Adaptive Authentication: Issues and Challenges

Khairul Azmi Abu Bakar and Galoh Rashidah Haron
Information Security Lab
MIMOS Berhad, Kuala Lumpur, Malaysia
Email: khairul.azmi@gmail.com, rashidah@mimos.my

***Abstract*—Authentication is a mechanism to establish proof of identities. Common practise for authentication is to challenge user to present authentication credential. User who can present the valid credential is considered as authenticated identity. Traditional authentication systems have a static security requirement which does not consider the change of user behavior or environment. In contrast, adaptive authentication systems are risk-based authentication that consider those changes to identify high-risk and suspicious illegitimate login attempts. In this paper, we examine some of the adaptive authentication systems that have been proposed recently in the literature. Some of the issues and challenges when developing an adaptive authentication system are also discussed. We also introduce our own Unified Authentication Platform (UAP) which incorporates adaptive control. We describe two typical processes that are used in our Adaptive UAP.**

***Index Terms*—adaptive authentication, behavioral profile, multi-factor authentication**

I. Introduction

Protecting confidential data in computing environments has long been recognized as a difficult and complex problem. All modern applications include some form of access control to protect resources from being accessed by unauthorized users. The systems for access management are commonly called AAA systems (Authentication, Authorization and Accounting)[7]. In this paper, we only concentrate on the authentication system.

Authentication is a process to confirm that someone or something is, in fact, who or what it is claimed to be. The process typically involves obtaining identification credentials such as a combination of a username and a password from user and validating those credentials against some authority. If the credentials are valid, the user who submitted the credentials is considered an authenticated identity.

In traditional authentication system, the decision on the level of authentication credential required solely depends on the application that the user trying to access or the transaction the user wants to perform. High sensitive applications (eg. internet banking) would demand the user to present stronger authentication credential than what less sensitive applications (eg. Facebook) would. The required authentication methods could also be a combination of two or more credentials, increasing the authentication security even more. However, for this type of system, the security requirement is static and does not consider the habitual behavior of the users

Adaptive authentication system uses environmental characteristics and behavioral profiling to identify high-risk login or activity attempts and dynamically customizes the authentication requirement accordingly. The system studies common behavior pattern of all users based on their past history login access. If a user follows the same patterns when logging into the system, the logon experience may probably be a combination of a username and a password indicating a low risk attempt. If a user tries to login under different behavior or environment, the identity of the user will be questioned. The system may adaptively challenge the user to present stronger or additional authentication credentials to prove his/her identity.

The remainder of the paper is organized as follows. Section II provides brief description example of adaptive systems that have been proposed for the last few years. The analysis and discussion on the example systems are presented in Section III-A. Section IV introduces our adaptive Unified Authentication Platform (UAP) along with description on its two main processes. Finally, Section V draws the conclusions.

II. Examples of Adaptive Authentication System

In this section, we will review various adaptive authentication systems that have been proposed over the last few years. For each model, we will review the working principle, in light of the discussions presented in the previous sections.

A. A2BeST

Rocha, Lima and Dantas presented a novel adaptive authentication service based on mobile user's behavior and spatio-temporal context (A2BeST) [8]. The approach offers dynamic authentication of users in pervasive and mobile environments that use mobile devices (e.g. smartphones and PDAs) as interface to access services and applications. Mobile devices provide access to resources such as user's calls, user's schedule, GPS, battery level, user's applications and sensor during his interaction with the system. These resources are used to identify user behaviour through the following mechanisms:

- Explicit profile: it is created during the first interaction between the system and user. This profile contains the events extracted from his personal contracts and schedule stored on the mobile device.

978-1-4799-0462-4/13/$31.00

- Session profile: consists of the execution context of the user. It also indicates the user's status which is determined by the user's last performed actions.
- Implicit profile: it is created by processing the user's events and his explicit profile. It contains the actions that were taken by the user as well as the spatio-temporal characteristics of these actions. This profile is determined by a strategy of recommendation based on the Vector Space Model (VSM).
- VSM Filter: it is a filter that uses the vector space model to determine the relevance of information. This filter uses a formalism to calculate the similarity among the profiles to be analyzed.

The context of time and location are used as the fundamental properties to determine successive events which defines behavioral profiles. By utilizing the resources found on mobile devices, current event will be analyzed using spatio-temporal permutation model to identify conformity in the behaviour standard and possible behaviour anomalies.

