Design and Implementation of a multi-factor web authentication system with MyNumberCard and WebUSB

Yuki Fujita[1,a)] Atsuo Inomata[2,b)] Hiroki Kashiwazaki[3,2,c)]

Abstract: As the number of Internet users increases, their usage also diversifies, and it is important to prevent Identity on the Internet (Digital Identity) from being violated. Unauthorized authentication is one of the methods to infringe Digital Identity. Multi-factor authentication has been proposed as a method for preventing unauthorized authentication. However, the cryptographic authenticator required for multi-factor authentication is expensive both financially and UX-wise for the user. In this paper, we design, implement and evaluate multi-factor authentication using MyNumberCard provided by public personal identification service and WebUSB, which is being standardized.

1. Background and objectives

Since the 2000s the number of Internet users has increased because of the spread of broadband and mobile Internet. Furthermore, in the 2010s because of the rapid spread of smartphones, personalization and mobilization of Internet access have advanced [1]. Their usage is also diversified. Among the functions and services that people used on the Internet, social networking services, free call applications and voice chats, video posting/sharing sites, use of map/-traffic information provision services, and purchasing/trading of products/services can be mentioned as the services highly used. In the use of these services, the user can be distinguished from other users to be dealt as individuals.

In the context of Digital Identity on the Internet, authentication establishes that the subject (including the user) attempting to access the service on the Internet is under the control of the technology used for authentication [2]. It is. If the subject revisits the service and is successfully authenticated, then a reasonable risk-based guarantee is provided that the subject accessing the service is the same as the subject that previously accessed the service. There are various technical issues in authentication on the Internet. As a result, impersonation and other attacks may occur, leading to unauthorized authentication.

Table 1 classifies threat/attack methods for which unauthorized authentication is performed and methods used for its mitigation. Authentication using multiple factors can make it more difficult for an attack to succeed. Multiple elements consist of knowledge, possession, and inherence[*1]. When an authentication service requires the knowledge, the subject is needed to provide secret knowledge to authenticate. Passwords, passphrases, and PINs are representative examples. When the service requires posession, the subject needs to provide a key for the lock. The basic principle is to embody the secret in which the key is shared between the lock and the key. Security tokens are a representative example. When the service requires inherence, a subject needs to provide its own unique factors. Fingerprints, faces, sounds, and irises are its examples. There are also examples of using behavioral biometrics represented by keystroke dynamics.

For example, when multi-factor authentication is performed using Google Authenticator, the user can authenticate the stored password and the 6-digit number (Time-based One-Time Password: TOTP) generated by Google Authenticator at regular intervals. Use It may be estimated that the cost for the attacker to discover both the terminal on which the Google Authenticator used by the user operates and the stored password is sufficiently high. On the other hand, in multifactor authentication using a cryptographic authenticator, as the number of services performing authentication increases, the cost for the user to search for the TOTP of the service also increases. The smartphone personal ownership rate has increased by more than 10% overall between 2013 and 2017. However, while in the 20s

[1] Georepublic Japan, Kobe, Hyogo, 658-0081, Japan
[2] Osaka University, Ibaraki, Osaka, 567–0047, Japan
[3] National Institute of Informatics, Chiyoda, Tokyo 101–8430, Japan
[a)] fujita@georepublic.de
[b)] inomata@mail.osaka-u.ac.jp
[c)] reo_kashiwazaki@nii.ac.jp

[*1] [FYI] Multi-factor authentication `https://en.wikipedia.org/wiki/Multi-factor_authentication`

Authenticator Threat/Attack	Threat Mitigation Mechanisms
Theft	• Use multi-factor authenticators that need to be activated through a memorized secret or biometric. • Use a combination of authenticators that includes a memorized secret or biometric.
Duplication	Use authenticators from which it is difficult to extract and duplicate long-term authentication secrets.
Eavesdropping	• Ensure the security of the endpoint, especially with respect to freedom from malware such as key loggers, prior to use. • Avoid use of non-trusted wireless networks as unencrypted secondary out-of-band authentication channels. • Authenticate over authenticated protected channels (e.g., observe lock icon in browser window). • Use authentication protocols that are resistant to replay attacks such as pass-the-hash. • Use authentication endpoints that employ trusted input and trusted display capabilities.
Offline Cracking	• Use an authenticator with a high entropy authenticator secret. • Store memorized secrets in a salted, hashed form, including a keyed hash.
Side Channel Attack	Use authenticator algorithms that are designed to maintain constant power consumption and timing regardless of secret values.
Phishing or Pharming	Use authenticators that provide verifier impersonation resistance.
Social Engineering	Avoid use of authenticators that present a risk of social engineering of third parties such as customer service agents.
Online Guessing	• Use authenticators that generate high entropy output. • Use an authenticator that locks up after a number of repeated failed activation attempts.
Endpoint Compromise	• Use hardware authenticators that require physical action by the subscriber. • Maintain software-based keys in restricted-access storage.
Unauthorized Binding	Use MitM-resistant protocols for provisioning of authenticators and associated keys.

