# An Authentication Framework for Nomadic Users

**Ahmed, Naveed; Jensen, Christian D.**

*Publication date:*
2009

*Document Version*
Early version, also known as pre-print

# An Authentication Framework for Nomadic Users

*Naveed Ahmed and Christian Damsgaard Jensen*
*Department of Informatics and Mathematical Modeling (IMM)*
*Technical University of Denmark, Denmark*
*nahm@kth.se, christian.jensen@imm.dtu.dk*

## Abstract

Security and usability are often horn locked and system administrators tend to configure systems so that they favor security over usability. In many cases, however, the increased security results in usability that is so poor that users feel the need to circumvent the security mechanisms. This is probably best explained by considering password based authentication, where a user is actively involved in the process. If the time required to log in to an account is considered too high, users tend to leave their terminals logged in throughout the day and share their account with other users. This is particularly true for nomadic users who move around in ubiquitous computing environments and avail from different IT services from many different locations. In many ubiquitous computing environments, where information processing is not considered the main priority, management often accepts this practise in order to increase productivity, e.g., in a hectic hospital environment, medical staff has to login and logout of various machines several times in an hour, but the repeated interactions consume a considerable amount of time, causing organizational inefficiency, job frustration and a tendency towards defeating the obstacle by leaving terminals logged in or choosing short and easy to type passwords. Therefore, a password based authentication mechanism, which is quite simple and secure in personal computing, has become too cumbersome for nomadic users, which means that other means of authentication must be developed for nomadic users.

In this paper, we focus on usability of authentication for nomadic users in a ubiquitous computing environment. We identify requirements for authentication of nomadic users and propose an authentication framework for this class of users. A prototype of the proposed authentication framework has been developed, which supports persistent and multi-factor authentication without the active intervention of a user.

We evaluate the usability of the developed mechanism by considering the time required to authenticate when logging in to a workstation and compare this to classic password based authentication. The evaluation shows that the proposed mechanism saves a significant amount of time for the nomadic users, which reduces the incentive to circumvent the authentication mechanism. Thus, the mechanism will both provide users with better job satisfaction and increased organizational efficiency, while at the same time increase the effective level of security of the system.

***Keywords:*** *Security, Usability, Ubiquitous Computing, Nomadic Users, Authentication.*

## 1. Introduction

In the last four decades, human-computer interaction has gone through an evolution. This evolution reflects three different paradigms of computing, which are identified by Allan Kay, Tomei[24], and Weiser[28]. We refer to these paradigms as centralized or mainframe computing, decentralized or personal computing and ubiquitous computing. In the early days of computing, centralized computing was the predominant paradigm, where a single mainframe computer was shared by multiple users on a time or resource basis. Even today, centralized computing plays an important role in the world's largest corporations, including many Fortune-1000 companies[25][27]. From the late 80s, personal computing started to dominate, marked by the number of PC users crossing fifty million[40]. The personal computing paradigm requires a computer to each user and it includes both fixed and mobile computing. More recently, ubiquitous computing paradigm has become more pronounced. In ubiquitous computing, users avails many different machines that are embedded into the environment. Computers used in ubiquitous computing environment are often characterized by being small, inexpensive, robust, shared, networked, and distributed all over the places where they might be required[29].

This evolution of computing paradigms is also reflected in the way computers are used. Depending on how

a person fulfills his computing needs, we identify three phases of user evolution. In the first phase, a user has to access computers that are placed in a single physical location whenever he has a problem that requires computational resources. Thus the freedom of mobility for a user is severely constrained by physical or virtual access to the computing resource. In the second evolution phase, which is mobile computing, users are enabled to move freely, because they can carry their computation resources with them. More recently, a third phase of evolution has emerged, as we start embedding computing devices into artifacts of the surrounding environment. We have termed this as nomadic use of computing, because it is characterized by the fact that there is no inherent need to carry a computing device nor is there any requirement to access a single computer in a central location, because computers have become part of the environment, thus a nomadic user can freely move and compute where ever and when ever required.

Unfortunately, development of suitable security mechanisms lag behind the user's evolution towards nomadic use of computing. The most obvious example of this, is password based authentication, which is used in most operating systems like UNIX, Linux and Windows etc. This mechanism does not impose any serious usability limitation for stationary or mobile users, but for nomadic users, who have to frequently login, logout and share terminals, usability factor plays an important role to determine its effective level of security. Most common security observations are use of small and easy to remember password, sharing password with colleagues and with in a group, omitting to logout etc. From a pure usability point of view, an ample amount of time is being consumed in authentication process, so it is often considered an obstacle to “real work” by users who do not appreciate the underlying need for security.