The architecture of the system is illustrated in Figure 1. The user context is responsible for capturing all situations that determine the occurence of a new event before sending the the event description (ei) to the beliefs analyzer. The beliefs analyzer is responsible for defining behaviors or beliefs as well as the classification of events and the inference of behaviors through the activities, stored profiles and the preceived and registered events. These behaviours are analyzed by similarity or probabilistically in order to:

1) define new occurences
2) determine the attitude adopted by the system
3) determine actions that will be taken

The VSM filter is responsible to determine the similarity degree. If the similarity rate is greater than a certain value, the profile is considered to be relevant and will be stored in the system as a new implicit profile with the same weight of the similarity rate.

The probabilities analyzer is used to categorize the user based on the conditional probabilities of his behavior. Based on user's past and present activities, the classification could be either one of the following four cases: normal, abnormal and two cases of suspicious execution. Normal execution entails the same activity in the same spatio-temporal context. Abnormal execution involves different activities in different spatio-temporal context. The two cases of suspicious execution occur when the same activity is performed is a different spatio-temporal context or when different activities occur in the same spatio-temporal context.

Finally, the challenger determine how the user will be questioned to prove his identity on the system, based on the categorization made by the probabilities analyzer and the level of authentication required for the desired operation. When the user responds the challenge correctly, he is authenticated in the system and be allowed to executes the desired operation. The requested event is inserted in the user profile which contains the history of his interaction with the system.

In A2Best, user behaviour analysis is done at module belief analyzer. First, behavioral attributes to be considered

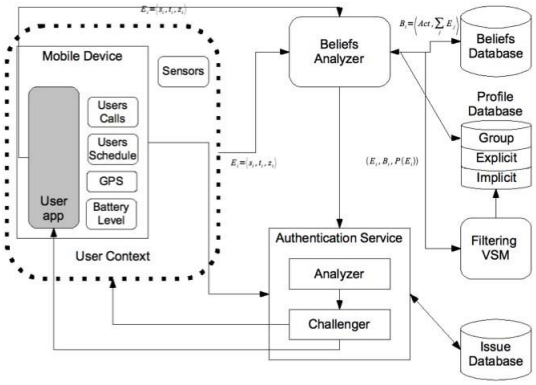

Figure 1: A2BeST System Architecture

for the proposed target scenario need to be defined. Each of the behavioral attribute is assigned with weight defined as *VectorP* in order to set different priority levels to the analysis of the different comparative attributes. Another vector defined as *VectorE* is determined by comparing the captured behaviour with the session profile. The attributes values that remained unchanged is set to one (1) as its value and those that changed is set to zero (0) as its value. For spatio-temporal attributes (space and time), the value is evaluated by using the spatio-temporal permutation model. By using SaTScan tool and previous collected events, the system compares the p-value of the two executions of the spatio-temporal permutation model (with co-variables and without co-variables) and take the minimum value between them.

Both vectors (*VectorP* and *VectorE*) are used to compute the similarity degree between the captured vector and the session profile. Depending on the value, the user will be categorized into any of the following profiles: normal, suspect and abnormal. Normal profile contains characteristics that can support the automatic authentication of the user. Under suspect or abnormal profiles, the user will be presented with a challenge question. The difficulty level of the challenge question corresponds with the authentication level required by the application being accessed. Similarly, questions for the abnormal profile are more difficult than for the suspect profile. If the user answers the challenge question correctly, the history of the interaction with the system will be inserted in the user profile. Consequently, the system is able to widen its knowledge base, refining the authentication process according to the number of interactions with the user.

B. Implicit Authentication through Learning User Behavior

Shi et al. [9] present an implicit authentication mechanism that observes user behavior for authentication. The mechanism focuses on the user of mobile devices which are capable to gather and record a rich set of information such as location, motion, communication and usage of application. Figure 2 outlines the framework of a machine learning algorithm which is used by the presented mechanism.

The information gather by the user device is used to form detailed historical profile of the user. Based on the user's past behavior, habitual events which characterize an individual

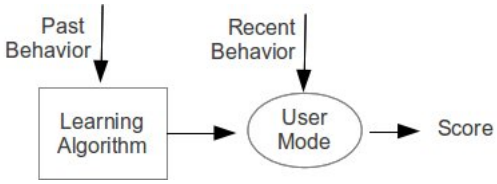

Figure 2: Machine Learning Algorithm

behavioral patterns are identified and stored as user model. Given the user model and some recently observed behavior, the system computes the probability that the device is in the hands of the rightful owner. That probability is referred to as an authentication score which can dynamically change with the occurence of a new event. Whenever a positive (habitual) events are observed, as buying coffee at the typical store, in a similar period of time, the score increases. Whenever a negative (sporadic) events are detected, such as the call to an unknown number or sudden changes of expected location (an event associated with the theft or the inappropriate use of the device), the score decreases.

