International Journal of Cyber-Security and Digital Forensics (IJCSDF) 5(2): 83-93
The Society of Digital Information and Wireless Communications, 2016 (ISSN: 2305-0012)

Performance Measures for Evaluating the Dynamic Authentication Techniques

Jing-Chiou Liou
Deaprtment of Computer Science,
Kean University
1000 Morris Ave.
Union, NJ 07083, USA
iliou@kean.edu

ABSTRACT

Authentication is the process of verifying users' credentials when access to secure IT systems. Today, most of the computer users rely on single-factor static authentication. However, many severe security breaches in each year prompt us that we need to develop a better authentication mechanism. One solution is to adopt the multi-factor dynamic authentication. Nevertheless, it is financially not possible to implement multi-factor, even just two-factor, authentication for all the computer users. In this paper, we will study currently available dynamic authentication techniques and evaluate their strength and weakness with both feasibility and security measures. We propose eight feasibility measures that can be categorized into two groups: cost and deployment. In addition, we also use five security measures to analyze their capability against security attacks. The comparison indicates there is no perfect solution to the authentication. However, software tokens, some of graphic user authentication (GUA) and the SiFaDA score highly in most categories.

KEYWORDS

Encryption, Multi–Factor Authentication, Dynamic Authentication, One-Time Password

1 INTRODUCTION

With advance in mobile technology and cloud computing, computers today have emerged and changed everything around the world. To most of us, it is becoming absolutely necessary to use technology in our daily lives. Any information can reach any part of the world any time wherever computers and the Internet are available. Computers take communication beyond the definition of communication. With the use of computers, anybody can communicate immediately with anyone around the world.

Through technological advancement, information is currently shared and accessed over millions of servers without boundaries. Even though computers are augmenting our daily lives, they require certain measures on access control and user authentication to assure the security. Authentication is the process of verifying a user's credentials when they are requesting services from any secure system.

A simple single-factor authentication only involves a username and password and this can be easily deciphered. Adding an extra strong factor will greatly reduce the chances of the user's identification from being hacked. For the second factor, there are many techniques available today. However, among billions of computers and the Internet users [1], due to deployment complexity, multi-factor authentication is only utilized in some close, controllable environment setting. These setting include those used by the employer in a company, suppliers in a supply chain, member with paid/profitable membership (e.g., Amazon Web Services AWS. Dropbox, PayPal, etc.).

Therefore, considering the human psychological factor and deployment complexity, we study all currently available dynamic authentication techniques and evaluate their strength and weakness against both feasible and secure measures. In this paper, we will propose, in Section 2, evaluation measures for authentication techniques. Then we will study in section 3 the dynamic authentication methods that are available today. In section 4, we will review dynamic authentication technologies that use virtual One-Time-Password (OTP). All of the authentication techniques will be assessed using the feasibility and security measures. Finally, in section 5, we conclude our discussion and project on possible future works.

2 AUTHENTICATION EVALUATION MEASURES

Authentication is the process of verifying users' identities when they are requesting services from any secure system. During the authentication process, several validation factors may be needed for verification of the client's identity. An authentication factor is a portion of information that is given by the client and used to verify identity the client who is applying for access under certain security constraints. The authentication factor is usually one of three techniques: "proof by knowledge" (e.g., username/password), "proof by possession" (smartcard or token), or "proof by property" (fingerprint scan).

In this section, we propose two sets of measures to evaluate the authentication techniques. The first set exams the feasibility with seven measures. The second set assesses the security performance against four types of attacks

2.1 Feasibility Measures

There exist seven feasibility measures that can be categorized into two sub-groups: cost and deployment. Each of these seven measures may appear in both categories based on their specific requirements.

- **Hardware requirement**: This measure identifies the hardware cost for both the server and the users.
- **Deployment Complexity:** This measure specifies how difficult it is to deploy the technique.
- **Portability:** This is the measure that indicates how easy for users to use the particular scheme in different devices, either private or public.
- **Identity backup:** This measure shows how difficult to get the identity recovered if stolen or lost.
- **Lost Recovery:** This measure indicates the efficiency of recovering the credential, once lost. Single-factor has the best lost recovery, so this measure is primary concerning about the loss of second authentication form.
- **Replacement cost:** This measures the cost of replacing damaged or lost device that is used for authentication process.
- **Multi-Tenant:** This measures the capability of providing a universal mean to be used for multiple online services.
- **Human factor:** This is to measure the human psychological behavior on using the authentication technique. Obviously, the single factor is the most popular with its easy to use and, traditionally, quite get used to it.