Table 1 Classification of the authenticator threats and the threat mitigation mechanisms [2]

and 30s, 90% or more of the users own smartphones, the percentage of smartphone owners in 70s is 18.8%, and the smartphone ownership rate in 80s is 6.1%. Is large. According to a Gfk survey, the average selling price of smartphones sold between October and December 2018 is $ 384[*2].

At the same time, electronic certificates based on public personal identification services are issued worldwide. This electronic certificate can be obtained with only a few dollars fee. In Japan, the personal number card (common name: MyNumberCard) issued based on the "Law on the Use of Numbers to Identify Specific Individuals in Administrative Procedures" corresponds to this. Therefore, in this paper, we design, implement and evaluate a system that introduces multi-factor authentication using a MyNumberCard with an authentication system (Web authentication system) used for services on the Internet, in particular, on the World Wide Web. The proposed method emphasizes that it does not depend on a specific OS, and uses WebUSB.

2. Related Works

As shown in Table 1, as a method of alleviating the threat of authentication, a method using biometrics authentication in multi-factor authentication has also been proposed. As an example of implementation, Apple's fingerprint authentication (Touch ID) provided by iOS can be used on all iPhones officially sold in Japan. Also, although it is some models, face recognition (Face ID) is also implemented. Android 9 Pie also provides BiometricPrompt API[*3] from Android 9 Pie, and supports face authentication and iris authentication as well as fingerprint authentication.

In addition to password authentication, SMS authentication using Short Message Service (SMS) messages is an implementation of multi-factor authentication. Unlike the method using the encryption authenticator, the cost for the user to find the TOTP of the service is sufficiently low. There is no burden on the user because SMS is free to receive, but since transmission is chargeable, the maintenance cost of the authentication system will increase for the service provider as the number of users increases. Also, according to the above-mentioned NIST Special Publication 800-63B [2], the out-of-band authenticator including SMS authentication is considered as RESTRICTED. In August 2018, a case of an attack on Reddit that was considered to be due to SMS intercept was reported[*4].

Native Messaging[*5] is an API available to some browsers.

[*2] Global smartphone sales reached $522 billion in 2018 https://www.gfk.com/insights/press-release/global-smartphone-sales-reached-522-billion-in-2018/

[*3] Android Developers BiometricPrompt https://developer.android.com/reference/android/hardware/biometrics/BiometricPrompt

[*4] Reddit Breach Highlights Limits of SMS-Based Authentication - Krebs on Security https://krebsonsecurity.com/2018/08/reddit-breach-highlights-limits-of-sms-based-authentication/

[*5] Native Messaging - Google Chrome https://developer.chrome.com/apps/nativeMessaging or Extensions - Native messaging - Microsoft Edge Development — Microsoft Docs

By using this API, an application or browser extension operating on the browser can exchange messages with a native application operating on the OS. Using this API, a Web application can exchange messages with an IC card via an application that exchanges messages with the IC card using a card reader recognized by the OS. However, it can not be denied that Native Messaging is opaque in terms of not being standardized internationally.

3. Design and Implementation

3.1 MyNumberCard

MyNumberCard is an IC card of ISO/IEC 7810:2003[*6] ID-1 standard. The card has an integrated circuit embedded and is equipped with an ISO/IEC 14443[*7] Type B RFID that supports contactless card readers. The digital certificate of MyNumberCard consists of a digital certificate for signing and a digital certificate for user certification. The signature electronic certificate is used to check if the document has been tampered with, for example, when sending an electronic document over the Internet. The law[*8] stipulates the standard for the electronic signature or electronic user certification to be satisfied for public personal identification service[*9]. In this standard, the system of electronic signature and the system of electronic user certification (encryption method) are RSA [3], and the composite number (key length) as modulus is 2048 bits.