The passage of time is treated as a negative event in which the score gradually decreases. The score is used to make an authentication decision. When the score reaches a minimum threshold, the user have to explicitly authenticate to the device, perhaps by inserting a passcode. The successful authentication will boost the score once more. The threshold can vary for different applications, depending on the sensitivity of the application security.

C. TBAS

Sathish Babu and Venkataram [5] present a Transaction-Based Authentication Scheme (TBAS) which aims to be an effective, dynamic and intelligent decision based authentication techniques for mobile communications. The proposed scheme uses a type of intelligent agents called Cognitive Agents (CA) which have high reasoning capability to solve complex problem. A cognitive act consists of three general actions [10]:

- perceiving information in the environment
- reasoning about those perception using existing knowledge
- acting to make a reasoned change to the external or internal environment

Figure 3 illustrates the architecture of TBAS which uses two types of cognitive agents: mobile cognitive agent (MCA) and static cognitive agent (SCA).

When a service request is initiated by a MU, SCA creates the MCA and sends it to the client mobile device along with the belief formulator logic. The MCA acquires new values for behavior parameters, computes probabilities of occurence of various behaviors and formulates beliefs using the belief formulator. It communicates the belief along with the transaction details to the SCA during every client transaction. The agent logs in all new observations and stores them in the observation storage. When required, the MCA also provides those observation to the SCA.

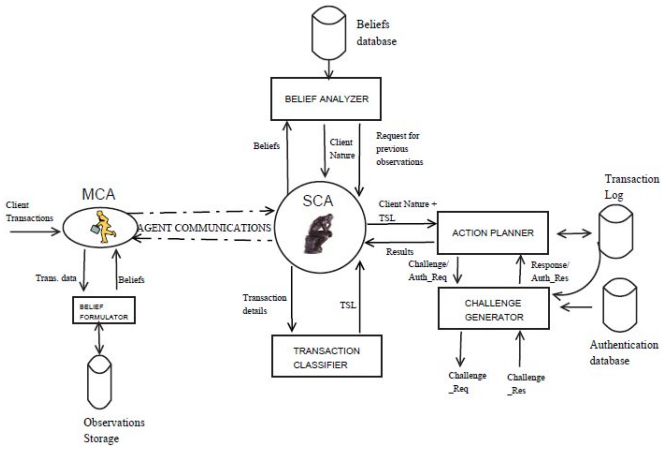

Figure 3: The TBAS Architecture

The SCA co-ordinates functioning of all the components at the system. The beliefs and the transactions details from the MCA are submitted to the belief analyzer and transaction classifier respectively. The belief analyzer computes the deviation factor with respect to the corresponding established beliefs in the belief database. Based on the value of the deviation factor, the belief analyzer passes its opinion about the client nature back to the SCA. The client nature can be either one of the following three groups: normal, suspicious and abnormal. When the client is turning up to be suspicious, the belief analyzer may generate new belief to support belief analysis. In that case, the belief analyzer sends a request to the SCA to get the generated observations from the MCA.

The transaction classifier decides on the Transaction Sensitivity Level (TSL) which is generated by analyzing various transaction parameters such as type of operation, type of data, sensitivity of data, volume of data etc. In the system, the TSL is ranging from level 0 to 3.

The SCA sends opinion on client nature from the belief analyzer and the TSL value from transaction classifier to the action planner to perform the futher action. All transactions with TSL value 0 are executed without require any authentication by the system. If the transactions are appearing for the first time and TSL>0, the action planner instructs the challenge generator to create an initial authentication for that transaction based on its sensitivity level. Otherwise, the action planner decides its future actions based on the value of client nature. If the TSL is 0, no authentication is executed.

The challenge generator generates authentication challenge question based on information stored in the authentication database, the beliefs database and the transaction log. The communication between the challenge generator and the MCA is encrypted using the security algorithms of the corresponding TSLs. The challenges are encrypted before being sent to the MCA. The MCA decrypts the challenge, obtains response from the MU and sends the encrypted response back.

D. APBAF

The authors presents session specific server side software authentication model based on user specific patterns [6]. The

model exploits the fact that software usage pattern or behavior of each user has a unique signature.