Common usability problems associated with password based authentication are already highlighted in the literature. A pilot study in health care[34] highlights typical nomadic use cases which cause security vulnerabilities. Jeyaraman and Topkara[3] have recognized usability problems of passwords very well and have proposed some complementary enhancements. A survey for phrase-based passwords shows many vulnerabilities in their practical use[6]. On the other hand, to achieve usability, some graphical password schemes have been proposed[38][4]. Hopper and Blum[5] proposes an alternative to passwords authentication but the result still imposes usability constraints for nomadic use. A security analysis of passwords based authentication by Gorman[19] overlooks this important factor of usability which is significant for nomadic users. The majority of proposed solutions focused on stationary and mobile users and thus lacks in addressing nomadic use of computing where usability requirements are more stringent.

Corner and Noble[7][12] have put forward an authentication mechanism which does not require user interaction at all. Bardram et al.[11] have proposed a secure user authentication mechanism which is proximity based. But their main focus is to avoid vulnerabilities that are part of personal computing paradigm such as theft. The idea of proximity based login has also been used in many other products in ubiquitous computing research like Active Badges[13][21], AT&T's ActiveBat[15] which also uses session migration, Microsoft EasyLiving[14] , IBM's BlueBoard[16], AwareHome[20] and Personal Interaction Points[12] which uses RFID tokens for authentication and XyLoc System[17] etc. However these mechanisms are designed for particular work environments which might be consider as subset of a general nomadic environment. Nevertheless their designer has not analyzed nomadic use cases extensively, to come up with generic requirements which might be applicable to any authentication mechanism used in a nomadic environment.

Similarly the use of multi-factor authentication has extensively been investigated in literature for biometrics[1][2]. Jonnson's 'Jury' framework[8] appears very useful to merge any number of authentication mechanisms together. However, primary focus of research in the area of multi-factor authentication is exploration towards a higher level of confidence in authentication mechanism and thus mostly deficit from usability aspects.

In this paper we have investigated usability aspects of classic password based authentication mechanisms. As a result we have came across some intriguing facts, which must be considered in order to make authentication mechanisms more usable and to raise effective level of security. In fact, our findings points to a simple criteria that any usability feature, which is linked to system security, must be conceded as an integral part of security architecture. Thus, we have designed a authentication framework and have implemented a scaled down version of it in form of a prototype, which consists of a Debian Linux system and RFID reader. The evaluation of system shows a considerable improvement in terms of usability and thereby security.

The rest of the paper is organized as follows; In next section we have presented a brief about password

based authentication in nomadic context to highlight several usability constraints and security vulnerabilities. We have also recounted some relevant security notions. The third section contains details of our designed authentication framework and its prototype implementation. In the fourth section we have presented results of our usability and security analysis, and in last section we have concluded our discussion.

## 2. Authentication and Nomadic users

To envisage the role of authentication in system security, let's recall some security notions. A security mechanism enforces a security policy. This security policy defines the intended level of security for a security mechanism[39] and we refer to this as 'Designed Security'. However, the actual security achieved in practice is often less than or at most equal to this intended level, we call this the 'Effective Security' level. Moreover, effective security tends to decrease with time, mainly due to the discovery of more vulnerabilities in security mechanisms, advances in technology, improved techniques of cryptanalysis etc. In an authentication mechanism for nomadic users, there is another aspect of this difference which is due to usability caused by repeated and lengthy interactions. Generally, human tends to become more casual with the passage of time. This is illustrated in the Figure 1, which depicts a gradual decline in the effective security over time. It is important to note, however, that the curve shown in Figure 1 is for illustrative purposes only; the actual shape of the curve depends on the type of the system, the configuration and operation of the system and parameters of the computational environment in which the system is deployed.

