Two/Multi-Factor Authentication (2FA/MFA): Wearable Devices

Wei Kuan Chua, Hong Jing Chua, Swee Khoon Tan, Junzhi Yeo

National University of Singapore, Singapore

Abstract. With the advancement in technology, many physical transactions are increasingly being executed online. More importantly, these transactions usually involve money, confidential corporate data or even personal information. Certainly, users will be concerned about the potential losses of this information to unauthorized parties. At the same time, improved technology can be seen as a double edged sword. People with ill-intent are constantly looking for loopholes to exploit them for benefits. As a result, using just a single login mechanism (i.e. PIN/credentials) is no longer considered secure. Therefore, the demand and use of 2FA/MFA for authentication is increasing. In this paper, the use of wearable devices as a form of authentication device will be covered.

Keywords: 2 Factor Authentication (2FA), Multi-Factor Authentication (MFA), RSA, One-Time passcode, Near-Field Communication (NFC), Wearable Computing

1 Introduction

1.1 The idea of 2FA/MFA

Two-Factor authentication (2FA) is a protocol where users are required to use 2 different factors to perform an authentication process. Authentication types are classified mainly under 3 different factors. The first factor is something that the user knows which a typical example would be passwords. The second factor is something that the user has which can be a physical device such as a card or token. The last factor refers to something that the user is, such as the fingerprint of the user.

Multi factor authentication is simply just using a combination of these 3 different factors to perform 1 single authentication.

However, the use of something that belongs to the same type of factor is not considered as 2FA/MFA. An example is that the user knows both the username and password, but this is not known as 2FA because username and password belong to the same factor which is something the user knows. In contrast, a 2FA authentication process will involve the use of a token issued by the bank as well as the username and password that the user knows to access the internet banking website.

1.2 The importance of 2FA/MFA

Users tend to use the same password across different systems such as Hotmail, Gmail and Yahoo mail. As a result, hackers just need to obtain one single password by any means, and he/she will be able to easily access all 3 different email accounts.

Some users have the habit of writing down or save their password elsewhere. Some examples are letting the browser remember the user password, saving passwords in a plaintext file, simply write it down on a piece of paper. A situation where the password can be compromise in an office environment is when a passer-by (perhaps a delivery man) practices social engineering and takes a look at the paper lying around the workstation to discover the user’s password. Another scenario could involve malicious plugins installed in the user’s browser that exploits a loophole to obtain all of the user’s saved passwords.

Most users changed their password only when necessary. According to a survey done by PayPal [6], 48% of the respondents changed this password only when required. If we were to sum up the 7% of the respondents that does not change their password, we actually have more than half of the respondents not changing their password if no policy is enforced. If a password is not changed, an attacker that manages to get a user password can have access to his/her account forever. Below is a chart taken from PayPal:

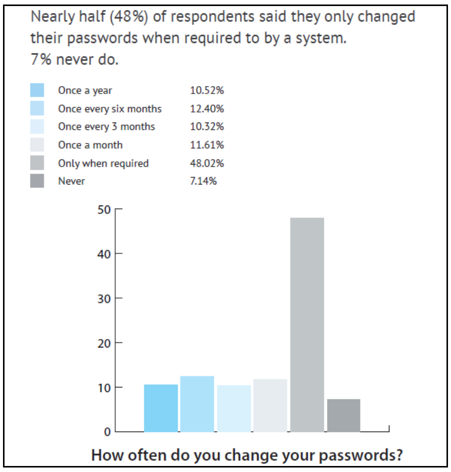


Figure 1: Survey result on PayPal regarding the frequency on changing passwords

From the examples explained previously, we have seen the use of just 1 factor of authentication where it only involves the user's username and password. Given enough time and resources, hackers can obtain the user's password quite easily. However, if at least 2 different factor of authentication are used, it becomes much tougher for hackers to break in.

Here is an illustration of combining the username and password with a physical token: Although an attacker might be able to retrieve the username and password of the victim in a few seconds (peeping at the password written on a notepad lying on the victim desk), the attacker is still unable to gain any access to the victim's account even if he/she does not change his/her password for a long period of time. This is because; the attacker will require the one-time passcode that was produced by the physical token at the point of request in order to gain access to the system.

