Multi-factor Authentication Updating System Evaluation Dynamically for Service Continuity

Hiroya Susuki[1], Rie Shigetomi Yamaguchi[1] and Shizuo Sakamoto[2]
[1]*The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan*
[2]*NEC Corporation, 7-1 Shiba 5-chome, Minato-ku, Tokyo, 108-8001, Japan*

Keywords: Multi-factor Authentication, Bayesian Model, Probabilistic Model, Security, Usability.

Abstract: In response to changes in security environments, an authentication framework has an important role for service continuity, which can evaluate both of security and usability and handle authentication methods. If the service provider cannot respond to problems such as new attacks immediately, the service must be stopped. In this paper, we propose a multi-factor authentication framework using a probabilistic evaluation method considering service continuity. Our framework includes a formal theoretical model, based on Bayesian approach, to be dynamically updated to use appropriate combinations of authentication factors in response to changes in the security environment. The model is important because it forms the basis on which the real-world systems is able to be evaluated security immediately and responded to weak factor.

1 INTRODUCTION

Various services are now available in online environments. These services require that the user be authenticated its access. On most online service sites, only password authentication is used. As a result, passwords are also a major source of vulnerability(Herley, 2009). There are advantages and disadvantages for authentication methods, and there is a trade-off between security and usability. Often, more secure methods are less convenient. An effective way to resolve these problems of single-factor authentication is to use multi-factor authentication(Shay et al., 2010). One such method is Google 2-step verification, which is a well-known method that uses multi-factor authentication to prevent spoofing(Google, 2015). Although authentication systems using multi-factors have become increasingly widely used, multi-factor authentication methods are not secure if the combination of multiple factors itself is not secure.

We also note that new techniques for cracking not only authentication factors but authentication methods are continuously being invented. User authentication systems are unable to rely on frameworks that use only a single factor or a single authentication method. When a service provider needs to change the authentication factor, this cannot be done with a single-factor system. Similarly, systems that use only a single authentication method cannot be changed to the other method. Since continuity of service is very important, there is a need for authentication frameworks that can change which factor and which method they use. However, currently, there are no ways to evaluate these multi-factor systems since service providers need to evaluate both security and usability formally. Basic idea has been already proposed in many articles but these schemes are heuristic and is difficult to present an overview for security on the system. Probabilistic models can estimate the evaluation of systems in an environment without enough statistical data. In addition, for example, most systems using password authentication return a binary value (either *accept* or *deny*). On the other hand, in the case of biometric authentications, such as fingerprints, irises, or faces, the score is not binary but a probability value, such as the likelihood(Damer et al., 2014). The evaluation using probabilistic model is appropriate for handling mixture of probability scores. Therefore, the evaluation of the authentication systems requires the use of probabilistic models.

In summary, the contribution of this work is a probabilistic framework for multi-factor authentication with a probabilistic evaluation model; with this method, authentication factors can be dynamically changed to correspond with the current security environment. Although our formulation is basic, our approach is important because the service providers become easier to manage the authentication system by

Susuki, H., Yamaguchi, R. and Sakamoto, S.
Multi-factor Authentication Updating System Evaluation Dynamically for Service Continuity.
DOI: 10.5220/0005681802730281
In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP 2016)*, pages 273-281
ISBN: 978-989-758-167-0

the formulation of multi-factor authentication framework. In this paper, we show how to use our proposed framework to select a combination of factors that are suitable in terms of both security and usability.

This paper is structured as follows. In the next section, we provide general background about multi-factor authentication and evaluation methods. In Section 3, we describe our proposed probabilistic method and multi-factor authentication model. In Section 4, we provide and analyze an application of the proposed method. Finally, in Section 5, we present our conclusions and discuss areas of future work.