In order to perform authentication using this key, Fig. 1 shows a schematic diagram of a proposed protocol in which a public key is registered on a service side that wants to use it and authentication is performed using a secret key stored in a MyNumberCard. The user prepares a client machine capable of reading ISO/IEC 14443 type B. When the user connects to a server providing Internet service, a user authentication screen is displayed. The following is the procedure until the user succeeds in the authentication.

1. A user enters a username and submit it.
2. The client machine sends this entered username to the server.
3. The server returns a random number string associated with the username.
4. The client machine receiving this random number sequence requests the user to input a PIN and contact the card (to the reading device).
5. The user enters a PIN.
6. The user contacts the card.
7. The client machine receiving the PIN sends the PIN to the card.
8. If the entered PIN is correct, the card returns a successful authentication to the client machine.
9. The client machine sends the random number sequence received from the server to the card.
10. The card returns a random number sequence encrypted with the secret key.
11. The client machine returns the random number sequence received from the server and the encrypted random number sequence received from the card to the server.
12. The server checks three items below:
    - A. The random number sequence sent to the client machine matches the random number sequence received from the client machine (unencrypted)
    - B. Whether the received encrypted random number sequence can be decrypted with a pre-registered public key
    - C. Whether the decrypted random number sequence matches the random number sequence sent to the client machine
13. Once all the checks in a. To c. Above are done, the server returns a JSON Web Token to the client machine.
14. The client machine notifies the user of login success from this JSON Web Token.

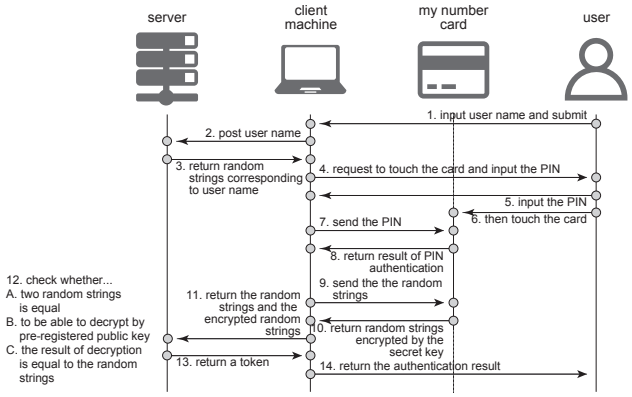

Fig. 1 Schematic diagram of service authentication protocol using MyNumberCard

The ActiveX signature transmission module used in e-Tax and authentication in the Java execution environment is also considered to perform authentication using the same communication as of this schematic diagram. In order to perform processing using a secret key stored in an IC card using ISO / IEC 14443 type B, an IC card reader is required, and the OS performs input/output processing to the IC card

https://docs.microsoft.com/en-us/microsoft-edge/extensions/guides/native-messaging

*6 ISO/IEC 7810:2003 - Identification cards – Physical characteristics https://www.iso.org/standard/31432.html

*7 ISO/IEC 14443-1:2018 - Cards and security devices for personal identification – Contactless proximity objects – Part 1: Physical characteristics https://www.iso.org/standard/73596.html

*8 The act on Certification Services of Local Government Information Systems Organization Related to Electronic Signatures, etc. (Vol.153, 2002) revised by the act on Development, etc. of Related Laws with Enforcement of Act on Use of Numbers to Identify Specific Individuals in Administrative Procedures (Vol.28, 2013)

*9 Technical standards for the implementation of certification services and tasks incidental thereto. http://www.soumu.go.jp/main_content/000392344.pdf

reader. Needs a driver. In the conventional method using ActiveX, the client machine OS is limited to Windows, and authentication in the Java execution environment requires the Windows Internet Explorer 32-bit version, Mac Safari, and a specific version of Java. For this reason, despite the provision of an inexpensive and widely distributable authentication means called MyNumberCard, the environment in which it can be used has been extremely limited. In order to provide an interface that can be widely used by the public, such as MyNumberCard, it is important to use a more versatile device, etc., and by applying a module that does not depend on the operating system or API, etc. In an effort to solve the problem, we decided to use the famous WebUSB as an interface that allows direct control of input/output devices and drivers from a common Web browser. In the next section, we describe a system using WebUSB.