Although every factor has its ways of being exploited, the result/ways of exploiting are usually isolated from each other. Using the same example above, the attacker can use the mean of peeping/brute force methods to get the username or password. However, the attacker can't use peeping/brute force methods to get the physical device unless the attacker steals it from the victim. This clearly shows that the loss of one factor does not have any direct link with another factor.

1.3 Current forms of 2FA/MFA

1.3.1 ATM card and User Personal Pin

This is a 2FA that involve something that the user has (ATM card) and something that the user know (Personal Pin). Without either one of them, the attacker will not be able to perform any transaction. Furthermore, if the attacker were to get hold of the ATM card and goes to an ATM machine to withdraw cash, the chance of the card being retained by the machine is relatively high as there is a maximum of 3 tries to enter pin. Once the 3rd try failed, the card will be retained by the machine. Therefore, the second factor (Personal Pin) effectively secures the user's bank account.

1.3.2 RSA token and User credential

This is another 2FA process that involves something that the user has and knows. This method of authentication is commonly used by banks to allow customers to perform internet banking. Customer can request for the one-time passcode to be displayed by pressing the button on the token. Using this one-time passcode together with the customer's username and password, he/she will be able to gain access to their bank account over the internet. Again, knowing user credential alone is not sufficient. [3]

1.3.3 Smartcard and User credential

This is yet another 2FA that involves something that the user has and knows. This method is commonly seen by people working in government sectors especially in the military. Anyone that wishes to access a computer connecting to military network will require a valid smartcard to be inserted into a card reader connect to the computer in order to activate it. The second factor would be keying in their user credential.

1.3.4 SMS for electronic transaction

Once again, this is a 2FA that involve something the user has and know. This method can be seen by people that do online shopping. One popular online shop that uses this method is *Qoo10*. When the buyer is making payment, a one-time passcode will be sent to the user registered phone number via SMS, from the credit card issuing bank. The buyer will then key in the one-time passcode through the bank's portal and will be verified and redirected to complete the transaction (The buyer will have to be logged into *Qoo10* before he/she can come to the transaction page).

2 Proposal: Wearable Computing

2.1 History of Wearable Computing

Wearable computers (a.k.a body-borne computers) are electronic devices, usually small in natures that are worn by users under, with or on top of clothing. The idea of a wearable computer is a device that augments human capability by wearing it. As such, the first "wearable" devices were eyeglasses and pocket watches which augment the human capability to see and track time. However, as both 'wearable' and 'computer' are broad terms, it depends on how broadly one defines it. In our case here, we are concerned with computers that are user-programmable items for complex algorithms, interfacing and data management that can be worn by the user.

As such, a significant contributor to the world of wearable computing is Dr. Steve Mann whom himself can be considered as the pioneer of wearable computing. He has developed many wearable devices and has been wearing them since the 1970s. It is because of the vast amount of wearable computers that he wears that he is given the nickname "cyborg". More information about him can be readily found online.

Wearable computing *is the study or practice of inventing, designing, building, or using miniature body-borne computational and sensory devices. Wearable computers may be worn under, over, or in clothing, or may also be themselves clothes (i.e. "Smart Clothing" (Mann, 1996a)).*

Properties of wearable computers [5]

- Unobtrusive
  - Mobile, small, lightweight, no wires
  - Body-wearable (sometimes in clothing)
- Supporting a primary (work) task
  - Don't disturb, be useful all the time
- Casual use, context-aware, "'smart"'

- Consistency (no need to on-off, constant reaction between user and device)
- Extension of user's mind and/or body

2.2 Upcoming trends of Wearable Computing

2.2.1 Pebble Watch

Pebble is an infinitely customisable watch that connects to iPhone and Android smartphones using Bluetooth technology which alerts the user with a silent vibration to incoming calls, emails and messages. Applications for the Pebble are customisable fully based on the user needs. Examples of possible applications includes an app for bikers which accesses the GPS on the smartphone to display speed, distance and pace data or a music control app to play, pause or skip tracks on the smart phone with the touch of a button. The Pebble watch aims to create a minimalist yet fashionable product that seamlessly blend into everyday life. Each Pebble is currently priced at US$150.