2 RELATED WORK

In this section, we provide the general background of multi-factor authentication and evaluation methods.
Various methods for risk assessment and risk control have been proposed, but not sufficient in actual operation. In recent years, risk-based authentication methods have become practical. There have also been theoretical investigations of risk-based access control models, which consider security risks when making decisions about access control(Karabacak and Sogukpinar, 2005)(Chen and Crampton, 2012)(Cheng et al., 2007). A study has compared multiple authentication methods to the text-based password authentication(Bonneau et al., 2012). Another study assessed the security of the United States e-government sites to identify opportunities for and threats to the sites and their users(Zhao and Zhao, 2010). Many of these approaches consider only the security aspects of authentication systems, whereas usability is also important. These models evaluate the security of the authentication system in each situation and then choose from multiple levels of security. The Electronic Authentication Guideline provided by National Institute of Standards and Technology (NIST) presents the technical requirements for each of the levels of assurance for authentication(Burr et al., 2004). Authentication systems can balance security and usability at each level, prior to making a decision (Kim and Hong, 2011)(Hocking et al., 2010). Existing studies have considered only the current evaluation of the authentication systems and have not considered updating the evaluation, although this is important for continuity of service.

Since the result of an authentication method is not necessarily a binary value, it is important to have an evaluation measure that can deal with fractional results. Therefore, evaluation of authentication systems requires probabilistic models that are based on a forecast and can deal with fractional results. Bayesian models are a popular type of probabilistic model, and even if the amount of sample data is small, the model can estimate the probability by using Bayesian subjective probability, and thus the probabilities can be updated continuously. There have been various relevant studies using Bayesian models(Pavlovic and Meadows, 2010)(Nguyen et al., 2011)(Kondakci, 2010). Thus, there have been methods proposed that use Bayesian models for user authentication. However, these models considered only security, and it is also necessary to consider usability of authentication methods.

Various multi-factor authentication methods have been proposed for improving the security of authentication systems. These methods consist of various combinations of factors and various ways in which they are combined. Two- and three-factor authentication methods have become popular(Yang et al., 2008). One typical example is a multi-biometric authentication method. Multi-biometric methods use multiple sources of biometric information to enhance performance and to overcome the limitations of conventional unimodal biometrics(Damer et al., 2014)(Al-Assam et al., 2010). With the growth in concern about security in the mobile environment, various multi-factor authentication methods have been proposed(Aloul et al., 2009)(Riva et al., 2012)(Sabzevar and Stavrou, 2008). As discussed above, various multi-factor authentication methods have been proposed, but most of them evaluate only the current security environment and do not consider changing the authentication factors. In addition, there has been limited research that considered usability. In the next section, we propose a probabilistic model for multi-factor authentication that considers both security and usability.

3 AUTHENTICATION MODEL

In this section, we begin by defining the multi-factor authentication method that will be considered in this paper, and define the related terms and notation.

3.1 Definition of Terms and Notation

We define authentication as the process of establishing confidence in the identity of a user or information system; multi-factor authentication is based on a combination of two or more factors.

Let u be a user who wishes to be authenticated, and let U be the set of all users; that is, $u \in U$. In order to allow a service *srvc* to be provided to u, an

online transaction tra is performed to establish confidence with the provider that u has legitimate authority to access that service. In addition to the request for a $srvc$, u provides a clear intention. Note that u must be authenticated by the service provider for each transactions tra. In this paper, we do not consider about machine-to-machine authentication. We will assume that the services are authenticated by the intention of u. With one-to-one (1:1) authentication, it is possible to identify the account to which u tried to gain access. On the other hand, with one-to-many (1:n) authentication, the account cannot be identified. In the case of one-to-many authentication using biometrics, u may be authenticated as a different user.

3.1.1 Definition of Function and Scenario

We define two terms for multi-factor authentication. When a user $u \in U$ wishes to be provided a service $srvc_i (i \in \mathbb{N})$, u must participate in multi-factor authentication. Any factors in any combination can be chosen, and it is possible to select different combinations of factors for different services. The particular combination of factors is called a scenario $S \in \mathbb{S}$.

Function. The functions are the processes used in single-factor authentication; these include such things as password authentication and fingerprint authentication. We denote a function as $f \in \mathbb{F}$.

Scenario. A scenario is a combination of functions, which results in a series of authentications. We denote a scenario as $S \in \mathbb{S}$. There are dependencies between some of functions.

