2012 Fourth International Conference on Digital Home

Combining Mouse and Keystroke Dynamics Biometrics for Risk-Based Authentication in Web Environments

Issa Traore, *Member IEEE*, Isaac Woungang, *Member IEEE*, *Mohammad S. Obaidat, *Fellow of IEEE*,
[1]Youssef Nakkabi, and [2]Iris Lai
Ryerson University, Canada, Monmouth University, USA and Khalifa University, UAE
*Monmouth University, NJ, USA and Khalifa University, UAE

***Abstract*—Existing risk-based authentication systems rely on basic web communication information such as the source IP address or the velocity of transactions performed by a specific account, or originating from a certain IP address. Such information can easily be spoofed, and as such, put in question the robustness and reliability of the proposed systems. In this paper, we propose a new online risk-based authentication system that provides more robust user identity information by combining mouse dynamics and keystroke dynamics biometrics in a multimodal framework. Experimental evaluation of our proposed model with 24 participants yields an Equal Error Rate of 8.21%, which is promising considering that we are dealing with free text and free mouse movements, and the fact that many web sessions tend to be very short.**

***Index Terms*—Risk-based authentication, network security, mouse dynamics, keystroke dynamics, biometric technology, Bayesian network model.**

I. Introduction

Risk-based authentication consists of using both contextual information (e.g. computer mapping, IP address, IP geo-location) and historical information (e.g. user's transaction patterns), along with data provided during Internet communications to determine the probability of whether or not a user interaction is genuine [1-4]. However, much of the historical and contextual data used in existing risk-based authentication systems may be subject to fraud [2]. For instance, a criminal could poll a user's computer and then replicate the settings on a different machine with the intention of fooling the authentication system.

Furthermore, existing systems use ad hoc or simplistic risk management models along with rule-based techniques for user behavioral and transactional patterns analysis. These systems are operationally ineffective due to the rapidly evolving nature of online fraud patterns as well as the fact that new and unseen fraud cases are emerging frequently. Such cases cannot be completely covered by static rules from a rule based system.

In this work, we propose a new risk-based authentication system that uses a combination of mouse dynamics biometrics and keystroke dynamics biometrics data captured in online sessions.

A key characteristic of risk-based authentication is that the process must be virtually invisible to end users. This means that in our model, mouse and keystroke dynamics must be captured and processed *freely.*

Although keystroke dynamics biometric has been studied extensively and used for authentication since the early 1980's, most of the existing proposals have focused primarily on fixed text recognition [5-6]. Fixed text recognition consists of enrolling the user using a predefined text or phrase, and performing the detection by asking the user to type exactly the same string. While fixed text recognition may be used in static authentication (i.e. login), it is not appropriate in risk-based authentication, where the user must be authenticated in a non-intrusive way. Under such scenario, the user must be authenticated based on text freely typed, which does not necessarily match the enrolment sample. This is referred to as free text detection [7]. Free text detection in web environments is very challenging because of the limited amount of keystrokes involved in many web sessions (e.g. online banking). Similar challenges are involved in free mouse dynamics biometric analysis [8-9].

In this work, we tackle the above challenges by developing an online risk-based authentication scheme that integrates mouse dynamics and free text analysis using a Bayesian network model. Due to the limited amount of mouse actions or keystrokes involved in a typical web session, relatively lower performance is achieved when using each of the individual modalities in isolation in our Bayesian network model. However, by combining the two modalities, the overall performance is improved significantly. The performance of the proposed scheme is computed by measuring the False Acceptance Rate (FAR), the False Rejection Rate (FRR), and the Equal Error Rate (where FAR=FRR)..

The experimental evaluation of the proposed multimodal framework with 24 participants yields an EER of 8.21%.

The rest of the paper is organized as follows. Section 2 provides a summary of the related work. Section 3 introduces our proposed approach and model. Section 4 describes the experimental evaluation of our approach. Section 5 makes some concluding remarks.

II. Related Work

A. *On Risk-based Authentication*

To our knowledge, most of the published proposals in the research literature on risk-based authentication have been at the transactional level [1, 10].

In this context, Dimmock *et al.* [10] introduced a computational risk assessment technique for dynamic and flexible access decision-making. The proposed approach allows basing access control decisions on risk and trust rather than on credentials only. However, the approach assumes a prior knowledge of outcomes of all possible combinations of

978-0-7695-4899-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICDH.2012.59

states and actions during the decision-making process, which is not realistic. Furthermore, the subjects in their model are autonomous agents, not humans; and we know that human identity and behavior are essential aspects of risk-based authentication.

