wearable device does not trust the session key.

The wearable device is hardly involved in the proposed authentication protocol. The wearable device simply sends an authentication request message and decrypts the encrypted session key. Since the mobile authentication proxy performs all calculations required in the authentication process on behalf of the wearable device.

We implemented the wearable device, the mobile authentication proxy, and TLS server for a prototype of the proposed authentication protocol. The wearable device was developed using Arduino Pro mini 328, Bluetooth module HC-06, Arduino Sketch, and so on. The mobile authentication proxy was developed using Android Studio and the Java Secure Socket Extension(JSSE). The TLS server was developed using the GNU Transport Layer Security Library(GnuTLS). To show how the prototype of the proposed authentication protocol works, we provide a supplement material, i.e., a video record, at the following link[1]. If you copy url address, please check the copied url address again.

V. Conclusion

In this paper, we proposed authentication protocol using the mobile authentication proxy combined with the TLS handshake protocol. In the proposed authentication protocol, we can authenticate the wearable device using the TLS handshake protocol which is verified protocol since the mobile authentication proxy performs the TLS handshake protocol instead of the wearable device. Since the proposed authentication protocol establishes the session key after the authentication process is completed, the wearable device can securely communicate with the application server. And the application server receives reliable data from the authenticated wearable device. In addition, the proposed authentication protocol is secure even when the mobile authentication proxy is *unreliable*.

Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(2014-1-00743) supervised by the IITP(Institute for Information & communications Technology Promotion) and basic science research program through national research foundation korea (NRF) funded by the ministry of science, ICT and future planning (NRF-2015R1D1A1A01057888).

References

[1] H. Ye, M. Malu, U. Oh, and L. Findlater, "Current and future mobile and wearable device use by people with visual impairments", *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3123-3132, 2014.

[2] O.D. Lara and M.A. Labrador, "A survey on human activity recognition using wearable sensors", *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1192-1209, 2013.

[3] J. West, T. Kohno, D. Lindsay, and J. Sechman, "Wearfit: Security design analysis of a wearable fitness tracker", Technical report, IEEE Center for Secure Design, 2016.

[4] A. Bianchi and I. Oakley, "Wearable authentication: Trends and opportunities", *it-Information Technology*, vol. 58, no. 5, pp. 255-262, 2016.

[5] R. Amin and G.P. Biswas, "A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks", *Ad Hoc Networks*, vol. 36, pp. 58-80, 2016.

[6] Q. Jiang, J. Ma, C. Yang, X. Ma, J. Shen, and S.A. Chaudhry, "Efficient end-to-end authentication protocol for wearable health monitoring systems", *Computers & Electrical Engineering*, 2017.

[7] oneM2M-TR-0008, "Security", v. 2.0.0, Aug, 2016.

[8] M. Shin, "Secure Remote Health Monitoring with Unreliable Mobile Devices", *BioMed Research International*, 2012.

[9] F.P. Diez, D.S. Touceda, J.M.S. Camara, and S. Zeadally, "Toward self-authenticable wearable devices", *IEEE Wireless Communications*, vol. 22, no. 1, pp. 36-43, 2015.

[10] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol", RFC 5246, 2008.

[1] https://1drv.ms/v/s!Ar3jtLtwPvN0ab7pBNR4vsoB_A8