Let S be a combination of functions $f_1, f_2, \ldots f_n, (n \in \mathbb{N})$, for which an order is defined, as follows.

$$S = ((f_1,1),(f_2,2),\ldots(f_n,n))(n \in \mathbb{N}) \quad (1)$$

S' is the set of functions without a defined order.

$$S' = ((f_1),(f_2),\ldots(f_n))(n \in \mathbb{N}) \quad (2)$$

Consider a scenario that uses both password authentications with ID and fingerprint authentication. S is composed of two functions: f_1 is password authentication with ID, and f_2 is fingerprint authentication.

Each function f returns a binary value $(0,1)$ for the result of a single-factor authentication. A return of 1 shows that authentication was successful, and 0 shows that it failed.

3.1.2 Service and Scenario

In this paper, we define that a scenario S is composed of one or more functions $f_1, f_2, \ldots f_n (n \in \mathbb{N})$. Furthermore, a scenario S that is composed of only one function f is a single-factor authentication. It is also possible to create a scenario that combines scenarios. As an example, we now consider a scenario in which a user accesses an online service on a website. A user u activates his/her mobile device by using function f_1 (fingerprint authentication). Then, u logs onto a website by using function f_2 (password authentication).

When a user u is provided a service $srvc$ by a service provider $SP \in \mathbb{SP}$, u must participate in authentication in accordance with the required scenario. Let $a \in A$ be an adversary who impersonates a legitimate user u. It is possible that there are multiple scenarios $S_1, S_2, \ldots S_k (k \in \mathbb{N})$ from which the SP can select a suitable scenario for the given $srvc_1$. Thus, if one scenario is cracked and thus loses security, the provider can require a different scenario. In this way, there is a mechanism to ensure that, in a changing environment, continuous service can be provided and safety can be maintained.

3.2 Combinations of Functions

In this subsection, we discuss the relations between functions. There are two types of combinations of functions: sequential and parallel.

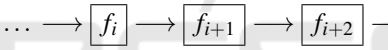

Figure 1: Sequential Functions.

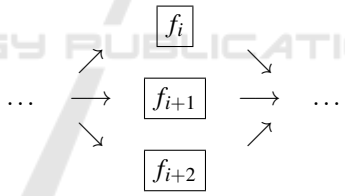

Figure 2: Parallel Functions.

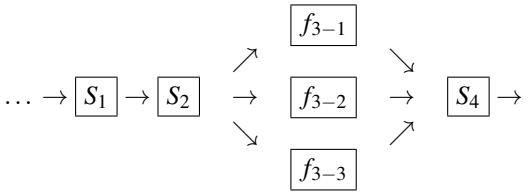

Figure 3: Combination of Sequential and Parallel.

3.2.1 Sequential Functions

Sequential functions result in step-by-step authentication. As shown in the Figure 1, each function is completed before the next is begun. For example, in the first function, f_i, u enters an ID and a password. In the second function, f_{i+1}, u enters additional secret information from a table of random numbers. Further

functions are processed in order until authentication is completed. The order of S is important in the sequential functions. The result of a function determines the result of later functions.

3.2.2 Parallel Functions

Parallel functions are all performed at the same time, as shown in Figure 2. For example, two functions (f_1; password authentication and f_2; device ID authentication) are performed simultaneously. The order of S does not have any meaning and can be changed. Thus, S can be written as follows.

$$S' = ((f_1),(f_2),\dots(f_n))(n \in \mathbb{N}) \tag{3}$$

3.2.3 Combinations of Sequential and Parallel

There are also authentication methods that use both sequential and parallel functions. Figure 3 shows an example of a combination scenario S_5, which is written as follows.

$$\begin{aligned} S_5 &= (S_1, S_2, S_3', S_4) \\ &= (S_1, S_2, (f_{3-1}, f_{3-2}, f_{3-1}), S_4) \end{aligned} \tag{4}$$

3.3 Definition of Probability

We discuss the probability of acceptance authentication for multi-factor authentication.

3.3.1 Probability of Acceptance for Multi-Factor Authentication

When a function $f \in \mathbb{F}$ is accepted, a value of 1 is returned, and when it is denied, 0 is returned. When a user u participates in a multi-factor authentication using scenario $S = (f_1(u),\dots,f_n(u))$ $(n \in \mathbb{N})$, at function f, the conditional probability of acceptance into the user's own account is $P(Acc|f(ID_u,u))$. If an adversary a impersonates a user u, the probability is $P(Acc|f(ID_u,a))$. The other probabilities are shown in Table 1. Similarly, the conditional probability of acceptance for authentication at scenario S for a user's own account ID_u is $P(Acc|S(ID_u,u))$.