Diep *et al.* [1] presented a framework for contextual risk-based access control for ubiquitous computing environments. In their scheme, access control decisions are made by assessing the risk based on the contextual information, the value of requested resources or services, and the consequences of unauthorized system transactions. A mathematical risk assessment and scoring technique called *Multifactor Evaluation Process (MFEP)* is proposed, in which numerical weights are assigned to risks factors in terms of confidentiality, integrity, and availability of the related outcomes.

Cheng *et al.* [11] proposed a new risk based access control model named *Quantified Risk Adaptive Access Control (QRAAC)*) that uses a quantitative risk assessment technique based on fuzzy logic. *QRAAC* replaces the traditional binary decision-making used in access control scheme with a dynamic, multi–decision access control based on objective risk measures. The scale of the quantified risk estimates is divided into several risk ranges, each associated with a specific decision and action assigned according to the risk tolerances.

B. *On Keystroke Dynamics*

Most existing literature on keystroke dynamics focused on fixed text detection [7, 13]. Limited amount of works has been accomplished on free text detection. Representative ones are captured in [7, 14, 17].

Monrose and Rubin [17] proposed an approach for free text analysis of keystrokes using clustering based on a variation of the *maximin-distance* algorithm. By collecting over a period of 7 weeks the typing samples from 42 users performing structured and unstructured tasks in various computing environments, 90% correct classification was obtained for fixed text detection while only 23% correct classification was achieved for free text detection.

Dowland *et al.* [14] collected keystroke dynamics samples by monitoring the users during their regular computing activities without any particular constraints imposed on them [15]. A user profile was determined by calculating the mean and standard deviation of the digraph latency and by considering only the digraphs that occur a minimum number of times across the collected typing samples. An experimental evaluation of their scheme with five participants yielded correct acceptance rates in the range of 60%.

Gunetti and Picardi [7] introduced and used a metric based on the degree of disorder of an array for free text analysis of keystrokes. An experimental evaluation of their scheme involved 40 users considered as legal users who provided 15 typing samples each, and 165 users considered as imposters who provided one sample each, by entering their login identifier and then freely typing some text through a web-based interface. Overall, a FAR of 0.00489%, a FRR of 4.8333%, and an EER in the range 0.5-1% were achieved. It must be noted that although the performance obtained in this case is excellent, the average length of the samples varied from 700 to 900 characters. Determining what the performance of the proposed scheme would be with session size as small as 100 characters (which is typical in web environments) is still an open problem.

C. *On Mouse Dynamics*

The last decade has witnessed an increasing interest in mouse dynamics biometric research [18-20]. Next, we discuss some of these proposals.

The Mouse-lock system proposed by Revett et al. [18] uses mouse dynamics biometric for static authentication (at the login time). The user interface consists of thumbnail images oriented in a circle. By using the mouse, the user enters a password by clicking 5 of the displayed images in the correct sequence similar to the rotary lock. An experimental evaluation of the proposed scheme involving 6 users providing 100 genuine samples each and 20 impostor samples yielded FAR and FRR between 2% to 5%.

Bours and Fullu [19] proposed a login system using mouse dynamics based on a specially designed graphical interface in the form of a maze. To log in, users have to move the cursor by following the paths. An experimental evaluation of the proposed approach involving 28 participants achieved an EER ranging between 26.8% and 29%.

Ahmed and Traore [20] proposed a mouse dynamics biometric recognition approach in which 39 mouse dynamics features were extracted and analyzed using neural networks. The proposed approach was evaluated by conducting various experiments involving both free and fixed mouse movements. An overall FAR of 2.4649% and FRR of 2.4614% were obtained in the main experiment with 22 participants, in which tests were conducted on various hardware and software systems. Seven test users participated in a second experiment providing 49 sessions. In this experiment, the same hardware and software applications were used. The test results consisted of FAR and FRR of 1.25% and 6.25%, respectively. A third experiment was limited to the same machine and while the previous 7 participants were asked to use the same application and perform the same predefined set of actions. FAR and FRR of 2.245% and 0.898%, respectively, were obtained in this experiment.