2.2.2 Google Glasses [9]

Google Glass is a lightweight, camera-equipped wearable computer in the form of a pair of glasses where it can supposedly capture videos, photos and gives wearer an augmented reality view of the surroundings which it can layer information over. Details of the product's release, full features and specifications have not been released yet to the public. Each Google Glass (Limited Alpha) is currently priced at US$1,500.

2.2.3 Others

- Olympus MEG4.0: http://crave.cnet.co.uk/gadgets/google-glass-high-tech-specs-get-a-rival-in-olympus-meg4-0-50008530/
- WIMM Android Watch: http://www.wimm.com/platform-technology.html

2.3 Differences between Wearable Computers and desktop or mobile devices

2.3.1 Size

Wearable Computers are generally smallest compared to desktop or mobile devices due to be lightweight for the user to wear.

2.3.2 Battery life

Wearable Computers uses less battery compared to desktop or mobile devices mostly due the size of the display screen. Furthermore, due to the overall size of the device, the capacity of the battery is also lesser compared to mobile devices.

2.3.3 Portability

They are more portable than mobile devices. They are generally small, lightweight and can be worn on the body compared to mobile devices which are mostly hand-held and kept in pockets or bags.

2.3.4 Technical Specifications

Wearable devices are almost comparable to the current age mobile devices. As our technology progresses, the size of devices shrinks while processing power increases. It is possible to include GPS, Accelerometer, Bluetooth, WiFi and internet capability.

2.4 Protocol Design

2.4.1 Description

The general authentication idea is based on RSA SecurID [4]. Basically, the service providers (Banks, Social networking platforms, online games, etc.) and users each hold the random seed to generate the one-time passcode at a given minute. However, instead of carrying RSA tokens, users will now have these tokens embedded into their wearable devices. Instead of having a central authority to distribute and manage these tokens, the responsibility is now split among different parties.

Firstly, service providers will not be tied to the central authority's software to add new tokens or authenticate users. They are free to build their own software or simply contribute to an open source one.

Secondly, users will be responsible for adding new services that they wish to enable Multi-Factor Authentication (MFA). Traditionally, users are given a token and told the specific systems that it is used for. However, our protocol design will enable users to add multiple services of their choice. Furthermore, instead of typing in the passcode by hand by looking at the RSA token, users will now simply tap the device to transfer the one-time passcode. This process is done through Near-Field Communications (NFC) [8]. The protocol is also designed to allow multiple devices.

Thirdly, various IT gadgets makers will take charge of designing and producing the physical wearable device.

2.4.2 Technologies and Terminologies used

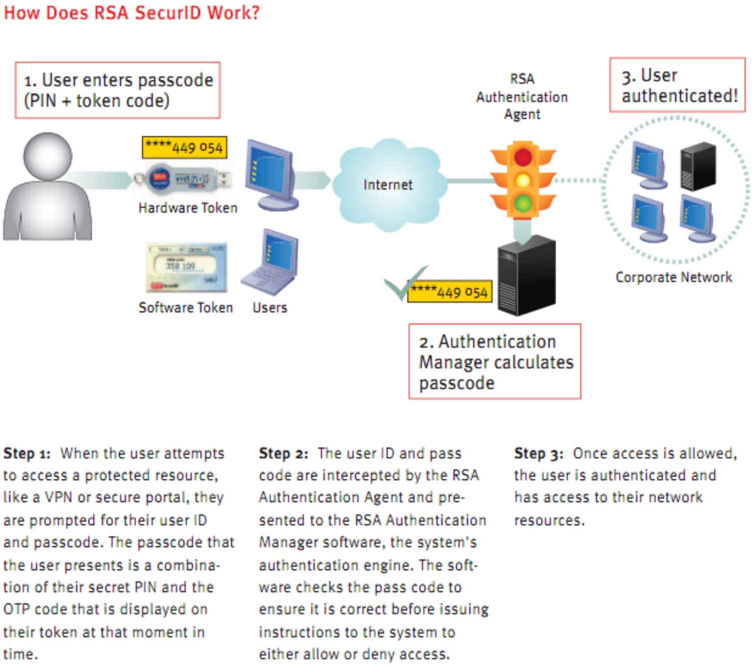
- Device RandomSeed
  - Used to generate One-Time passcodes
- One-Time passcodes
  - Random passcodes appended to the credentials of the user for authentication.
- Service Provider
  - Provides a certain service to user that requires authentication.
  - E-banking, Facebook, Gmail, etc.
  - Users[]
    - Typical User database to store usernames, hashed passwords, etc.