Note that the notation $P(x|y)$ expresses the conditional probability of x given the value of y, whereas $P(x|y,z)$ expresses the conditional probability of x given the values of both y and z. These apply when two or more events will happen at the same time.

We propose a multi-factor authentication framework based on a Bayesian model. Bayesian updating of our model is based on fraud detection during authentication. The probability is updated at every login attempt.

Table 1: Probability of the Authentication Function.

Function	Who	Accept /Deny	Probability
$f(ID_u,u)$	u	1	$P(Acc\|f(ID_u,u))$
$f(ID_u,u)$	u	0	$P(Deny\|f(ID_u,u))$
$f(ID_u,a)$	a	1	$P(Acc\|f(ID_u,a))$
$f(ID_u,a)$	a	0	$P(Deny\|f(ID_u,a))$

3.3.2 Probability of Sequential Authentication

When a scenario S consists of a dependent series of sequential functions, we can define the case in which a legitimate user is accepted as follows. Note that, in the case of sequential authentication, when a function fails in the middle, the entire authentication is unsuccessful. Also note that when the events are correlated, their probabilities affect those of the others.

$$\begin{aligned} P(Acc|S(ID_u,u)) = \quad & P(Acc|f_1(ID_u,u)) \times \\ & P(Acc|f_2(ID_u,u,f_1(ID_u,u))) \times \\ & P(Acc|f_3(ID_u,u,f_1(ID_u,u),f_2(ID_u,u))) \\ & \times \cdots \times P(Acc|f_n(ID_u,u,f_1(ID_u,u), \\ & f_2(ID_u,u),\dots,f_n(ID_u,u)) \end{aligned} \tag{5}$$

When an adversary a tries to impersonate u, we have

$$\begin{aligned} P(Acc|S(ID_u,a)) = & P(Acc|f_1(ID_u,a)) \times \\ & P(Acc|f_2(ID_u,a,f_1(ID_u,a))) \times \\ & P(Acc|f_3(ID_u,a,f_1(ID_u,a),f_2(ID_u,a))) \\ & \times \cdots \times P(Acc|f_n(ID_u,a,f_1(ID_u,a), \\ & f_2(ID_u,a),\dots,f_n(ID_u,a)) \end{aligned} \tag{6}$$

3.3.3 Probability of Parallel Authentication

When a scenario S consists of only independent parallel functions, the probability that a legitimate user is accepted is as follows.

$$\begin{aligned} & P(Acc|S(ID_u,u)) = P(Acc|f_1(ID_u,u)) \times \\ & P(Acc|f_2(ID_u,u)) \times \cdots \times P(Acc|f_n(ID_u,u)) \end{aligned} \tag{7}$$

When an adversary a tries to impersonate u, we have

$$\begin{aligned} & P(Acc|S(ID_u,a)) = P(Acc|f_1(ID_u,a)) \times \\ & P(Acc|f_2(ID_u,a)) \times \cdots \times P(Acc|f_n(ID_u,a)) \end{aligned} \tag{8}$$

3.3.4 Probability of Combination

We now consider the probability of a combination of sequential and parallel authentication, as was discussed in Section 3.3.2. We will assume that the parallel authentication does not have a prior probability.

The probability of the generalized sequential authentication is as follows.

$$P(Acc|S(ID_u,u)) = P(Acc|f_1(ID_u,u)) \times P(Acc|f_2(ID_u,u,f_1(ID_u,u))) \times P(Acc|f_3(ID_u,u,f_1(ID_u,u),f_2(ID_u,u))) \times \cdots \times P(Acc|f_n(ID_u,u,f_1(ID_u,u), f_2(ID_u,u),\ldots,f_n(ID_u,u)) \quad (9)$$

4 AUTHENTICATION OPERATION

In this section, we discuss authentication operation using proposed probabilistic model based on a Bayesian model.