2.2 Security Measures

We will compare the four security measures for different authentication techniques. These will demonstrate that we should not use the single-factor authentication as it performs the worst in each of these measures.

- **MitM prevention**: This measure exams how well the authentication scheme preventing man-in-the-middle (MitM) attack. Single factor techniques are more vulnerable to this type of attack.
- **Phishing Prevention**: This measures how easy an attacker can acquire the account/password by masquerading as a trustworthy online service. Most of the OTP techniques will perform strong in this measure.
- **Spoofing Prevention**: This measure designates if the authentication scheme can withstand spoofing attack. The single factor does not achieve high in this measure due to it being incapable of protecting the user's identity from unauthorized parties.
- **Password Cracking**: This measure indicates the impacts of the password cracking on the password file stored on the server when it is suffered from data breach.
- **Shoulder Surfing Prevention**: This is the measure for how the user can avoid losing the credentials caused by shoulder surfing. The single factor performs weakly in this measure due to it being incapable of protecting the user's identity from direct observation.

3 DYNAMIC AUTHENTICATION SCHEMES

Authentication schemes can be categorized into two different classes based on their attributes. Most commonly, they are grouped based on the number of authentication factor. Also, they can be classified by the frequency of passcode change.

When they are grouped by the number of authentication factor, they can either classified as either Single-Factor Authentication (S-FA) or Multi-Factor Authentication (M-FA). In M-FA, the authentication system requires the use of two or more different authentication factors.

If they are defined by the frequency of passcode change, they are referred as either static authentication or dynamic authentication. In static authentication, the passcode is usually not changed until it is required by the security policy or per user's will. For Dynamic authentication, the passcode changes in very short period of time, usually in minutes. Sometime the

passcode changes every time it is generated which is called One-Time-Password (OTP).

S-FA focuses on only one factor: username/password, and is mostly widely accepted technique which is proved to be weak method especially when it comes to protecting data.
In a study by a data security firm [2] that analyzed 32 million passwords exposed in the Rockyou.com breach in December 2009, the top five most common passwords among those 32 million users are: 123456, 12345, 123456789, Password, and iloveyou. In 2013, 4 years later, SplashData's annual "Worst Passwords" list shows again that 123456, password, and 12345678 top all other passwords among 38 million users [3].

Even using secure passwords, phishing and spoofing attacks may use a site that looks like a legitimate one to tricks the user into supplying the password. As a matter of fact, news on October 8, 2009 reported that phishing scheme almost catches FBI Chief [4].

In addition, people usually don't change their passwords frequently. It was reported, in some cases, that less than 25 % of people change their password monthly and some 34% in a survey said they never change their passwords [5]. Therefore, a keystroke logger can be installed physically [6] or in the form of software to catch passwords entered manually on a login screen.

One improvement in S-FA is to utilized password management utility. Password management is achieved by using various password valet applications, such as RoboForm [7] and KeePass **Error! Reference source not found.**, which store user passwords and can automatically enter the required fields in a web form.

Nevertheless, the data is still kept on the host computer or device and can potentially be stolen through browser exploits, Trojan horses, etc..

3.1 Multi-Factor Dynamic Authentication

Multi-factor dynamic authentication requires extra factor(s) other than username/password. Using MFA will increase security, but also will increase difficulty in deployment complexity, hardware requirement and other aspect such as portability, lost recovery, identity backup and replace cost.

The FFIEC issued supplemental guidance on this subject in August 2006 [9] "By definition true multifactor authentication requires the use of solutions from two or more of the three categories of factors. Using multiple solutions from the same category … would not constitute multifactor authentication." To balance the tradeoff between security and deployment issues, the most popular MFA is two-factor authentication. Using two factors as opposed to one factor generally achieves a higher level of authentication assurance.

Smart cards [10] are about the same size as a credit card and require special reader. The downside is that the smart card is not a small device and the card reader is an extra expense. It also needs special middleware application due to the mismatch between smart card communication standards [11] and the communication protocols [12] used by mainstream PC applications.