Like keystroke dynamics, most of the existing works on mouse dynamics target predefined or fixed mouse movement. Under this category fall the *MouseLock* system introduced by Revett et al. [18], and the maze-based login scheme by Bours and Fullu [19] as outlined above. In the few works where free mouse movements were studied, like in [20], the main challenge has been the degradation of the accuracy as the session length decreases. Such challenge must be addressed in online environments where small sessions are common.

III. Approach and Model

A. *Approach Overview*

As mentioned above, we consider in our work two different biometric modalities, namely, mouse and keystroke dynamics. We extract a separate model of the user behavior for each of the individual biometric modalities using a Bayesian network [21]. The global user profile is obtained by fusing the outcome of the individual Bayesian networks using the Bayesian fusion scheme explained in the next section.

To enroll a user, we extract for each separate modality a specific Bayesian network model from the enrollment sample. The extracted Bayesian network structure is stored as part of the user profile. Hence, each user will have different Bayesian network structures for each of the two modalities.

To authenticate a user, we apply the monitored sample to the Bayesian network structures corresponding to the profile for the claimed identity and compute the individual biometric scores. The individual scores are then fused giving the global matching score, which is compared to a threshold for decision making purpose.

B. Bayesian Fusion

Let n denote the number of different modalities involved in our multimodal framework, and let s_i denote the matching score between a monitored sample and an individual profile for modality $i\,(1 \leq i \leq n)$. We estimate the a posteriori probabilities using Bayes rule as follows:

$$P(G|s_1, \dots s_n) = \frac{P(s_1, \dots, s_n|G)P(G)}{P(s_1, \dots, s_n|G)P(G) + P(s_1, \dots, s_n|I)P(I)}$$

where *P(I)* and *P(G)* are the prior probabilities of the impostor and genuine classes, respectively; and $P(s_1, \dots, s_n|I)$ and $P(s_1, \dots, s_n|G)$ are the conditional probabilities of the impostor and genuine classes given score $s_i\,(1 \leq i \leq n)$. Assuming a conditional independence between the matching scores for the different modalities, given *G* or *I*, we obtain the following:

$$P(G|s_1, \dots s_n) = \frac{P(G)\prod_{i=1}^{n} P(s_i|G)}{P(G)\prod_{i=1}^{n} P(s_i|G) + P(I)\prod_{i=1}^{n} P(s_i|I)}$$

Often, the prior probabilities of individual modalities *P(x)* are unknown. It is customary in this case to assume that all concepts are equally likely, which gives the following:

$$P(G|s_1, \dots, s_n) = \frac{\prod_{i=1}^{n} P(s_i|G)}{\prod_{i=1}^{n} P(s_i|G) + \prod_{i=1}^{n} P(s_i|I)}$$

This can also be expressed as:

$$P(G|s_1, \dots s_n) = \frac{\prod_{i=1}^{n} P(s_i|G)}{\prod_{i=1}^{n} P(s_i|G) + \prod_{i=1}^{n}(1 - P(s_i|G))}$$

The above represents the Bayesian fusion score for n different modalities.

C. Keystroke Dynamics Model

Keystroke dynamics biometric analysis consists of extracting unique behavioral patterns from how a user types on a keyboard. Two main types of information are usually extracted from the keystrokes, namely, the dwell time and the flight time. The dwell time (also called the hold time) is the time between pressing a key and releasing it. The flight time (also called inter-key time) is the time between pressing two consecutive keys. The dwell and flight times can be used to compute the times associated with monograph (i.e. dwell time), digraph (i.e. flight time) or *n*-gram (in general).

In this work, keystrokes are divided into four categories based on the character ASCII codes and the mechanical keyboard layout[1]: *Upper Case Keystrokes*, *Lower Case Keystrokes*, *Control Keystrokes*, and *Other Keystrokes*. Keystrokes print characters such as "%", "&" and capitalized letters are categorized as Upper Case Keystrokes. The reason is that users must either press Caps lock key ahead or press Shift key at the same time to print these characters. All Upper Case Keystroke characters are listed in Table 1.

TABLE 1: UPPER CASE KEYSTROKE CHARACTERS

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z	!	‶	#	$	%	&	(
)	*	+	:	<	>	?	@	^	_	{
}	|	~								

Lower Case Keystrokes allow printing lower case letters on the computer screen; characters from "a" to "z" fall under this category. Control Keystrokes do not result in printing characters. Examples of Control Keystrokes are tab key, back space key, and delete keys. The remaining keystrokes are grouped into the Other Keystrokes category. For each category of keystrokes, we calculate the mean and standard deviation of the dwell times as well as the distribution of each type of keystrokes within a sequence of keystrokes. The extracted keystroke dynamics features are listed in Table 2.