4.1 Service and Scenario

When a user $u \in \mathbb{U}$ is provided a service $srvc_i (i \in \mathbb{N})$ by a service provider $SP \in \mathbb{SP}$, u and SP perform multi-factor authentication. If u and SP are able to select several types of single factors as a scenario S, they must decide which types and how many factors they will use for each types of service.

In Section 3, we defined the probability of acceptance for our multi-factor model in which the factors could be changed: $P(Acc|S(ID_u,u)$. To take a simple example, consider the difference between a bank that requires high security when a user wants to transfer a large amount of money, and a coffee shop that requires very little security but high speed when a user wants to buy a cup of coffee. We say that the SP needs to be able to select many security levels and from many scenarios, depending on the immediate need.

4.1.1 Definition of User Authentication

We discuss the relation between scenarios and services. Suppose a SP needs to verify whether someone who is requesting a service $srvc$ is indeed the person registered to receive it; they do this by exchanging various information through the Internet. This procedure is called authentication.

The reason that the SP needs to verify the user is that the service $srvc$ is only open to anyone. The SP must ensure that u pays the amount required in order to be provided $srvc$, that u is a member of a particular organization, or that for some other reason, u has the right to receive that service. Note that the SP provides $srvc$, and u requests $srvc$; but SP does not provide the service if this is not u's intention.

We assume that $srvc$ is provided through the Internet as an online service, such as entry to an e-commerce site, a shopping or auction site, a communication site, or SNS. These kinds of sites do not provide services continuously, but only when u has an intention to receive a service. This means that u requests the service on a regular basis, but only once per visit.

As human, we are not able to connect directly to the Internet, but only through specialized equipment, such as a personal computer or a smartphone. We will assume that this equipment is always able to communicate with the Internet. The equipment is not able to form an intention, and so even when services are provided through such equipment, the user is receiving a direct service.

4.2 Security and Usability

We now define parameters for the security and usability of a service. For every authentication system, security is a first priority when evaluating the system. Because we are considering a multi-factor authentication system, the security evaluation must be based on a combination of several factors, not just a single-factor authentication. It is also important to consider usability, but this must be balanced against security.

4.2.1 Security

In any authentication system, u must be given access, but an adversary $a \in \mathbb{A}$ must fail. That is, the probability that adversaries $a_1,\ldots a_n$ achieve success with function f should satisfy the following.

$$P(Acc|S(ID_u,a)) = 0 \quad (10)$$

However, the more secure system, the higher cost is required. In terms of construction cost of the system, the system that the probability of acceptance for adversaries is zero is not realistic. The cost is required for the system construction as well as operation. SPs need to pay attention the cost.

Let $k(0 < k < 1)$ be a security parameter, where the SP must determine the acceptable level of risk k. That is, SP should build a system that requires a level of security that is not more than k.

$$P(Acc|S(ID_u,a)) < k \quad (11)$$

4.2.2 Usability

If a system denies any of u's requests, that system is not serviceable. The SP must choose the probability that u is denied service. If the SP decides to emphasize only usability and u is able to be access any factors, then the probability that access is denied should

Table 2: Parameters for the Service and the Cost.

Service	Cost	Security parameter	Usability parameter
$srvc_1$	cost_1	k_1	ub_1
$srvc_2$	cost_2	k_2	ub_2
$\vdots$	$\vdots$	$\vdots$	

be as follows.

$$P(Deny|S(ID_u,u)) = 0 \quad (12)$$

However, the input by u may vary if there is user error or if u's biometric measurements fluctuate. It is not realistic to design a system that is based on the assumption that u is always able to input the correct value. Let ub, $0 < ub < 1$, be the parameter for the service availability, as chosen by the *SP*. The *SP* should build a system that requires availability of not more than ub, as follows.

$$P(Deny|S(ID_u,u)) < ub \quad (13)$$

4.2.3 Parameters

Note that neither k nor ub should be unique.

When a person wants to transfer a small amount to an account to which he or she has previously transferred money, the value of k should be low. When a person wants to transfer a large amount to an account to which he or she has never transferred money, the value of k should be high. The service provider should be able to allow for flexible responses.