However, among billions of the Internet users, due to deployment complexity, TFA is only utilized in some close, controllable environment setting. These setting include those used by the employer in a company, suppliers in a supply chain, member with paid /profitable membership (e.g., Amazon Web Services AWS. Dropbox, PayPal, etc.)

3.1.1 Security Token

Security tokens, also called OTP tokens, have an LCD screen that displays fixed number of alphanumeric characters. The OTP tokens are mainly based on two types of algorithms: Time synchronized and event-based.

- Time synchronized algorithm produces a pseudo-random number with a built in pseudo-random number generator. Pseudo-random number changes at pre-determined intervals, usually every 60 seconds.
- Event-based algorithm such as that proposed by the Open Authentication (OATH) consortium [13] uses a user event, such as the user pushing a button on the token.

Figure 1 Security tokens

Some devices, such as RSA SecurityID [14] and VeriSign (now part of Symantec) [15] shown in Figure 1, display 6 digits pseudo-random number and require periodically resynchronize the server with the token.

Taking portability into account, these security tokens must use materials that are small and consume less power. Still, these tokens need to be replaced every few years when the battery is dead. In addition, once the token is lost, the time and cost to replace can frustrate the user due to not being able to access their data. Finally, the security tokens do not prevent Man-in-the-Middle (MitM) based attacks against online transaction along with being unable to defend against malicious users who could use the legitimate user's credentials for authorizing an illegitimate operation as explained in [16].

3.1.2 Virtual Token

Virtual tokens were first introduced in 2005 by a security company, Sestus [17] Virtual token enables any portable storage devices to work as an authenticate token, that's a protected file stored on the device for authentication. Figure 2 displays a Safenet hybrid token [18] that use a USB drive as the security token.

SMS-based T-FA uses user's cellphone as a virtual token. In this type of scheme, when a user enters the first set of credential, a text message is sent to the user's cellphone. This text message contains either a message for approval or an OTP for user to enter into the 2nd factor field of the login screen. A recent development, called "Duo Push: One Tap Authentication," that uses smart phone as the security token by [19] is shown in Figure 6.

Figure 2 Safenet hybrid token

Figure 3 Duo Push: One Tap Authentication

Virtual tokens reduce the costs normally associated with implementation and maintenance of multi-factor solutions by utilizing the user's existing portable storage device. Since the user's portable storage device is communicating directly with the authenticating website, the solution claims to not suffer from man-in-the-middle attacks and other forms of online fraud. However, if not implemented with OTP, can be attacked by phishing and spoofing.

3.1.3 Contactless Token

Contactless tokens form a wireless connection to the client computer that makes them more convenient than both connected and disconnected tokens. Examples of popular contactless tokens are RFID tokens and Bluetooth tokens.

RFID tokens are a comparably new concept in multi-factor authentication. This type of tokens, illustrated in Figure 4, uses RFID tags that store an agent (a small application program) and a pre-defined code (second factor code). The client computer should equip with RFID reader. The need for RFID reader on the client computer significantly increases the hardware requirement.

One example of the RFID token is the RFAA that was firstly introduced in 2011 by Liou, Egan, Patel, and Bhashyam [20]. RFAA token utilizes any RFID devices to work as an authenticate token, that's a protected file stored on the RFID tag for authentication.

Figure 4 RFID reader and tags

With the advance in smartphones and tablets, a new development in contactless tokens is to utilize NFC (Near Field Communication) that is already implemented in mobile devices.

Bluetooth tokens can be used in contact or contactless connection. When the client computer does not equip with Bluetooth, a USB input device is required to plug into the client computer. Thus, this causes uncertainty in hardware requirement.

3.1.4 Software Token

There are two primary architectures for software tokens: Shared secret and public-key cryptography. Shared secret architecture is considered more vulnerable than the hardware token. For both types, the configuration file can be compromised if it is stolen and the token is copied.

As an example shown in Figure 5, RSA SecurID software tokens [14] basically support the same algorithms as their RSA SecurID hardware authenticators. Therefore, like its hardware token, its software token produces either 6 or 8 digits number, called tokencode, and display next tokencode, every 30 or 60 seconds. For online transaction service, it requires, in addition to a web server, RSA Authentication Manager for token provisioning.