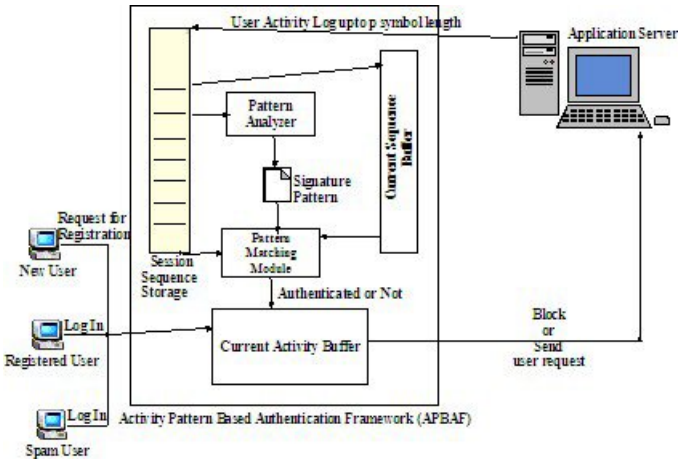

Figure 4: APBAF

As illustrated in Figure 4, there are several modules in the framework. The SessionSequence is a storage that contains all activity patterns. The PatternAnalyzer module checks the whole SessionSequence storage and identifies the repetitive sequence(s) of activity patterns called SignaturePattern of the user.

One main advantage of this model is the ability to mitigate the risk of software corruption by any spam user. During the permissible monitoring period, all the user activities are stored temporarily into a buffer. Only after the successful authentication, the content of the buffer is transfered to the main application. Otherwise, it is cleared. The authors claim that this process referred to as Activity Transfer provides a probable solution to the main disadvantages of any Risk Based Software Authentication.

III. Discussion

In this section, some of the issues in adaptive authentication system are discussed.

A. Geographical Information

All of the proposed mechanisms make use of geographical location information to derive behavior profiles and the trust level of the user being authenticated. Generally, there are two ways to get user's geographical location. There are number of free and paid subscription geolocation databases that can identify the real-world geographic location of an object based on its IP address. URL sites such as [1], [2], [3], [4] provides geolocation information ranging from country level to state or city, each with varying claim of accuracy. However, those services could not give the actual location if the user is accessing the system through proxy or VPN (Virtual Private Network) server. In that case, the user will be detected bearing IP address of the proxy or VPN server. Most mobile devices have built-in global positioning system (GPS) that offer relatively reliable way to track the user's location. A reliable GPS reading requires unobstructed line of sight to at least four satelites well above the horizon. As a result, a GPS receiver does not work at indoor environment such as inside a building or a tunnel.

B. Predefined Values

Although adaptive authentication system requires no system administrator intervention during the operating time, all of the proposed mechanisms mentioned in this paper requires a predefined values to be set mainly in weightage and threshold.

1) Weightage: In the trust evaluating process, there are many attributes that need to be considered for user classification. Some attributes are more significantly important to reflect user behavior pattern than the others. For example, logging using unfamiliar type of browser brings is less suspicious login attempt than logging from an uncommon geographical location. In this case, each of the attributes (browser type and location) will be assigned a different weightage value.

2) Threshold: The result from the trust calculating process is a score representing by some numbering value. This score reflects a confidence level about the identity of the logging user. Based on the score, the system decides on the user category: for instance normal, suspicious or abnormal group. Each of the group has a range of values which are threshold value to decide the category of the user.

C. Performance Metrics

Not all proposal measures their algorithms with some kind of performance metrics. Sathish Babu and Venkataram [5] use two metrics: average authentication delay and average authentication cost. Average authentication delay is defined as the sum of the time taken for the mobile user to receive the authentication reply over a number and type of transactions in a unit time. The average authentication cost is defined as the sum of signaling load and processing load for cryptographic techniques during each authentication operation over a number of authentication requests per unit time.

Shi et al. [9] evaluate their proposal by using two metrics. Since their algorithm focuses on authenticating user for the mobile device, they measure the performance by 1) calculating the number of times the legitimate user used the device before a failed authentication and 2) the number of times the adversary used the device before detection.

However, of all the proposal mentioned in this paper, none of them compare their approach against the others.

D. Challenge Questions

All the mentioned proposals, except in [9], challenge the user with a question when the trust score falls below the minimum threshold. The difficulty level of the challenge question depends on the user category (suspect or abnormal) and the authentication level required by the application (high, medium or low). The most difficult challenge question is set when the user category is abnormal and the required authentication level is high. In Shi et al. [9], no challenge question will be asked. Only if the trust score falls below the threshold then the system will ask the user to present the conventional password.