On the other hand, the *SP* must change the service level depending on the value of ub. In general, although the level of security should be high, excessive security is not always appropriate for a given service. There is a tradeoff between security and usability, and it is important that these be balanced when considering the overall quality of service.

See Table 2 for the parameters that must be set by the *SP*. The *SP* creates the table for cost and parameters before starting the service.

4.3 Advanced Evaluation and Update

We will discuss how to continually and dynamically evaluate the security and usability of a service.

When the *SP* provides a service *svrc*, the *SP* must decide which kinds of scenario $S_1, S_2, \dots$ are suitable for *svrc*. Note that there must be many different types of scenario, and s must be framed by several types of functions $f_1, f_2, \dots$. If *srvc* is provided by s, which includes a vulnerable function f', then another S that has not yet been included should be selected.

4.3.1 Advanced Evaluation

When *SP* starts to provide a service at a time t_0, it is impossible to assume that there are no adversaries. The *SP* must evaluate the security.

Supposing that the *SP* chooses values for the security parameter k and the usability parameter ub, the *SP* then chooses a scenario set $S = (S_1, S_2, \dots)$ that satisfies the following.

$$P(Acc|S(ID_u,a)) < k, P(Deny|S(ID_u,u)) < ub \quad (14)$$

The *SP* must pay attention to the usability of S. The user does not want to be asked to input to keyboard many times. Let ub' be the usability parameter for S. Note that *SP* will then require ub', and so ub' can be regarded as being included in ub.

However, before starting the service, it is impossible for the *SP* to have full knowledge of the security environment. In a multi-factor authentication system, there are cases that cannot be applied existing evaluation. To determine whether the service should be provided, it is necessary to evaluate the system based on some assumption, which can be empirically deduced, obtained from a similar system, or by updating the security parameter based evaluation of the need during the operation.

We suggest that the evaluation method should not assume that the security level will be constant during provision of the service, since prior to initiating it, it is not possible to obtain an accurate evaluation. However, before starting the service, it is possible to estimate the probabilities. As the service continues, the probabilities continue to be updated to the appropriate values, by using the strategy for optimization of security that we will discuss in Section 4.4.

In the proposed probability model, the evaluation value of the system is updated by the posterior probability using a Bayesian update.

4.4 How to Update the Parameter

We now discuss how the *SP* updates the evaluation value. We suggest two methods for updating the value: automatic and manual.

Auto-Update Auto-Update is automatically performed, and the new value is the log of the frequency of successful attacks by an adversary a.

Manual-Update Manual-Update is performed by the system administrator. This method is for reflecting changes in the security environment, such as the emergence of new technologies or the appearance of new attacks.

The continuity of service is important. For small vulnerabilities in one scenario S, it is not realistic to re-evaluate the entire system each time a new factor is added or deleted. Such a system could not continue until the evaluation had been completed, and this would result in a lack of continuity of service. Thus, it is necessary to construct a method that allows the factor to be updated dynamically in a multi-factor authentication system. However, for the users, the system must be flexible and convenient, and it is important that the users encounter factors for which they are prepared, rather than always being forced to adapt to a new method. We suggest that the actual scenario in use should be able to be chosen from several different scenarios by either the *SP* or u. This would allow the system to maintain both security and usability, and it would be a low-load technique for the user.

4.4.1 Auto-Update

The *SP* can observe the access probabilities from t_0 to t_{i-1}; $P(Acc|S(ID_u,a))$ will continue to change if the *SP* provides the *srvc* with scenario $S_i(i \in \mathbb{N})$. It is also possible to estimate the risk of the system by continuing to update the probabilities of acceptance of the user and the adversary assuming that there are attacks to the system.

Let $P_{t_i}(Acc|S(ID_{u_j},a))$ be $P(Acc|S(ID_{u_j},a))$ at t_i, so that

$$P'_{t_i}(Acc|S(ID_{u_j},a)) = \delta P_{t_i}(Acc|S(ID_{u_j},a)) + (1-\delta)\alpha(\delta = 1) \quad (15)$$

iff

$$P'_{t_i}(Acc|S(ID_{u_j},a)) = P_{t_i}(Acc|S(ID_{u_j},a)) \quad (16)$$

where δ is 0 or 1, and α is a fixed value.