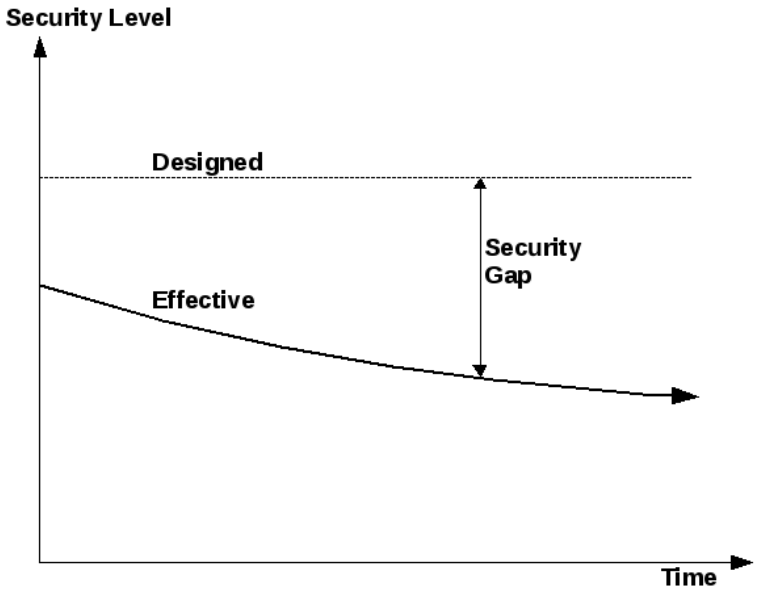


*Figure 1: Difference of Effective and Designed Security levels*

Password based authentication is a prerequisite for most access control mechanisms implemented in modern operating system like Linux, Unix and Windows etc. Thus passwords are our first and most fundamental security check to ensure computer security. An authentication mechanism is primarily designed to verify the claim made by a subject about their identity[19]. All mechanisms, which authenticate a human, can be divided in two classes. First class is called human-centric authentication and involves recognizing intrinsic or pseudo-intrinsic property of a human. Although Dhamija and Perrig[37] split this class in knowledge based and biometric authentication but we consider knowledge as human pseudo-intrinsic property. Second class is called device-centric authentication, where user is not directly authenticated. In fact, authentication mechanism authenticate a device on behalf of user.

A password based authentication is essentially a human-centric authentication. Generally a human-centric authentication needs active involvement of user and thus can't be done frequently due to usability constraints. For example it is difficult to do persistent authentication after each 500ms for pass phrase or voice recognition. However in device-centric approach, for instance repeated scan of RFID tag after each 500ms is quite common.

Security vulnerabilities using password based authentication have already been identified, especially in the context of health care[35][36]. For instance, modern hospitals have quite ubiquitous infrastructure in form of well connected network with centralized access to electronic patient's record (EPR). Doctor's work place have a couple of terminals to facilitate the check-up of patient or to carry out some research work. Conference halls and meeting rooms have their own terminals. Nurses use terminals at their work place to keep track of patients. Drug stores have a few terminals of their own which are connected to central data base to get information about prescriptions. Emergency rooms and operation theaters also have associated terminals. In a ward, there could be a terminal behind each patient's bed displaying important information regarding the care of the patient to authorized staff. It might be desirable that when a doctor visits a patient in a ward, all relevant data is available in a particular context. Some hospitals also provide terminals in ambulances which are then connected to the hospital information infrastructure. For example this may be used for triage in large emergencies or in-advance preparation in emergency treatment for heart patients.

It's interesting to look at how nomadic users work in the hectic environment of hospitals. For example a doctor has to see dozens of patients, he has to respond to emergency calls and he has to carry out surgery. After a few days, each doctor will have a complete different set of patient and possibly work at many different locations in the ward. As hospitals have to operate 24 hour a day, so typically they work in three or more shifts. One can imagine how computing devices are being shared with in single shift and across shifts. Moreover most of these nomadic users are handling sensitive health data of patients.

The typical behavior of such nomadic users is summarized in following points[36];They tends to use short and easy passwords[41][34], in their desire to login quickly. If the system enforces mechanisms to ensure the quality of passwords, people tend to write their password down, in order to avoid forgetting it. Sometimes they do not even logout. This might be on purpose to save time on their return or due to age related forgetfulness[22]. Passwords may be sniffed due to continual need to be typed in during nomadic use. Users tend to give away their password to their colleague for delegation purpose and sometime share it in a group.

Beside the wastefulness of ample amount of time, there are also very obvious security vulnerabilities. Short and easy passwords are always subject to numerous attacks as indicated in some security analysis[19][9][10], and this is more certain for nomadic use. The common problem with classic knowledge based authentication is that people tend to forget it[37][41], causing denial of service. Writing a complex password on a piece of paper is violation of very basis of human centric authentication and vulnerable to stealing. We agree with Bardram[11] and consider logout as an integral part of the authentication process. Failure to do so, is in fact a failure of the authentication mechanism. Password sniffing is very difficult to avoid in nomadic use. Giving away one's password is like making a copy of an authentication token and giving it to someone-else, and moreover allowing to make further copies of that. This is the worst form of delegation. In group based password scheme, no one is accountable although every one is authorized. Revocation is difficult to achieve in password based authentication. Accenture Terminated Employee Survey[41]shows that about 10% of ex-employee access the databases of their former employer. This is because employer were not able to properly revoke their permissions. Moreover revocation takes effect when user tries to re-authenticate and thus in some cases attacker get a chance of stealing a session. Active user involvement in authentication process consumes lot of time when we add small chunks of time of all nomadic users. Apparently this is a matter of efficiency but may cause partial denial of service attack. An attack on system security is always expected from weakest point[31] and thus these vulnerabilities are points of attack and must be addressed.