3.2 WebUSB

WebUSB is a JavaScript API that provides secure access from Web pages to USB devices, which is being standardized by W3C[*10]'s Web Platform Incubator Community Group[*11]. As of May 20, 2019, Google Chrome, Chromium, and Opera support WebUSB. Microsoft Edge, Microsoft's standard OS browser, has been announced to be based on Chromium[*12], and it is expected that Microsoft Edge will also be able to use WebUSB. On the other hand, in consideration of security, the connection with the wireless LAN communication device or the smart card reader is designed to be rejected, so it is necessary to explicitly specify and connect VendorID or ProductID depending on the connected device. Note that if a driver dedicated to the connected device is loaded in the OS, it can not be used from the browser.

In this paper, we targeted RC-S380[*13] as an IC card reader for general usage and connection. RC-S380 is an IC card reader called PaSoRi manufactured by Sony Corporation and is a device for reading and writing data contained in various IC cards including Felica, which is an IC card system also developed by Sony. In order to exchange messages with RC-S380 using WebUSB, the driver for RC-S380 was implemented in JavaScript and released on GitHub[*14]. In addition, when implementing the driver, we refer to nfcpy[*15], which is a Python2 NFC driver. Messages of ISO / IEC 14443 type B are exchanged using the WebUSB driver of RC-S380. The implementation of the ISO / IEC 14443 type B message exchange required in Figure 1 of this paper and the implementation of the proposed system is described in the next section.

3.3 Proposed System

Implementation of the proposed system consists of reading public key as authentication certificate, signing function to message using the private key, authentication using a PIN code, API for acquiring MyNumberCard information, basic 4 information (address, Name, age, gender) API for acquisition. The details of the methods required for each function are described below. The implementation of the ISO/IEC 14443 type B message exchange described in this section and the implementation of the proposed system is available on GitHub[*16].

In order to read out MyNumberCard information, it is necessary to input a PIN code for inputting ticket surface information.

3.3.1 getMyNumber

`getMyNumber(pin)` is a method to acquire MyNumberCard information.

1. Connection to MyNumberCard `device.connecToCard()`
2. Select DF `selectCardInfoAP()`
3. Pin EF `selection selectCardInfoPinEF()`
4. Verification of Pin `verifyPin(pin)`
5. Select MyNumberCard EF `selectMyNumberEF()`
6. Read MyNumberCard `readBinary(16)`
7. Communication end `disconnect()`

3.3.2 getPersonalData

As a flow for reading basic 4 information, it is necessary to obtain the size of the data length of personal data. `getPersonalData(pin)` is a method to acquire Personal data in the MyNumberCard.

1. Connection to MyNumberCard `device.connectToCard()`
2. Select DF `selectCardInfor()`
3. Pin EF selection `selectCardInfoPinEF()`
4. Verification of Pin `verifyPin(pin)`
5. Select personal data EF `selectPersonalDataEF()`
6. Read personal data length `len=readBinary(7); parser=ASN1Partial(len)`
7. Reading personal data `readBinary(parser.size)`
8. Communication end `disconnect()`

*10 World Wide Web Consortium (W3C) https://www.w3.org/
*11 Web Platform Incubator Community Group https://www.w3.org/community/wicg/
*12 Microsoft Edge: Making the web better through more open source collaboration — Windows Experience Blog https://blogs.windows.com/windowsexperience/2018/12/06/microsoft-edge-making-the-web-better-through-more-open-source-collaboration/
*13 Sony Global - FeliCa - USB NFC Reader - RC-S380/S https://www.sony.net/Products/felica/business/products/RC-S380.html
*14 aruneko/WebUSB-RC-S380: RC-S380 driver for WebUSB https://github.com/aruneko/WebUSB-RC-S380/
*15 Python module for near field communication - nfcpy 0.13.5 documentationhttps://nfcpy.readthedocs.io/
*16 aruneko/WebUSB-MyNumberCard https://github.com/aruneko/WebUSB-MyNumberCard/

3.3.3 signMessageWithPrivateKey

It is necessary to specify the 4-digit PIN code, the message to be signed, and the algorithm of the hash function used for the message digest.