These probabilities are based on statistical assessments. The *SP* observes that the user or adversary is accepted or denied access to the system and updates P'_{t_i}. Note that if the *SP* continues to offer the service based on the scenario S_i, the following formula should be satisfied: $P'(Acc|S(ID_{u_j},a)) < k$ and $P'(Deny|S(ID_{u_j},u_j)) < ub$.

4.4.2 Manual-Update

When the *SP* provides the service, a person who can predict rapid changes in the access probabilities because they have access to knowledge about new attacks or ongoing attacks controls the system.

If there is a reason that the administrator be considered as security is reduced, such as an attack will be widespread, the administrator needs to change the evaluation of scenarios related to the attack immediately. We will consider some examples of these kinds of attacks, which include password list attacks and attacks on biometric authentication, such as by gummy finger or wolf attacks.

Attack Forecast. At a time $t_i \in \mathbb{T}$, the administrator expects a rapid increase in attacks.

At t_i, the administrator sets $\delta = 0$. The access probabilities of all users are changed to

$$P'_{t_i}(Acc|S(ID_{u_j},a)) = \delta P_{t_i}(Acc|S(ID_{u_j},a)) + (1-\delta)\alpha(\delta = 0, \alpha = \alpha') \quad (17)$$

iff

$$P'_{t_i}(Acc|S(ID_{u_j},a)) = \alpha'. \quad (18)$$

Note that α' should not be constant before t_i, but it becomes fixed at t_i. It is possible to change many different values to account for the level of attacks.

Update From The Observation of Another User. The access probability is not only defined for u_i but also for many other users, including u_j. The administrator might make the following change:

$$P'_{t_i}(Acc|S(ID_{u_j},a)) = \delta P_{t_i}(Acc|S(ID_{u_j},a)) + (1-\delta)\alpha(\delta = 0, \alpha = \alpha') \quad (19)$$

The service provider will decide α'.

4.5 Replacement of Factors

If the access probability of S, $P'(Acc|S(ID_{u_i},a))$, is larger than the security parameter k, then the *SP* needs to take measures to fix this, and various options are available. There are multiple strategies that the system administrator can be taken. One of the strict strategies is service outage; a less drastic strategy is to change some scenarios, which means changing the authentication factors; the simplest strategy is to notify the system administrator. The evaluation value must be effectively utilized as a criterion for the operation of the system.

4.5.1 Changing Scenarios

If the access probability of $P'(Acc|S_i(ID_{u_j},a))$ is greater than the value of k for scenario S_i, the *SP* decides to stop using S_i. The *SP* selects another scenario S_l, and the service is provided using S_l. Note that S_l should satisfy $P'(Acc|S_l(ID_{u_j},a)) < k$.

S_l should be selected by considering usability and the impact on service availability.

4.5.2 Evaluation of a Scenario

After evaluating each scenario, the *SP* must find a new scenario that has not previously been considered. The *SP* should be able to adapt to a new scenario that is brought about by technical innovations or by a significant decrease in the security level of a particular factor.

4.6 Stop or Change the Service

For security measures, the *SP* is able to change of the quality of service, such as by temporarily lowering the maximum amount of money transferred or by reducing the amount of credit that can be accessed. To make the appropriate choice, the *SP* should use Tables 2. The *SP* must stop the service *srvc* if the access probabilities of all scenarios are greater than k: $P'(Acc|S_l(ID_{u_j}, a)) > k$ for all S_l. Also, the *SP* must stop the service *srvc* if the usability probability of all scenarios are greater than ub: $P'(Acc|S_l(ID_{u_j}, u_j)) > ub$ for all S_l.

5 CONCLUSIONS

This paper focuses on probabilistic framework for multi-factor authentication. In recent years, the security environment has been changing rapidly due to diversifying cracking methods and the improved functionality of computers and mobile devices. We discussed the need for evaluation methods that can change the authentication factor dynamically, and we proposed a probabilistic framework based on a Bayesian model. Our research makes two contributions. First, we showed a probabilistic framework for multi-factor authentication considering combination of authentication factors. Second, we showed a theoretical model that is able to change authentication factors dynamically. Moreover, we proposed a method for selecting a combination of authentication factors for changing them when the security environment changes. Our framework can improve the security and usability of multi-factor authentication. In the future, it is necessary to evaluate using actual case studies and data.