The generation of token code is not triggered by the server, but is on client's device(s). User enters the PIN to the installed application, and the client software generates the tokencode. The major concern with such time-based software tokens is that it is possible to borrow an individual's cell phone or laptop, to set the clock forward, and to generate token codes that will be valid in the future. In addition, anyone who provides the PIN correctly can retrieve the tokencode and use it for two-factor authentication on a web server from any cloned devices, such as an SIM card in a cell phone, or a USB installed with such application.

Figure 5 Examples of software token

SofToken is an improved software token technique. SofToken was firstly introduced in 2010 by Liou and Bhashyam [21]. SofToken, rooted on software token, sends not just a pseudo-random number (an OTP), but also the encrypted key to the server for authentication. The technique significantly improves on feasibility and deployment cost of the two-factor authentication.

When the user successfully establishes the user account through online access by providing sufficient information to the service provider; the server delivers client software to the user's computer. This client software installs two components onto user's computer with user's consensus: A logon application and a pseudo-random number generator.

During the initialization process, an encrypted public key will be created and issued to the user's computer as the seed of pseudo-random number generation. The key can be produced based on either a user's favored challenge-response or by the server. This encrypted key will be stored at the user's computer as part of the pseudo-random number generator.

Figure 6 Login scheme for SofToken

Shown in Figure 6, the logon application is directly communicating between the server and user's computer. The logon application requires filling in users credentials that are set up with the server. The user provides the first-factor to the server, username/password. When the server verifies the first-factor, the server sends a request to the pseudo-random number generator installed on the user's computer to trigger the generation of a random number, called code word.

The logon application will provide the user the code word. The user is now able to enter the code word as the second-factor authentication. The code word will be verified again by the server. Depending on the code word, if it is correct the server will grant access to the database otherwise it will close the connection. SofToken acts as second-factor authentication. RFAA is an enhancement process of SofToken. RFAA will required a hardware specification that will be used as Second –factor authentication.

4 VIRTUAL OTP IN DYNAMIC AUTHENTICATION

Dynamic authentication uses cryptography or other techniques to create per-session authentication. A dynamic authentication changes with each authentication session between the claimant and verifier [22].

Dynamic authentication may have different types of protocols, such as challenge-response and virtual OTP. In a challenge-response protocol, the server which runs

the authentication submits a *challenge*, e.g. a random sequence of bytes, to which the client computer responds by computing a cryptographic function which uses both the challenge and a secret data contained in the device.

Protocols using virtual OTP can be viewed as a challenge-response protocol in which the challenge is not sent by the server, but is a publicly known ever changing value. In this case, both the client computer and the server compute the next random credential information based on the first factor. And the two computed random credential information is authenticated at the server side.

4.1 Graphical User Authentication

Graphical password was firstly proposed by G. Blonder [23]. Researchers have latterly developed into several different fashions and now are all referred as Graphical User Authentication (GUA). As the name stands, graphical user authentication uses a graph, instead of text, as the password.

There are two types of Graphical User Authentication techniques based on how they are being used:

- Recall based: The user needs to reproduce something that is created or selected earlier during registration phase.
- Recognition based: A set of images is presented to the user to recognize and identify the images selected earlier during registration phase.

4.1.1 Recall based GUA

This is somewhat very similar to handwriting scheme in the biometrics which may heavily rely on the touch screen on the device. A user will draw something in a way similar to what was drawn during registration. This technique can be further separated into two sub-groups.

Figure 7 Signature technique by Syukri

- Signature scheme [24]: User will hand write on a 2-D lattice the signature shown in Figure 7, or some pre-entered words at certain area of the lattice.
- Draw-A-Secret [**Error! Reference source not found.**]: As shown in Figure 8, user will draw a pattern on a 2-D lattice that correctly matches to the coordination pre-entered.