            - Contains MinimumDeviceCount
                - Used to determine the minimum number of devices that are used as MFA during authentication.
            - Devices[]
                - Stores devices information of users (Device RandomSeed and Device Name)
    - ServiceName
        - Identifier of the Service Provider
    - Service MasterSeed
        - MasterSeed of Service Provider used to generate RandomSeeds for User Devices
    - Cert
        - A certificate given by the Certificate Authority such as Versign.
        - It is required to ensure that only the ServiceProvider can request for their own One-Time Passcodes
        - A dummy cert is used in the demo, which is basically "[ServiceName].com"
- User Device
    - Any wearable device capable of using the protocol.
    - DeviceName
        - Name or ID to identify device
    - Services[]
        - Contains a list of (ServiceName, Service RandomSeed)
    - Active
        - To allow passcode request only when active is set.
- User Interface
    - User interface from service provider to request users to authenticate.
    - Web interface, mobile apps, etc.
- Near-Field Communications (NFC)
    - Standards for smartphones and similar devices to establish radio communication with each other by touching them together or bringing them into close proximity, usually no more than a few centimetres.
    - Replaces the traditional input mechanism of looking at RSA SecurID tokens and typing in manually.
- One-Time passcode generator
    - Algorithm to generate the one-time passcode.
    - In the prototype, a simple solution is chosen, which is to hash (RandomSeed + Time in minutes).
    - The timeframe for passcode is set to 1 minute in the prototype.
- Hash algorithm
    - Used to generate One-Time passcodes and to hash passwords.
    - SHA-256 is used in the prototype [1].
    - Other hashing algorithms can be used in actual implementation.

2.4.3 Assumptions

- User devices are NFC-enabled.
- User devices have an input mechanism to manage the services' information.
- User devices have enough computing power for computing hash values, etc.
- Separate input device for users to input login credentials and One-Time passcodes.
- HTTPS used to ensure that only the ServiceProvider can request for their own One-Time Passcode

2.4.4 Protocol Use Case Diagrams



Step 1: When the user attempts to access a protected resource, like a VPN or secure portal, they are prompted for their user ID and passcode. The passcode that the user presents is a combination of their secret PIN and the OTP code that is displayed on their token at that moment in time.

Step 2: The user ID and pass code are intercepted by the RSA Authentication Agent and presented to the RSA Authentication Manager software, the system's authentication engine. The software checks the pass code to ensure it is correct before issuing instructions to the system to either allow or deny access.

Step 3: Once access is allowed, the user is authenticated and has access to their network resources.

Figure 2: Traditional RSA SecurID [10]