E. Handover Protocol

One interesting feature that is mentioned in [5] is the idea of implementing a handover scenario. Handover process in authentication system allows the authentication status from one service provider to be transfered to another without loss or interruption of service. This is an important feature in the enterprise environment where there are many service providers offering similar services to the mobile device. When the device is moving away from the area covered by one service provider and entering the area covered by another provider, the service is transfered to the second provider to avoid service interruption. At the same time, the authentication status including belief deviation factor, session key in use, challenge information and beliefs need to be transfered securely with the new service provider so that the authentication process will continue without any discontinuity. Unfortunately, this interesting feature is not implemented yet by the authors and only suggested as a future work.

F. Privacy

The ability to capture great amount of information leads can lead to privacy issue. Privacy is defined as the right of users not to reveal information about themselves and the right to keep personal information from being misused. Shi et al. [9] protect user privacy using a keyed hash to obfuscate phone numbers, SSIDs and URLs. The key is randomly generated during install time and stored only on the device. As a result of the obfuscation, the system can only identify instances of the same phone number or URL on each user's log but is unable to determine whether two users have overlapping contacts or URLs.

IV. Unified Authentication Platform (UAP)

In MIMOS Berhad, we have developed a security product called Unified Authentication Platform (UAP). UAP is a centralised multi-factor authentication system with web-based single sign-on (SSO) capability to manage user authentication profiles. It is designed to manage front-end application authentcation using an established protocol, Secure Assertion Markup Language (SAML) protocol, which provides a centralised authentication framework and aims to reduce significant application changes at the backend. Unlike many of the proposed adaptive authentication system mentioned before, UAP is targetted to serve for both desktop and mobile users. The objectives of UAP are as the following:

1) provide an infrastructure that offers authentication service to applications
2) provide information technology that de-couples authentication function from application
3) grow indigenous authentication mechanism industry throughout the country
4) a unified authentication platform initiative for enabling government e-services application

UAP has the equivalent functionalities as the Shibboleth which is a free open-source project but with the support of multiple authentication methods. Users can choose any of the authentication methods supported by UAP to get authenticated and be allowed to access various applications without having to go through the same authentication process again.

For the new version of UAP, we will incorporate adaptive control based on security risk and level of assurance. Each application will be assigned a threshold level which denotes the minimum required trust score that user need to acquire before access is granted. Sensitive application such as online banking would require high trust score.

The trust score acquired is depending on the authentication method that has been selected by the user. Each authentication method would carry different security strength value. Every time the user present a valid credential to the system, the user would acquire a trust score. The user can step-up the trust score by presenting different valid credential that has not selected before.

The final trust score can be adjusted based on the security risk from the user behavioral profile. If a user logins under different environment from the user's formed behavioral profile, the system will consider the login attempt as a suspicious activity and decrease the user's trust score. The user may have to present additional credential to raise the trust score to gain access. If the difference is too big, the system can have two options. It can either block the suspicious user from logging in or grant access only after all the possible credentials have been presented by the user.

A. Processes in an Adaptive UAP

Adaptive UAP consists of two basic processes: Pattern Generating and Trust Evaluating.

1) Pattern Generating: The pattern generating process is responsible to analyse the users behavior from the history past events and comes with the corresponding patterns. As depicted in Figure 5, there are three components in the pattern generator: events storage, patterns generator and patterns storage. The events storage contains records of context informations from the past events. These records resides in a database. The pattern generator is the essential part of the pattern generating process. The pattern generator analyses and identifies similar repetitive sequence events and convert them into behavioral patterns. All of the generated patterns will be stored at the pattern storage and used in the next process.

This process consumes processing power since it needs to retrieve and analyze a great amount of data. As a result, in Adaptive UAP, the pattern generating process will be executed during midnight or when the system is not busy.