After analyzing password based authentication mechanisms in the context of nomadic users, we have reached on some conclusions. First of all, there should be no conscious interaction between a user and the authentication mechanism. As we have seen previously, that a small amount of active interaction ends wasting a substantial amount of time when we add authentication events over a period of time. Moreover, frequent interruptions and delays provide strong motivation for a busy nomadic user to circumvent the authentication mechanism.

Further we believe that the underlying architecture of an authentication mechanism should support persistence and preferably context based authentication. This avoids a major usability constraint by shifting responsibility of logout from users to the authentication mechanism in a way that preserves accountability. We also conclude that the mechanism should support a user-friendly delegation preferably at authentication level. This reflects the fact that in practice, delegation is normally accomplished by giving away one's password or using a group password. And at last, authentication mechanism should be multi-factor. This is

because human centric authentication is formidable to perform periodically for sake of persistence and in most of the cases a conscious user engagement is required. On other hand device centric authentication has problems of it's own in form of stolen tokens and lack of trust on single authentication device etc.

Some of these conclusions, although in a different context, are already in the state of art in form of Corner and Nobel's work on persistent authentication[12], Bardram's proximity based login[11] and Jonsson's co-authentication framework[8]. Importantly, our conclusions are implementation independent and also not aimed to increase the level of designed security. In fact, their expressed purpose is to avoid usability constraints but indirectly they avoid the decrement in effective level of security in nomadic environments.

## 3. Authentication Framework for Nomadic Users

We have designed an authentication framework based on our analysis presented in the previous section. Our framework is intended for a networked nomadic environment and uses persistent and multi-factor authentication techniques without conscious engagement of users. We have termed it as 'Authentication Framework for Nomadics', and abbreviated as AFN. Following is the design detail and rationale for AFN.