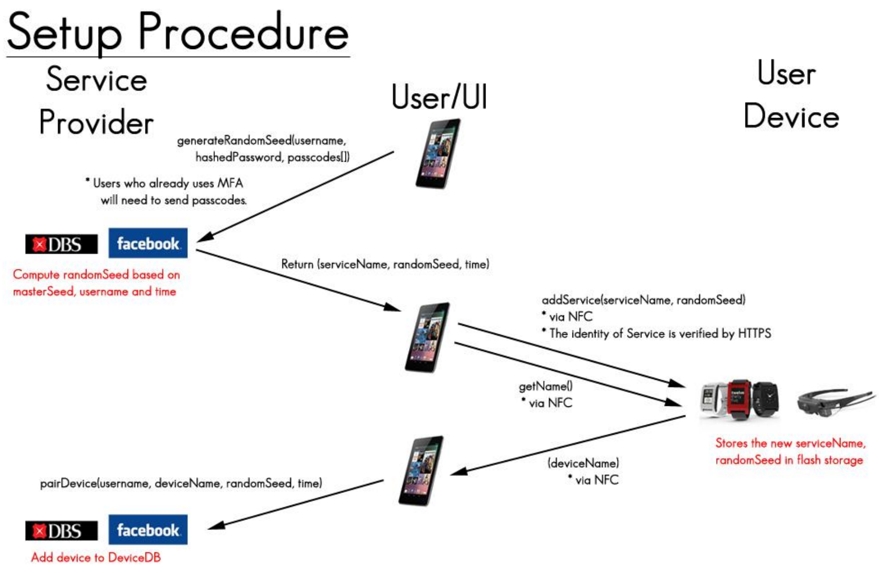

Setup Procedure
Service Provider
User/UI
User Device
generateRandomSeed(username, hashedPassword, passcodes[])
* Users who already uses MFA will need to send passcodes.
DBS
facebook
Compute randomSeed based on masterSeed, username and time
Return (serviceName, randomSeed, time)
addService(serviceName, randomSeed)
* via NFC
* The identity of Service is verified by HTTPS
getName()
* via NFC
Stores the new serviceName, randomSeed in flash storage
(deviceName)
* via NFC
pairDevice(username, deviceName, randomSeed, time)
DBS
facebook
Add device to DeviceDB

Figure 3: Setup/Initialization phrase

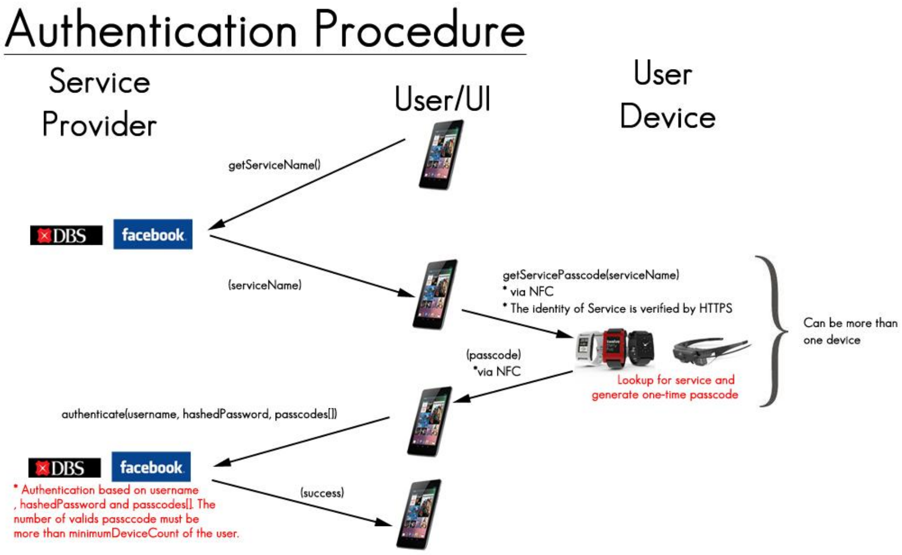

Authentication Procedure
Service Provider
User/UI
User Device
getServiceName()
DBS
facebook
(serviceName)
getServicePasscode(serviceName)
* via NFC
* The identity of Service is verified by HTTPS
Can be more than one device
(passcode)
*via NFC
Lookup for service and generate one-time passcode
authenticate(username, hashedPassword, passcodes[])
DBS
facebook
* Authentication based on username , hashedPassword and passcodes[]. The number of valids passccode must be more than minimumDeviceCount of the user.
(success)

Figure 4: Authentication

Other characteristics:

- Device Active bit
    - If device is set to inactive, it will not respond to getServicePasscode() requests. This prevents cases where hackers attempt to obtain the passcode by getting close to the user.
- Minimum Devices count
    - Forces a minimum number of factors (excluding password) for authentication. However, it must not be more than the total number of devices that the user has chosen as MFA.