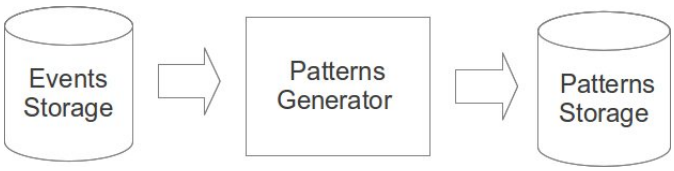

Figure 5: Pattern Generating

2) Trust Evaluating: The trust evaluating is responsible to analyze, decide and act upon every login request from users. As illustrated in Figure 6, the trust evaluating process

contains five components: contexts collector, patterns storage, trust calculator, challenger and events storage. The context collector generates a collection of data reflecting the current environment parameters of the users. There are many types of data that could be captured. Typical environment parameters used are the location of the user and the time the event is taking place. Patterns storage contains the registered pattern generated from the pattern generating process mentioned before. The trust calculator is the main component of the trust evaluating process. The trust calculator compares the current contexts captured by the context collector with the patterns from the pattern storage to decide the total trust score of the user. The challenger component determines the action to be taken based on the final trust score of the user calculated by the trust calculator and the threshold level of the application that the user wants to access. The action could be either to grant access to the user or challenge the user to provide additional information or in the worst case, block the user from logging into the system. The events storage stores the data from the context collector and the decision from the challenger. The events storage will be used by the pattern generating process which then completes the closed-loop mechanism for the two processes.

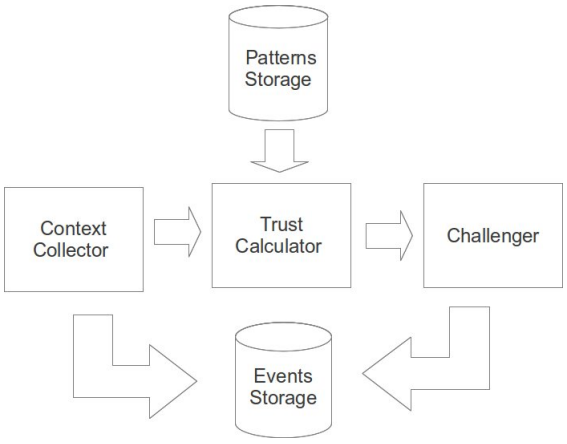

Figure 6: Trust Evaluating

V. Conclusion

Adaptive authentication is a means of providing additional security layer to an authentication system for end users without them having to know it is as work. An adaptive authentication system looks at various parameters such as the time the logon request is taking place and the location of the users to form their behavior profile. It will then takes into consideration that behavior profile to confirm the identity of the users. Any deviation from the profiles is granted as a potential risk and the system may request the users to perform additional steps to verify their identity. In this paper, we studied some of the systems that have been proposed in the recent years. Some of the issues on adaptive authentication system are also discussed in several aspects. We also presented our Unified Authentication Platform (UAP) which will be incorporate adaptive element in it. We present two general processes that are used in the adaptive UAP. The system defines behavioral profile based on the contexts from the past history records. The profile is then compared against the current context to come out with the final trust score which characterizes the login attempt.

Acknowledgement

We acknowledge the support provided by Ministry of Science, Technology and Innovation (MOSTI) in funding the MIMOS Unified Authentication Platform (UAP) project through the Tenth Malaysia Plan (10MP). The completion of the project allows the delivery of a centralized authentication infrastructural platform for web applications.

References

[1] http://www.maxmind.com. [Retrieved 13 April 2012].
[2] http://www.quova.com. [Retrieved 13 April 2012].
[3] http://www.ip2location.com. [Retrieved 13 April 2012].
[4] http://www.ipligence.com. [Retrieved 13 April 2012].
[5] Sathish Babu B. and Pallapa Venkataram. A dynamic authentication scheme for mobile transactions. *International Journal of Network Security*, 8(1):59–74, January 2009.
[6] Ayan Chakraborty, Shiladitya Munshi, and Anirban Kundu. An adaptive server side software authentication framework based on user's activity pattern. In *Second International Conference on Emerging Applications of Information Technology*, pages 153–156, February 2011.
[7] Rui He, Man Yuan, Jianping Hu, Hong Zhang, Zhigang Kan, and Jian Ma. A novel service-oriented AAA architecture. In *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communication 2003 (PIMRC 2003)*, volume 3, pages 2833–2837, 2003.
[8] Cristiano C. Rocha, Joao Carlos D. Lima, M. A. R. Dantas, and Iara Augustin. A2best: An adaptive authentication service based on mobile user's behavior and spatio-temporal context. In *IEEE Symposium on Computer and Communication (ISCC)*, pages 771–774, June 2011.
[9] Elaine Shi, Yan Niu, Markus Jakobsson, and Richard Chow. Implicit authentication through learning user behavior. *Information Security*, 6531:99–113, 2011. Lecture Notes in Computer Science.
[10] Todd Shimoda. A theory belief model for cognitive agents, 2000. Colorado State University.