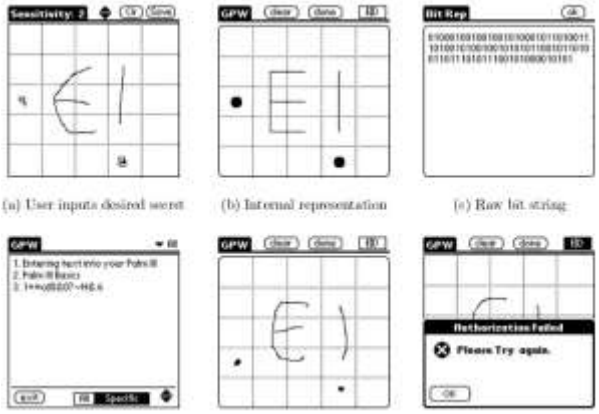

Figure 8 DAS technique by Jermyn

One weakness for this type of GUA having is the shoulder surfing security attack. Since neither technique uses OTP, one can learn other's password by direct observation while the user in entering the password.

Another issue is the user has to correctly re-produce not just the word but also the coordination which may be a challenge to many people. Operate with a mouse to redraw can be also difficult to reproduce correct result. Moreover, this technique requires more storage and process time for authentication.

4.1.1.1 Recognition based GUA

There are quite a few techniques proposed in this type of GUA. One of the disadvantages for recall based technique is the shoulder surfing attack, therefore, many researchers proposed this type of technique to minimize the attack.

Some typical techniques in this category are:

- Dhamija and Perrig [26]: The user will select a number of images out of set of random pictures during registration. Later, during login the user has to identify the pre-selected images for authentication. An example of random images is depicted in Figure 9.

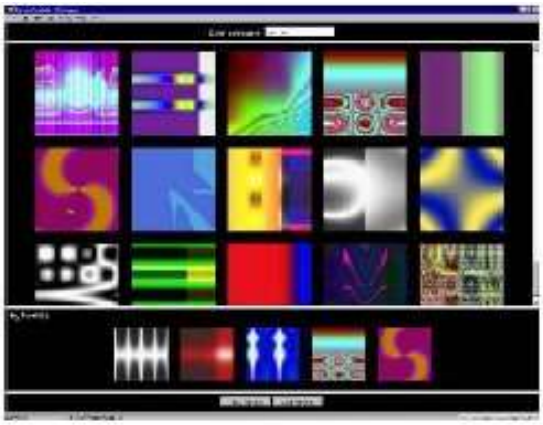

Figure 9 Random images used by Dhamija and Perrig

- Passface [27]: User sees a grid of nine faces and selects one face previously chosen by the user as shown in Figure 10. Here, the user chooses four images of human faces as their password and the users have to select their pass image from eight other decoy images. Since there are four user selected images it is done for four times.

Figure 10 An example of Passfaces

- Shoulder Surfing Resistant: Quite a few recent researches focus on shoulder surfing resistant GUA. Haichang et al [28] proposed a new shoulder-surfing resistant scheme as shown in Figure 11 where the user is required to draw a curve across their password images orderly rather than clicking on them directly. This graphical scheme combines DAS and Story schemes to provide authenticity to the user.
- Graphical user authentication has the strength in all security measures but shoulder surfing. Although some recognition based GUAs claim to reduce the chances for shoulder surfing attack, they significantly increase the time for registration and later to enter correct password. The same drawbacks, as what the recall based GUA has, is the storage and process time for authentication. GUA is strong for feasibility measure, except for lost recovery and human factor.

Figure 11 Haichang's shoulder-surfing technique

To maintain the security level high, the images arrange cannot be too simple. And that complicates the process of memorizing many different passwords. For many people, especially senior citizen, it is convenient for them to use simple password that they can remember. This is the psychological effect that keeps the single factor authentication alive. Hence, graphical user authentication is best used for private device authentication, but not online services.

4.2 SiFaDA VOTP Technique

With advances in computer technology, touch screen computer systems are getting its popularity. And the fact that more and more computer users today use mobile devices for light to medium computing makes a single-factor dynamic authentication a better option as the first line of defense in security.

The single-factor dynamic authentication (*SiFaDA*) was firstly proposed by Liou and Conway [29]. The idea centers in a virtual one-time password process. As the name stands, user just needs to keep hold one factor of

credential. However, when the single factor is entered into the client computer, the client computer will generate, according to a binding code chosen during the registration process, a one-time password and transmit it through the Internet to the server. The server will compute, based on the same binding code, its own one-time password and match it with the one it received. The binding code is actually the virtual second factor for the dynamic authentication.

The VOTP technique starts with a Registration process. Users register for service from secure computer system will download and install an agent program on its private device and receive a binding code. Once the agent program is installed and the binding code is stored on the device, the registration process ends.