3 Evaluation

3.1 User Perspective

3.1.1 Value added devices

Wearable device provide various different form of value-added services such as displaying time, reading SMS as well as email as compare to just 1 RSA tokens that only give you the one-time passcodes only. Furthermore, with so many different function available on the wearable device, user are more willing to purchase as it is similar to purchasing a watch together with a RSA tokens at the price of a watch. According to DBS bank, a fee of S$20 will be imposed for loss/replacement of a RSA tokens. [2]

3.1.2 Extra security with 2FA/MFA

If a customer loses the watch/glasses and hackers were to found it, the hacker will not be able to perform any action. Firstly, he/she will still require the user's second factor, which could be the username and password. Secondly, if other devices (watch/glasses) were added by the user and he/she had specified a minimum device count of 2, the hacker will then require both devices. As a result, it is much more secure to use a combination of these devices for various online services.

3.1.3 Ease of use

One of the key reasons to reject 2FA/MFM is due to the additional work required to authenticate. However, with wearable device, authentication is simple as illustrated in the protocol section. When it comes to checking of time, taking a look at the watch is definitely faster than taking out a mobile phone and tapping on the power button to display the time. This also demonstrates that wearable devices can be more instant and thus more useable - there is also flexibility to use 1 out of 2 valid devices to authenticate in cases where users might have both devices at the point of authentication. NFC has made it easier to transfer the one-time passcode generated as compared to viewing and typing it manually.

3.1.4 Single set of protocol across different platform

If a user has various internet banking account with different bank, he/she will have to possess the same number of RSA token to the number of account he/she have. As a result, misplacing RSA token will definitely be seen as a negative point. However, with just one set of wearable devices, this problem will be solved and it increases the usability of these devices.

3.2 Service Provider Perspective

3.2.1 Fraud Prevention

Comparing to just using a set of username and password (single factor), hackers' success rate to exploit the system is likely to decrease. If a fraud happens, the losses can involve huge amount of money in the case of banking. Furthermore, it deters the public from trusting the bank. With the protocol we have described, fraud is less likely to happen. Thus, the expense to fraud losses reduces.

According to DBS, the cost of making the token will be absorbed by the bank as the first token will be given to the customer for free. However with wearable devices, the cost of implementation will be shift to the consumer as the consumer will be more likely to make their purchase of wearable device. The cost of maintaining SMS passcode services can also be saved as such services are no longer required with wearable devices.

3.2.2 Business Opportunity

If a customer is out for a short trip to the gym, he/she is likely to bring little cash. Thus, they might decide not to purchase a drink from the café. With pebble watch, cash withdrawal is made possible anytime as long as there is an ATM machine or a payment device that supports the protocol.

3.3 IT Gadgets Maker Perspective

3.3.1 Business Opportunity

Since customers benefit from the improved security feature, they are likely to be more willing to purchase these wearable devices. Therefore, it would eventually lead to increase revenue for IT gadgets maker.

3.3.2 Room for innovation

A RSA token or smart-card usually have a fix design and are usually kept in pocket. As a result, the physically design of it is not crucial. However with a wearable device, customer will pay more attention to it as others are looking at them when they are wearing it. Thus, the physical design of the gadget can be made appealing to attract sales. Hence, it will increase not only the chance of innovation for the IT gadgets maker; revenue will also increase.

3.4 Security Experts/Field Perspective

3.4.1 Wide range of 2FA/MFA in action

2FA/MFA usages can be ubiquitous. With 2FA/MFA being made simpler and more appealing to be used with wearable devices, more people will make the switch. Hence, the digital world will be much more secure with more users playing a part for the security of their own information. Furthermore, more jobs can be created to meet this demand.

3.4.2 Improving the fields

With more usage in this field of security, experts will be more interested to contribute. Thus, flaws will be discovered earlier and resolved faster. Research and development are also expected to increase, to improve these technologies.