ACKNOWLEDGEMENTS

We would like to thank Mitsubishi UFJ NICOS Co., Ltd. for a grant that made it possible to complete this work.

REFERENCES

Al-Assam, H., Sellahewa, H., and Jassim, S. (2010). On security of multi-factor biometric authentication. In *Internet Technology and Secured Transactions (ICITST), 2010 International Conference for*, pages 1–6. IEEE.

Aloul, F., Zahidi, S., and El-Hajj, W. (2009). Multi factor authentication using mobile phones. *International Journal of Mathematics and Computer Science*, 4(2):65–80.

Bonneau, J., Herley, C., Van Oorschot, P. C., and Stajano, F. (2012). The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 553–567. IEEE.

Burr, W. E., Dodson, D. F., and Polk, W. T. (2004). *Electronic authentication guideline*. Citeseer.

Chen, L. and Crampton, J. (2012). Risk-aware role-based access control. In *Security and Trust Management*, pages 140–156. Springer.

Cheng, P. C., Rohatgi, P., Keser, C., Karger, P. A., Wagner, G. M., and Reninger, A. S. (2007). Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. In *Security and Privacy, 2007. SP'07. IEEE Symposium on*, pages 222–230. IEEE.

Damer, N., Opel, A., and Nouak, A. (2014). Cmc curve properties and biometric source weighting in multi-biometric score-level fusion. In *Information Fusion (FUSION), 2014 17th International Conference on*, pages 1–6. IEEE.

Google (2015). Google 2-step verification [retrieved 18 sep. 2015]. https://www.google.com/landing/2step/.

Herley, C. (2009). So long, and no thanks for the externalities: the rational rejection of security advice by users. In *Proceedings of the 2009 workshop on New security paradigms workshop*, pages 133–144. ACM.

Hocking, C. G., Furnell, S. M., Clarke, N. L., and Reynolds, P. L. (2010). A distributed and cooperative user authentication framework. In *Information Assurance and Security (IAS), 2010 Sixth International Conference on*, pages 304–310. IEEE.

Karabacak, B. and Sogukpinar, I. (2005). Isram: information security risk analysis method. *Computers & Security*, 24(2):147–159.

Kim, J.-J. and Hong, S.-P. (2011). A method of risk assessment for multi-factor authentication. *JIPS*, 7(1):187–198.

Kondakci, S. (2010). Network security risk assessment using bayesian belief networks. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pages 952–960. IEEE.

Nguyen, N. T., Zheng, G., Han, Z., and Zheng, R. (2011). Device fingerprinting to enhance wireless security using nonparametric bayesian method. In *INFOCOM, 2011 Proceedings IEEE*, pages 1404–1412. IEEE.

Pavlovic, D. and Meadows, C. (2010). Bayesian authentication: Quantifying security of the hancke-kuhn protocol. *Electronic Notes in Theoretical Computer Science*, 265:97–122.

Riva, O., Qin, C., Strauss, K., and Lymberopoulos, D. (2012). Progressive authentication: Deciding when to authenticate on mobile phones. In *USENIX Security Symposium*, pages 301–316.

Sabzevar, A. P. and Stavrou, A. (2008). Universal multi-factor authentication using graphical passwords. In *Signal Image Technology and Internet Based Systems,*

2008. SITIS'08. IEEE International Conference on, pages 625–632. IEEE.

Shay, R., Komanduri, S., Kelley, P. G., Leon, P. G., Mazurek, M. L., Bauer, L., Christin, N., and Cranor, L. F. (2010). Encountering stronger password requirements: User attitudes and behaviors. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*, SOUPS '10, pages 2:1–2:20, New York, NY, USA. ACM.

Yang, G., Wong, D. S., Wang, H., and Deng, X. (2008). Two-factor mutual authentication based on smart cards and passwords. *Journal of Computer and System Sciences*, 74(7):1160–1172.

Zhao, J. J. and Zhao, S. Y. (2010). Opportunities and threats: A security assessment of state e-government websites. *Government Information Quarterly*, 27(1):49–56.