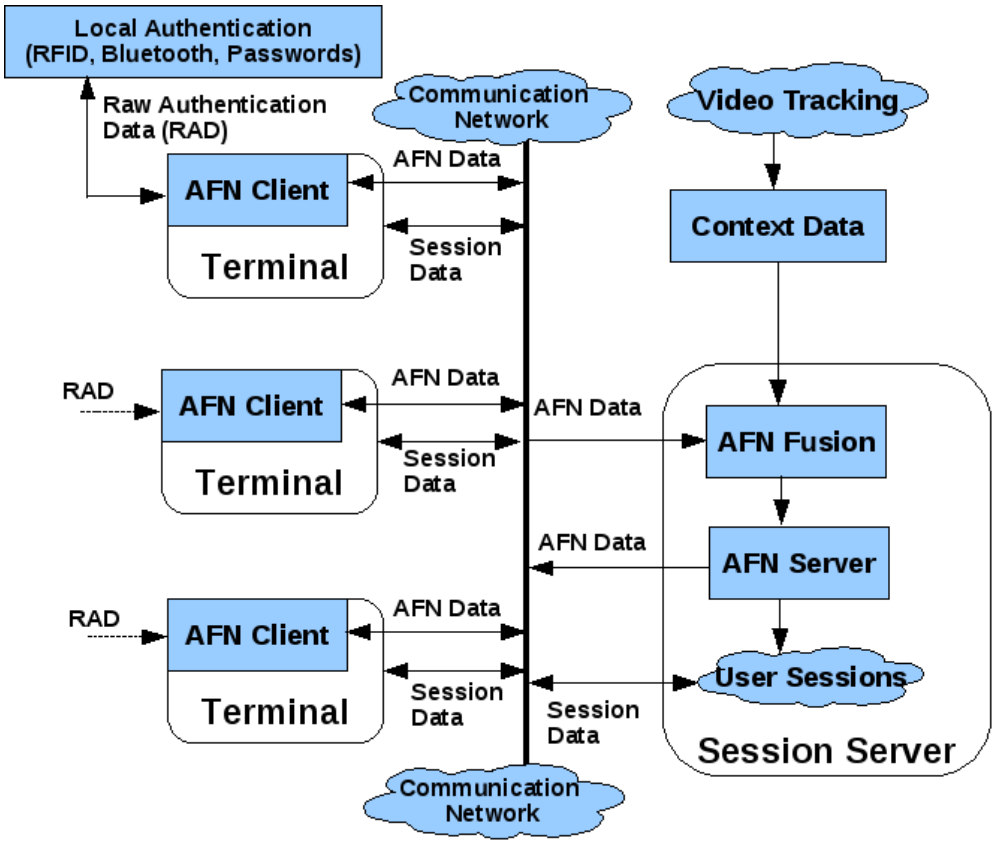


*Figure 2: Top-level Architecture of AFN Authentication Framework*

The basic architecture of AFN is shown in the Figure 2. It is a distributed network framework and it consists of three main parts. On each client terminal, we have a 'AFN-Client' application who interact with local authentication sources (e.g. RFID reader, Bluetooth etc). This application sends authentication data to 'AFN-Fusion', which is part of the main session server. 'AFN-Fusion' is a fusion engine for all type of authentication data in the system, including context data. It output authentication decision to 'AFN-Server' application which is responsible for activating, suspending, locking and unlocking of a user session. All user session reside on session server and can be remotely accessed from any terminal. We can configure it to implement a specific authentication policy. For example, an authentication policy can specify that authentication should be granted only if user's RFID badge and his bluetooth mobile are both present and validated. Underlying detail on multi-factor technology can be found in Jonnson's co-authentication framework[8]. Similarly the detail about context data (i.e. user's location) and corresponding authentication

techniques can be found in 'PAISE' scheme and related experiment by Kirschmeyer, Hansens and Jensen[33].

'AFN-Server' is responsible for starting, stopping, locking, unlocking of a user session which can be remotely accessed from a terminal. The decision of AFN-Server depends on authentication decisions being received from AFN-Fusion module. For example, if a user is sufficiently authenticated on a terminal, his relevant session is automatically invoked at that terminal. When user departs, the terminal is locked and session is suspended back to session server.

On client side, the most natural and efficient way of authentication for nomadic user is proximity-based login technique, which allows users to be authenticated on a device simply by approaching it physically[11]. However specific techniques for proximity based login are not in the scope of this paper, but it appears obvious that the choice of proximity based login mechanism will have a strong impact on the overall security of the system. However, in this paper we are more interested in usability and security improvements that can be obtained through the usability of the security mechanisms. It was indicated earlier that most human-centric authentication techniques are not suitable for persistence and seamless active authentication. Although, there are plenty of device centric approaches, these again come with a risk of device being stolen or tampered with. The best solution could be to combine the benefit of each technique, which leads to a hybrid solution where multiple authentication techniques are merged to form a unified authentication mechanism, which is also known as multi-factor authentication.

For nomadic users we may use the Jury framework[8] for achieving multi-factor authentication in our proposed framework. It enables an easy integration of arbitrary many authentication techniques, including biometrics, knowledge based and device centric techniques. Moreover it's easy to integrate with wide range access control systems which work on either probabilistic or binary authentication result. For instance, we can combine RFID, Bluetooth and password based authentication along with machine learning algorithms with help of this framework. A typical authentication policy for this framework could be as follows:

1. When user enter in the nomadic environment first time, he should enter his password.
2. System automatically associate RFID chips present in his clothes, watch, shoes and badge, with user identity. Similarly an active Bluetooth from user mobile is an additional binding parameter. We can specify that whenever RFID badge along with one of additional binding parameter is present, system should consider it as an authenticated user.
3. Machine learning may be used to not automatically login on a system, when user just pass nearby. Also a mechanism should not automatically try to authenticate a user on a system, which a user normally does not use.

In our framework, authentication data used in fusion process and authentication data used by 'AFN-Server' to invoke corresponding sessions is independent. The mapping between these two types depends on security certificate of users. A user's security certificate binds a user to multiple authentication devices. This architecture also enables us for a user level delegation.

To demonstrate our authentication framework, we have implemented a scaled down version. This prototype consists of a Intel based desktop computer, running UBUNTU Linux(which is most popular distribution of Debian Linux[18]) with GNOME desktop. Both server and client part of authentication are present on single machine and communicate with each other using network sockets. For authentication purpose we have augmented classic login based mechanism with RFID based authentication mechanism, which represents a branch in multi-factor authentication.

Multiple user sessions are simultaneously present on computer. When a person wearing a RFID tag approaches the machine, his relevant session becomes active and displayed. Also this mechanism provides persistent authentication which means that user is authenticated as long as he is present on machine. If he depart from the scene authentication is automatically revoked in the system and relevant session is suspended which might be reactivated in future.

## 4. Analysis and Result

First of all, let's consider security vulnerabilities caused by usability and were described in previous

section. In our mechanism, user has to enter password once so there is no motivation for using short and easy passwords. Moreover, disclosure of password is not a complete system failure but in fact, it can be detected by our multi-factor authentication. We have used persistence authentication and thus user logout as soon as he departs from scene. As user does not have to enter password in normal flow of work so password sniffing by observation is also not possible. We support a nice user level delegation which is secure way of giving one's password to other person and also there is no need to have a group password. Our mechanism is persistent with very short term authentication memory which makes attack due to stealing of pre-authenticated session not possible. As authentication process is automated, it saves a user from wasting valuable time in authentication process. These facts suggest that our mechanism is able to address a large set of security vulnerabilities and thus contributes to increase the effective level of system security.