3.5 Reflections on Security Design Principles

3.5.1 Economy of Mechanism

The communications overhead of the authentication protocol is relatively simple. For a typical authentication request, it takes only 3 Round Time-Trip (RTT) as seen in the previous section.

3.5.2 Fail-Safe Defaults

By default, each new device added is expected to be used for the authentication process. i.e. When a user adds a new device, the minimumDeviceCount is automatically incremented. The user can then decrease the count if he wishes to after adding the device. This way,

3.5.3 Complete Mediation

Not applicable. Whether a service provider wishes to achieve completion mediation is independent of our design.

3.5.4 Open Design

Diagrams and illustration of our protocol is freely available. The source code for a scaled down demonstration is also available on Github. More details can be found in the user guide for the demo. However, technologies that our protocol uses, such as hashing algorithms should also be completely open to achieve a truely open design.

3.5.5 Separation of Privilege

The device(s) act as 2FA/MFA and thus losing one device (or user’s password) will not compromise the user’s account.

3.5.6 Least Privilege

In our design, HTTPS is used to enforce that only service A can obtain/request for service A’s One-Time Passcode. Thus, services are not allowed to obtain another service’s One-Time Passcode.

3.5.7 Least Common Mechanism

The same master seed is used for all users to generate the device's seed. If the master seed is stolen, it will be much easier to crack and obtain every user's devices' seed. On the other hand, users will probably use the same device for multiple services. The device thus becomes a common mechanism to be attacked.

3.5.8 Psychological Acceptability

Interface to the devices are made easy with the use of NFC. Moreover, with the option to set a minimum device count, users are likely to adopt 2FA/MFA as it is now more flexible.

4 Concerns

4.1 Device

4.1.1 Battery Life

Due to the small size of the wearable devices in nature, the battery capacity would be limited. This is further subjected to the usage intensity of the user with regards to the other functions of the device. It would be an inconvenience to the user in an event the battery runs out when the user require the authentication token. Thus, we suggest that a separate battery is used to power the device for One-Time passcode generation. This way, even if the device's main battery is depleted, the components responsible for authentication will still be able to operate.

4.1.2 Stability

As ATM cards, RSA tokens are extremely stable, it is expected that the wearable devices are equally stable for usage. The devices will have to be well designed to handle new functions that are required in our protocol design, such as handling multiple services. Furthermore, if the protocol becomes widely adopted for more types of services, the usage will definitely increase and the device must not wear out easily.

4.1.3 User Input device (UI) vs. User Wearable Device

Some wearable devices come with built in browsers that are capable of accessing internet content. However, because we are using NFC as an interface for passcode authentication, the device is not able to login using the generated token by itself. (i.e. A watch's NFC cannot "tap" to itself). One way to address this concern is to add in a functionality to allow the device to extract the passcode to the browser to authenticate with the web page. However, that may introduce additional security vulnerabilities that may only be fixed from the server side.

4.2 Technical

4.2.1 Technologies Used

Our authentication method is heavily dependent on NFC and the NFC interface. There is already existing exploitation on mobile devices through the NFC interface which raises the issue on how secure are the passcode stored on the wearable device. One solution is to activate NFC only when required so that it is not running passively such that it will accept any foreign NFC connections. That means there can be a button on the device such that the user can conveniently press it when NFC is required and the NFC will be activated for a short period.

4.2.2 Initialization process must secure

To enable the wearable device to generate authentication tokens for different services, the service side needs to communicate over a randomSeed to the device so that the resulting tokens generated by from the seed tally with the table from the companies' side. This is typically done so by secure channels in the browser. However, in an event that the security of the web page is compromised, the authenticity of the randomSeed will be lost. Thus, a more secure way of communicating the seed over is to send via mail much like how banks mail over the credit card to the clients. The setback would be that some companies are not based in the country the user is at and hence the time taken for the initialization process would lengthen.