The binding process can be applied to use's own private device or a public device, regardless if it is a computer, smartphone, or tablet. For user's private device, the binding code is already stored in the device. Therefore, no extra action is needed for binding the device to the service. The binding code will be used later to create a random keypad.

For a private device, user must firstly enter the username and the agent program will send it to server to request for a random code. Once the client computer receives the encrypted random code, the agent program will use it, along with the binding code, to produce a random pattern keypad as shown in Figure 12.

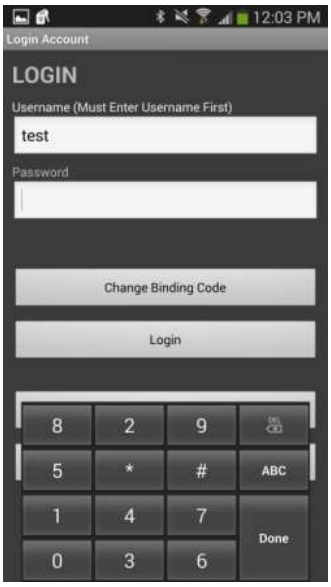

Figure 12: A random pattern keypad on Login screen

If password is 1234, the encrypted and transmitted password will be 7208 for this particular keypad pattern. Since the encrypted random code sent by the server lives one session only, the random pattern keypad will change every time the user log on to the server. Hence, SiFaDA will create a one-time password that is actually transmitted to server; even the entered password is actually the same.

When the server receives the username and VOTP from client device, it will retrieve both the encrypted binding code based on the username and the random number just sent to the client device, and produce its own VOTP. The server then matches its own VOTP with the received VOTP.

For public device, the authentication process is very similar to what is for private device, except that the user needs to enter the binding code first to temporally bind the device for service. The server will generate a random code as it would be sent to the private device. Thus the same random pattern keypad will be produced, according to provided binding code, as it would appear to the private device.

In this case, the server will receive the username, VOTP, and the private binding code at the same time from client device. The server will retrieve user's binding code and match with the one sent by the client device. If they match, the server then matches its own VOTP with the received VOTP. Since the server knows the user is now using a public device, it will force the user to change the private binding code.

5 PERFORMANCE COMPARISON ON DYNAMIC AUTHENTICATION SCHEMES

Based on the proposed two sets of authentication evaluation measures, we have summarized a performance comparison on currently available dynamic authentication schemes. As shown in Table 1, each dynamic authentication scheme is evaluated as "Low," "Medium," or "High" in Feasibility Measure, while in Security Measure, each is assessed as "Weak," "Medium," or "Strong." The best performer is highlighted in bold font style.

5.1 Feasibility Measures

In the set of Feasibility Measures, obviously those schemes with no hardware achieve "Low" as the best performance for "Hardware Requirement" measure. These include Software Token, SofToken, GUA, and SiFaDA. All these schemes use the host computing device as the required hardware and, hence, eliminating the need for extra hardware. That is why the same group of schemes also performs the best in "Replace Cost" measure.

Similar results from previous group appear in the "Lost Recovery" measure, except now the RFAA replace GUA in the best performance. For "Identity Backup"

measure, GUA re-joins the group of best schemes in the group of "Lost Recovery" measure.

For both "Deployment Complexity" and "Human Factor Impact" measures, the group of best schemes is similar to the group of "Lost Recovery" measure, but adding the Virtual Token (SMS) scheme.

"Portability" measure produces a significantly different result that has only Virtual Token (SMS), GUA, and SiFaDA in the group. Meanwhile, taking out of "GUA" from the group becomes the group of best schemes for "Multi-tenant" measure.

With only two schemes, Virtual Token (SMS) and SiFaDA, performing best in the group, it seems the "Multi-tenant" measure is the most difficult feasibility measure for any dynamic authentication schemes to achieve.

5.2 Security Measures

As all of the dynamic authentication schemes are used as the second factor in a multi-factor authentication, most of these schemes perform very well in security measures. Among them, there are four schemes with "Strong" performance in all security measures: Virtual Token (USB), SofToken, RFAA, and SiFaDA.

Security Token suffers a bit from MitM attacks as the pseudorandom algorithm is stationary in the token and adversary can secretly observe the pattern of random numbers to possibly decrypt and predict the sequence of the future random number generation.