We have also analyzed implementation of our security mechanism by dividing it in three steps and then performing detailed security analysis at each step. Some underlying assumptions for our prototype, which are part of access control mechanism provided by Debian Linux, are as follows; Firstly, there is only one super user in server system. Secondly, 'AFN Fusion' and 'AFN Server' runs with privileges of this super user. Thirdly, their is a certain trust level on physical integrity of sensors, for instance RFID reader is not tampered. Fourthly, each terminal is running a trusted copy client application 'AFN Client' and Fifthly, interprocess communication using network socket is secure. These assumptions are quite reasonable for a Linux based network.

In the first step 'AFN fusion' receives all authentication information from individual sensors. For our prototype this means a list of identifications for all tag present in the field of RFID reader. Since we have assumed that hardware is not tampered with, thus this data present a valid input. One may argue that a tag might be cloned, but this can be avoided by using tamper-resistant tags e.g. Zero knowledge RFID as described by Engberg, Harning and Jensen[32]. Implicitly we are also assuming some basic conditions which are part of every device centric authentication, like RFID. This implies that tag presented for authentication is currently possessed by owner. Moreover a inherent deficit of confidence on any device centric mechanism is well represented during fusion where each authentication sensor has certain level of trust.

In the second step 'AFN fusion' is able to combine all authentication data it received. The only security vulnerability which is possible in this case is denial of service. For example tag is not scanned in previous interrogation. It is also possible, for example,if their are large number of tags present in field. To avoid this, 'AFN Server' samples authentication data at relatively high rate and operate on principle of moving average. Third step is correctly mapping authentication data to user session. This involves scanning the user's security certificates to find out the user for which output of 'AFN fusion' is valid. All these certificates are signed so they can't be tampered with and also user's don't have write access to them. 'AFN Server' will not activate a session on a particular terminal, if a valid 'AFN client' is not running their.

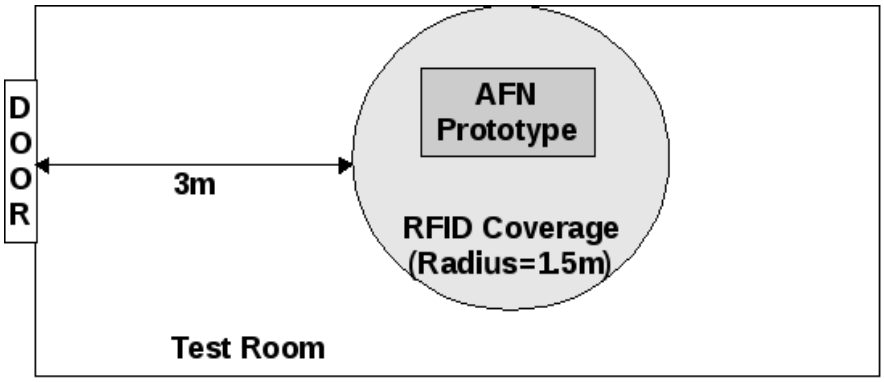


*Figure 3: Experimental setup*

To estimate usability performance let's define a benchmark to represents a typical nomadic environment. We assume a user login 12 time in an hour and their are 8 working hours per day. The physical infrastructure for our experiment is shown in the Figure 3.

Test results are shown in the Table-1, by averaging one hundred individual authentications for each case. Most notable is total authentication time in a whole day, which turns out to be just 1.8 minutes for our prototype. This is quite significant as compared to almost 20 minute time consumed by password based authentication. This is because an average user takes around 8 second to type a good complex password. We have experimented with passwords of length ten, and they all include alphabets, numeric and special character. This quality of password is roughly equivalent to 64-bit random identification stored in RFID tags we have used in the experiment. In other words, by using the prototype, one can have an extra person for every 24 persons which might be very attractive especially for those environments where price for a working hour is high, for instance in hospitals.

| **S/N** | **Feature** | **AFN prototype** | **Password based system** |
|---|---|---|---|
| 1 | Session Lock | Automatic | Manual |
| 2 | Session Unlock | Automatic and takes less than 1 second | Manual, On average take about 8 seconds for good passwords. |
| 3 | Switching User | Automatic and takes less than 1 second | Manual, On average take about 25 seconds, if system does not support simultaneous multiple sessions. In other case it takes 10 seconds. |
| 4 | Run-time System Resources | ~500MHz of Celeron D processor | Zero |
| 5 | New Session Creation Time | 10 seconds | 20 seconds |
| 6 | Daily time consumed in Authentication | 1.8 minutes | 20 minutes |

*Table 1: Performance Comparision*

# Conclusion

Unfortunately, the evolution towards nomadic use of computing is not accompanied by use of appropriate security mechanism. A password based authentication mechanism, when used by nomadic users, leads to many security vulnerabilities, causing a considerable drop in level of effective security. It turn out that most of these security vulnerabilities are due to usability constraints present in classic authentication mechanisms As nomadic users are very conscious about usability issues, so they try to circumvent security checks. Thus usability should be considered as an integral part of overall design of an authentication mechanism.