4.2.3 Security of Devices

The devices used must be reasonably secured so as to prevent unauthorized control of device which would then expose all the passcode to the unauthorized user. Such an attack can be done by exploiting on vulnerability in the device's firmware and can usually be done remotely without the user realizing until it is too late. Some example of such attacks are remote control of iOS * and ** with the latter being more relevant due to the exploit coming from NFC.

(*) http://www.theregister.co.uk/2007/07/24/iphone_security_vulnerability/

(**) http://www.androidauthority.com/hackers-can-exploit-nfc-chrome-browser-to-take-over-your-android-phone-104130/

4.3 Attacks

4.3.1 DDOS

Hackers might steal the devices physically from the user instead of using conventional methods. Although this would not compromise the user's accounts, it poses as an inconvenience because firstly, a device of value is lost and the user would not be able to login to services without the device's authentication factor. In an attempt to remedy this issue, we have incorporated a feature which allows users to set a minimum device count so that only some but not all devices are required for authentication.

4.3.2 Brute Force

Technically, it is still possible to brute force the One-Time passcodes. Without in contact with the wearable device, hackers can still attempt to authenticate by continuously trying different passcodes. To reduce the accessibility of such attacks, service providers should limit the number of authentication attempts every minute.

4.4 Society

4.4.1 Increasing Thief Rate

Associating authenticating tokens into wearable devices could potentially increase chance of such devices to be stolen due to their increase in worth. In an event where such a device is unfortunately stolen from the user, the user can directly report a loss to service providers and the device would be disabled as an authentication device. Alternatively, it is also possible for gadget makers to implement a system to blacklist devices so that service providers will react to disable the device. Another way is to initiate a locking program that disables the entire device.

5 Summary

As our daily lives get more and more involved with the Internet, security becomes extremely vital for the Internet to remain trusted. Although 2FA/MFA solutions are already in the market and in use, it is still not popular and not used as the primary mode of authentication in many services. By using wearable computing with 2FA/MFA, the adoption of 2FA/MFA can become widespread. With our protocol design, the entire 2FA/MFA system can become more usable for users and flexible different services to adopt the system for authentication. We have discussed the key advantage of using wearable devices, such as higher security, ease of use and business opportunities. Nonetheless, several challenges, such as battery life and security of devices are also listed. These concerns should be tackled during the implementation of the system. Otherwise, the system will benefit everyone involved in the Internet world.

6 References

1. Bichlmeier, C. (2008, Feburary 16). *SHA256 - Javascript*.
2. DBS. (2009). *DBS iB Secure FAQ - iBanking | DBS Singapore*. Retrieved from http://www.dbs.com.sg/personal/ibanking/additionalinfo/faq/2fa/default.page?
3. DBS. (2009). *New Generation iBanking Secure Device - An Introduction | DBS Singapore*. Retrieved from http://www.dbs.com.sg/personal/ibanking/newtoken/default.page?
4. EMC Corporation. (2012). *RSA SecurID - Two Factor Authentication, Security Token - EMC*. Retrieved from http://singapore.emc.com/security/rsa-securid.htm
5. Kenn, H. (2007, November 05). *Wearable Computing - Vorlesung im Wintersemester 2007/2008.* Retrieved from http://www.cubeos.org/lectures/W/ln_2.pdf
6. MacGibbon, A., & Phair, N. (2011, September). *Paypal Press Center.* Retrieved from https://www.paypal-media.com/assets/pdf/fact_sheet/cis_paypal_whitepaper_final.pdf
7. Mann, S. (2012). *Wearable Computing*. Retrieved from http://www.interaction-design.org/encyclopedia/wearable_computing.html
8. NearFieldCommunication.org. (2012). *Near Field Communication*. Retrieved from http://www.nearfieldcommunication.org/
9. Trenholm , R. (2012, June 27). *Google Glass Explorer Edition high-tech specs cost £1,000 | CNET UK*. Retrieved from http://crave.cnet.co.uk/gadgets/google-glass-explorer-edition-high-tech-specs-cost-1000-50008422/
10. Whitelegg, D. (2011). *IT Security Expert: March 2011*. Retrieved from http://blog.itsecurityexpert.co.uk/2011_03_01_archive.html