Although Virtual Token (SMS) has no such concern in "MitM Prevention" measure, the use of cellphone in such scheme becomes the weakest link in authentication process. The use of cellphone causes the scheme perform very weak in both "Phishing Prevention," and "Spoofing Prevention." In many occasions, people allows love ones and friends to use their phone with no sense of security concerns. People also download and install apps onto their phone without knowing the complete functions of the apps can do on their phones.

Software token performs a little better than the Virtual Token (SMS) does in the two security measures. However, it hurts from MitM attacks with the same cause as what is in the "Security Token." SofToken improves on all three security measures by allowing the users to change the seed for pseudorandom number generation and change the future random number sequence.

GUA authentication scheme is very popular today, especially used in the mobile devices. Even so, it is really not a very secure authentication technique. The main reason is that most of GUA schemes are actually not dynamic authentication. They use pre-selected pattern from a set of images(s) that will not change during every authentication. Some of the GUAs generate a random sequence of the image(s) for users to enter the same pattern. It is weak to against to "Shoulder Surfing" in the public space.

And with limited possible patterns in the set of image(s), it is not strong in preventing from "Password cracking." This is especially true when the GUA scheme uses coordinate positions on an image, because there are just few interesting/significant points in the image. To improve on this issue, user has to use a more complicate image. However, doing so will create "Passwords you'll never forget, but can't recall" issue that was discussed in [30].

6 CONCLUSION

In this paper, we propose the performance measures for evaluating the dynamic authentication schemes on feasibility and security. We then conduct a study on currently available dynamic authentication schemes, and use the two sets of measures to assess these authentication techniques. The performance comparison indicates there is no perfect solution to the authentication. However, SofToken, SMS virtual tokens, SiFaDA and some GUAs (when using dynamic scheme) score better in most categories. This is primary due to the use of OTP or VOTP that attribute not only to its characteristics, but also its ability to maintain a higher level of security for the users. GUA is better for private mobile devices. The SiFaDA is the only single factor authentication that stands out from all schemes and provides a better human-factor effect, multi-tenant on user friendly and security measures.

7 References

1. Internet World Stats. Last retrieved on 5/12/2016. http://www.internetworldstats.com/stats.htm.
2. Imperva Releases Detailed Analysis of 32 Million Breached Consumer Passwords. Last retrieved on 5/12/2016. http://www.imperva.com/ld/password_report.asp
3. SplashData's annual "Worst Passwords" list. Last retrieved on 5/12/2016. http://splashdata.com/press/worstpasswords2013.htm
4. Phishing Scam Spooked FBI Director Off E-Banking. Last retrieved on 5/12/2016. http://voices.washingtonpost.com/securityfix/2009/10/fbi_director_on_internet_banki.html
5. Furnell, S.: Computer Insecurity: Risking the System, pp. 54 – pp.56, Springer, London, UK, (2005).
6. Keysweeper. Last retrieved on 5/12/2016. http://samy.pl/keysweeper/