We have developed an authentication framework and have evaluated its prototype, from both usability and security perspectives. We have improved the quality of interaction between human and computer by minimizing usability constraints. This contributes to saving time which may result in more job satisfaction along with economic advantages for an organization. We have increased the level of effective security by addressing usability constraints related to authentication mechanisms. As we have shown in our security analysis we have removed most of those security vulnerabilities which are normally present in a system where user is actively involved in authentication process.

At last, we also confess that a true analysis of nomadic users require cross-disciplinary approach including anthropology and psychology along with a couple of clinical and field trials, in order to truly discover the best usability options that can be incorporated in any security mechanism.

# References


[1] Lin Hong, Anil K. Jain, and Sharath Pankanti, " Can multibiometrics improve performance", Technical Report MSU-CSE-99-39, Department of Computer Science, Michigan State University, 1999.

[2] Imran Naseem and Ajmal Mian, "User Verification by Combining Speech and Face Biometrics in Video", Advances in Visual Computing, ISBN 978-3-540-89645-6, Pg. 482-492, 2008.

[3] Sundararaman Jeyaraman and Umut Topkara , "Have the cake and eat it too – Infusing usability into text-password based authentication systems", Proceedings of the 21st ACSAC, Pg. 473 –482, 2005.

[4] D. Davis, F. Monrose and M. K. Reiter, "On User Choice in Graphical Password Schemes," In Proceedings of the 13th UNIX Security Symposium, August 2004.

[5] Nicholas J. Hopper and Manuel Blum, "A secure human computer authentication schemes", CMU-CS-00-139, School of Computer Science, Carneige Mellon University, May 2000.

[6] Cynthia Kuo, Sasha Romanosky and Lorrie Faith Cranor, "Human Selection of Mnemonic Phrase-based Passwords", ACM International Conference Proceeding Series Vol. 149, Pg. 67–78, 2006.

[7] Mark D. Corner and Brian D. Noble, "Zero-interaction authentication", Proceedings of the 8th annual international conference on Mobile computing and networking Atlanta, Georgia, Pg. 1–11, 2002.

[8] Einar Jonsson, "Co-Authentication - A Probabilistic Approach to Authentication", Master's thesis, IMM-Thesis-2007-83, Informatics and Mathematical Modeling, Technical University of Denmark, DTU, 2007.

[9] Bruce L. Riddle, Murray S. Miron, and Judith A. Semo, "Passwords in use in a university timesharing environment", Computers and Security Vol 8 (7), Pg. 569 – 578, November 1989.

[10] Daniel V. Klein, "Foiling the cracker: A survey of, and improvements to, password security", Proceedings of the second USENIX Workshop on Security, Pg. 5-14, July 1990.

[11] Jakob E. Bardram, Rasmus E. Kjær, and Michael Ø. Pedersen, "Context-Aware User Authentication: Supporting Proximity-Based Login in Pervasive", UbiComp 2003: Ubiquitous Computing, Pg. 107-123, 2003.

[12] Mark D. Corner, Brian D. Noble, "Protecting applications with transient authentication", Proceedings of the 1st international conference on Mobile systems, San Francisco, California, Pg. 57 – 70, 2003.

[13] F. Bennett, T. Richardson, and A. Harter, "Teleporting- Making Applications Mobile", Proceedings of the IEEE Workshop on Mobile Computer Systems and Applications, Pg. 82–84, 1994.

[14] B. Brumitt, B. Meyers, J. Krumm, A. Kern and S. Shafer, "EasyLiving: Technologies for Intelligent Environments", Handheld and Ubiquitous Computing, Pg. 97-119, 2000.

[15] A. Ward, A. Jones, and A. Hopper, "A new location technique for the active office", IEEE Personal Communications, Vol. 4(5), Pg. 42-47, October 1997.

[16] Daniel M. Russell and Rich Gossweiler, "On the Design of Personal & Communal Large Information Scale Appliances", Ubicomp 2001: Ubiquitous Computing, Pg. 354-361, January 01, 2001.

[17] Xyloc family of products, Ensure Technologies (Ypsilanti, Michigan) , <http://www.ensuretech.com>, Last visited March 24th, 2009.

[18] Ladislav Bodnar, "Top Ten Linux Distributions", <http://distrowatch.com/>, Last visited April 1st, 2009.