`signMessageWithPrivateKey(hashType, pin, message)` is a method to sign a message with a private key.

1. Connection to `MyNumberCard device.connecToCard()`
2. Select DF `selectCardInfoAP()`
3. Pin EF selection `selectCardInfoPinEF()`
4. Verification of Pin `verifyPin(pin)`
5. Select secret key IEF `selectRSAPrivateKeyIEF()`
6. Signature `signMessage(hashType, message)`
7. Communication end `disconnect()`

3.4 getPublicKey

Contact MyNumberCard with the IC card reader to acquire the public key of MyNumberCard. Naturally, no certification is required. `getPublicKey` is a method to acquire the public key.

1. Connection to MyNumberCard `device.connecToCard()`
2. Select DF `selectCardInfoAP()`
3. Select EF `selectRSAPublicKeyEF()`
4. Check public key length size `checkPublicKeyLength()`
5. Get public key `readBinary(publicKeyLength)`
6. Communication end `disconnect()`

4. Evaluation

In order to evaluate the superiority of the proposed method over the existing one, we qualitatively compare the existing method and the proposed one with password and TOTP. The existing method inputs a username and a password to the web service. When this authentication is successful, the screen changes, and a screen for inputting TOTP is displayed. Take your smartphone out of the box (unlock it) and launch the Authenticator application. Find the relevant service among multiple services displayed by the started application. This work cost is proportional to the total number of registered services. After finding the corresponding service, memorize the displayed TOTP and enter it (Figure 2).

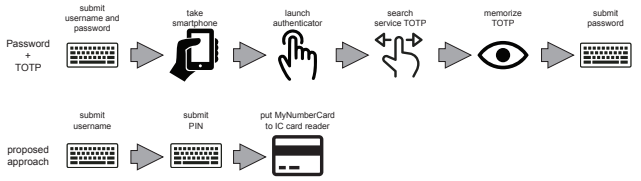

Fig. 2 Qualitative comparison between a password with TOTP and a proposed approach

On the other hand, in the proposed method, when the user name is input, the user is prompted to input a PIN, and when MyNumberCard is applied to the IC card reader, the authentication ends. The operation cost is not proportional to the total number of registered services, but is constant. There may be a dispute that few people carry MyNumberCards, but those who do not possess a driver's license or a passport have a high probability of carrying MyNumberCards. It can be argued that few people carry smartphones in the first place. Anyway, it is a problem that is solved by the spread of MyNumberCard, and it is expected that the spread will be supported by the development of various services using MyNumberCard.

5. Conclusion

Multi-factor authentication has been proposed as a method for preventing unauthorized authentication in Web services on the Internet. In this paper, we propose an authentication method using WebUSB and MyNumberCard as an inexpensive multi-factor authentication method that can be realized on various platforms, and designed and implemented an authentication system for Web services. We also compare qualitatively with the existing password and multi-factor authentication by TOTP, and show that the low operation cost of the proposed method becomes apparent especially when the number of Web services performing authentication increases. In the future, we will conduct code-based quantitative comparisons and quantitative comparisons of authentication strengths, as well as design and implementation of authentication modules for various Web application frameworks.

Acknowledgments Part of this work was carried out under the Cooperative Research Project Program of the Research Institute of Electrical Communication, Tohoku University. The research was supported by ROIS NII Open Collaborative Research 19FA08, JSPS KAKENHI Grant Number JP19K20256.

References

[1] of Internal Affairs, M. and Communications, J.: *Information and Communications in Japan WHITE PAPER 2018*, WHITE PAPER - Information and Communications, Nikkei Publishing (2018).
[2] Grassi, P. A., Fenton, J. L., Newton, E. M., Perlner, R. A., Regenscheid, A. R., Burr, W. E., Richer, J. P., Lefkovitz, N. B., Danker, J. M., Choong, Y.-Y., Greene, K. K. and Theofanos, M. F.: Digital identity guidelines: authentication and lifecycle management, Technical report (2017).
[3] Rivest, R. L., Shamir, A. and Adleman, L.: A Method for Obtaining Digital Signatures and Public-key Cryptosystems, *Commun. ACM*, Vol. 21, No. 2, pp. 120–126 (online), DOI: 10.1145/359340.359342 (1978).