7. Roboform official site. Last retrieved on 5/12/2016. http://www.roboform.com/index.html
8. Keepass official site. Last retrieved on 5/12/2016. http://keepass.info/
9. FFIEC press release. Last retrieved on 5/12/2016. http://www.ffiec.gov/press/pr081506.htm
10. ISO/IEC 7816-3:1997 "Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 3: Electronic signals and transmission protocols". International Organization for Standards; http://www.iso.org
11. Timothy, M. J., Scott B. G.: Smart Cards: the developer's toolkit. Prentice Hall Professional, Upper Saddle River, NJ (2002).
12. Postel, J.: Internet Protocol. RFC 791, and Transmission Control Protocol, RFC 793 September (1981).
13. Open Authentication Consortium supports event based, and even time based OTP algorithms, http://www.openauthentication.org
14. RSA security http://www.emc.com/security/rsa-securid.htm Safenet. Last retrieved on 5/12/2016.
15. VeriSign. Last retrieved on 5/12/2016. http://www.verisign.com/static/043732.pdf
16. SC Magazine, Web Application Security in Un-trusted Client Scenarios, Last retrieved on 5/12/2016. http://www.scmagazineuk.com/web-application-security-in-un-trusted-client-scenarios/article/110448/
17. Virtual Tag™ multi-factor authentication. Last retrieved on 5/12/2016. https://sestus.international/
18. Safenet. Last retrieved on 5/12/2016. http://www.safenet-inc.com/multi-factor-authentication/
19. Due Push: one authentication. Last retrieved on 5/12/2016. https://www.duo.com/
20. Liou, J.-C., Egan, G., Patel, J. K. Bhashyam, S.: A Sophisticated RFID Application on Multi-Factor Authentication, In: Proc. 8th International Conference on Information Technology: New generation, pp. 180--185, Las Vegas, NV, (2011).
21. Liou, J.-C., Bhashyam, S.: A Feasible and Cost Effective Two-Factor Authentication. In: Proc. 2nd International Conference on Software Engineering and Data Mining (SEDM '10), pp. 47--51, Chengdu, China, (2010).
22. NIST Guide to Selecting Information Technology Security Products (NIST Special Publication 800-36, Oct. 2003)
23. Blonder, G. E.: Patents: Graphical Password. Last retrieved on 5/12/2016. WWW.google.com/patents/US5559961
24. Syukri, A. F., Okamoto, E., Mambo, M.: A User Identification System Using Signature Written with Mouse. In: Third Australasian Conference on Information Security and Privacy (ACISP): Springer-Verlag Lecture Notes in Computer Science (1438), pp. 403--441, (1998).
25. The design and analysis of graphical passwords. Last retrieved on 5/12/2016. https://www.usenix.org/legacy/events/sec99/full_papers/jermyn/jermyn.pdf
26. Dhamija, R., Perrig, A.: Déjà Vu: A User Study Using Images for Authentication. In: 9th USENIX Security Symposium, pp.4-4, (2000).
27. Real User Corporation: Passfaces. Last retrieved on 5/1/2016. www.passfaces.com
28. Gao, H., Ren, Z., Chang, X., Liu, X., Aickelin, U.: A New Graphical Password Scheme Resistant to Shoulder-Surfing. In: Proceedings of the International Conference on CyberWorlds, Singapore, 194--199, (2010).
29. Liou, J.-C., Conway, J.: A Single-Factor Dynamic Authentication for Computer Systems with Touch Screens, In: Proc. 23rd International Conference on Software Engineering and Data Engineering (SEDE '14), Paper ID. 29 in CD-ROM, New Orleans, LA, (2014).
30. Weinshall, D., Kirkpatrick, S.: Passwords you'll never forget, but can't recall, Conference on Human Factor in Computing System 2004 (CHI '04), pp. 1399--1402, (2004).

Table 1 Performance Comparison on Dynamic Authentication Schemes

Performance	**Security Token**	**Virtual Token (USB)**	**Virtual Token (SMS)**	**Software Token**	**SofToken**	**RFAA**	**GUA**	**SiFaDA**
Hardware requirement	Medium	Medium	Medium	**Low**	**Low**	Medium	**Low**	**Low**
Deployment complexity	High	Medium	**Low**	**Low**	**Low**	**Low**	Medium	**Low**
Portability	Medium	Medium	**High**	Medium	Medium	Medium	**High**	**High**
Identity backup	Low	Medium	Medium	**High**	**High**	**High**	**High**	**High**
Lost recovery	Low	Low	Low	**High**	**High**	**High**	Medium	**High**
Replace cost	High	Medium	High	**Low**	**Low**	Medium	**Low**	**Low**
Multi-tenant	Low	Medium	**High**	Medium	Medium	Medium	Low	**High**
Human factor impact	High	High	**Low**	**Low**	**Low**	**Low**	Medium	**Low**
MitM prevention	Medium	**Strong**	**Strong**	Medium	**Strong**	**Strong**	**Strong**	**Strong**
Phishing prevention	**Strong**	**Strong**	Weak	Medium	**Strong**	**Strong**	Medium	**Strong**
Spoofing prevention	**Strong**	**Strong**	Weak	Medium	**Strong**	**Strong**	Medium	**Strong**
Password cracking	**Strong**	**Strong**	**Strong**	**Strong**	**Strong**	**Strong**	Weak/Medium	**Strong**
Shoulder surfing	**Strong**	**Strong**	**Strong**	**Strong**	**Strong**	**Strong**	Weak/medium	**Strong**