[19] Lawrence O'Gorman, "Comparing Passwords, Tokens, and Biometrics for User Authentication", Proceedings of the IEEE, Vol 91(12), Pg 2019-2040, 2003.

[20] K. Nagel, C. D. Kidd, O'Connell, T. O'Connell, A. Dey and G. D. Abowd, "The Family Intercom: Developing a Context-Aware Audio Communication System", Proceedings of UBICOMP, Pg. 176-183, 2001.

[21] R. Want, A. Hopper, V. Falco, and J. Gibbons, "The Active Badge Location System," ACM Transaction

on Information Systems, Vol 10(1), Pg. 91-102, January1992.

[22] Science News University of California, San Francisco. "Age-related Memory Loss Tied To Slip In Filtering Information Quickly." ScienceDaily dated 5 September 2008. <http://www.sciencedaily.com/releases/2008/09/080902143234.htm>, Last visited April 1st, 2009.

[23] Department of Defense, Trusted Computer System Evaluation Criteria dated 1985, <http://csrc.nist.gov/publications/history/dod85.pdf>, Last visited March 30th, 2009.

[24] Lawrence A. Tomei , "Encyclopedia of Information Technology Curriculum Integration", Information Science Reference; illustrated edition , ISBN-13: 978-1599048819, February 5, 2008.

[25] Mike Ebbers, Wayne O'Brien and Bill Ogden, "Introduction to the New Mainframe: z/OS Basics" dated July 2006, <http://publibz.boulder.ibm.com/zoslib/pdf/zosbasic.pdf>, last visited March 26th, 2009.

[27] Pam Snaith and Rob Steiskal, "Mainframes are still mainstream", White paper by CA Inc, June 2007. <www.ca.com>, Last visited March 30th, 2009.

[28] Mark Weasor, "Nomadic Issues in Ubiquitous Computing", Xerox PARC (Palo Alto Research Center), <http://www.ubiq.com/hypertext/weiser/NomadicInteractive> , last visited March 26th, 2009.

[29] Marcia Riley, "Ubiquitous Computing: An Interesting New Paradigm", <http://www.cc.gatech.edu/classes/cs6751_97_fall/projects/say-cheese/marcia/mfinal.html>,Last visited March 26th, 2009.

[30] J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege and D. Spence, "Network Working Group: RFC 2904", August 2000.

[31] Charles P. Pfleeger and Shari Lawrence Pfleeger, "Security in Computing", Prentice Hall Professional Technical Reference, 2002.

[32] Stephan J. Engberg, Morten B. Harning and Christian Damsgaard Jensen, "Zero-knowledge Device Authentication:Privacy & Security Enhanced RFID preserving Business Value and Consumer Convenience", Proceedings of the 2nd Annual Conference on Privacy, Security and Trust (PST'04), 2004

[33] Martin Kirschmeyer, Mads S. Hansen and Christian D. Jensen, "Persistent Authentication in Smart Environments", 2nd International Workshop on Combining Context with Trust, Security and Privacy. Trondheim, Norway, 2008.

[34] J. Bardram, T. Kjær and C. Nielsen, "Mobility in Healthcare - Reporting on our initial Observations and Pilot Study",Technical report of a clinical study, CfPC 2003-PB-52, Center for Pervasive Computing, 2003.

[35] Jens Bæk Jørgensen and Claus Bossen, "Executable Use Cases for Pervasive Healthcare", IEEE Software Volume 21 , Issue 2, Pg. 34 – 41, ISSN:0740-7459, March 2004.

[36] Jakob Bardram, "The trouble with login: on usability and computer security in ubiquitous computing", Personal and Ubiquitous Computing Vol9(6), Pg. 357–367, ISSN:1617-4909, November 2005

[37] Rachna Dhamija and Adrian Perrig, "Deja Vu: A user study using images for authentication", In the Proceedings of the 9th USENIX Security Symposium, Denver, Colorado, August 2000.

[38] I. Jermyn, A. Mayer, F. Monrose, M. Reiter and A. Rubin. "The Design and Analysis of Graphical Passwords", Proceedings of the 8th UNIX Security Symposium, August 1999.

[39] Matt Bishop, "Computer Security: Art and Science" , book published by Addison-Wesley Professional, ISBN-13: 978-0201440997, 2002.

[40] Computer Industry Almanac, "25-Year PC Anniversary Statistics", Press release August-2006, <http://www.c-i-a.com/pr0806.htm>, Last visited April 1st, 2009.

[41] Password Research, "Authentication Statistic Index" maintained by Bruce K. Marshall, <http://passwordresearch.com/stats/statindex.html>, Last visited April 1st, 2009.