We extract the keystroke dynamics features by processing batches of consecutive keystrokes. By default, we consider batches of size *n* = 10. From each batch, we extract a feature vector consisting of 20 features organized under 11 keystroke dynamics factors listed in Table 2. Each factor is represented by one or several features.

Hence, each factor can be considered as a separate feature vector. The concatenation of these individual feature vectors yields our global feature space for keystroke dynamics. Every session consists of a number of keystroke dynamics feature vectors or records. Every record corresponds to a sequence of *n = 10* consecutive keystrokes.

For the mean and standard deviation of the flight time feature vectors M_FT and SD_FT mentioned in Table 2, we only consider the down – down (DD) flight times and the release – down (RD) flight times. Each of these categories yields a separate feature.

For the percentage of occurrences per keystroke category (i.e. PER_TP feature vector), we consider all four categories.

When a user is typing a capitalized letter, he/she might be holding the Shift key and the letter key at the same time. For this type of behaviour where the user is holding multiple keys, we calculate the percentage of occurrences within a sequence of keystrokes. This is represented as the PER_MUL feature in Table 2.We compute the means for two different types of flight times based on the mechanical keyboard layout and the user behaviour in typing consecutive keys. The first type of flight time is the flight time corresponding to when both consecutive keys belong to the Upper Case Keystrokes

[1] In this work, we consider the most popular keyboard layout, which is the United States keyboard layout for Windows, Mac OS, and Linux.

category, or neither of the consecutive keys is from the Upper Case Keystrokes category.

TABLE 2: KEYSTROKE DYNAMICS BIOMETRICS FEATURES

Factor	Acronym	Unit	Number of Features	Description
Mean of dwell time	M_DT	Second	1	The mean dwell time of a sequence of keystrokes.
Mean of flight time	M_FT	Second	2	The mean flight time of a sequence of keystrokes.
Mean of trigraph Time	M_TRIT	Second	1	The mean trigraph time of a sequence of keystrokes.
Standard deviation of dwell time	SD_DT	Second	1	The standard deviation of dwell time of a sequence of keystrokes.
Standard deviation of flight time	SD_FT	Second	2	The standard deviation of flight time of a sequence of keystrokes.
Standard deviation of trigraph time	SD_TRIT	Second	1	The standard deviation of trigraph time of a sequence of keystrokes.
Mean of dwell time per category	M_DTTP	Second	4	The mean of dwell time for each keystroke category in a sequence of keystrokes.
Percentage of occurrences per category	PER_TP	%	4	The distribution of each keystroke category in a sequence of keystrokes.
Percentage of occurrences of holding multiple keys	PER_MUL	%	1	The percentage of occurrences of holding multiple keys in a sequence of keystrokes.
Average Typing Speed	ATS	Character / Second	1	The average typing speed of a sequence of keystrokes.
Mean of flight times per type of user behaviour	M_FTTP	Second	2	The mean of flight time for each type of user keystroke behaviour.

The second type of flight time is when only one key out of two belongs to the Upper Case Keystrokes category (i.e. while the other keystroke does not). These two types of means are represented as the M_FTTP feature vector in Table 2.

D. Mouse Dynamics

Mouse dynamics biometric analysis consists of extracting unique behavioural characteristics for a user based on his mouse actions, which consist of mouse movements and mouse clicks.

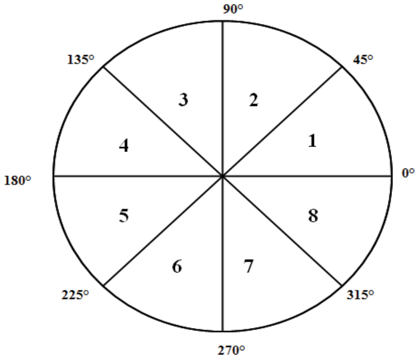


Figure 1. Mouse movement directions.

The raw mouse data consist of mouse movement coordinate, movement angle, the time to move the mouse from one location to the other, and the time of mouse clicks. As proposed in [20], the mouse movement directions can be divided into eight areas of 45° each as shown in Figure 1. Mouse features are extracted from batches of consecutive mouse actions. By default, we consider batches of size $n = 30$. We extract 66 features from the raw data organized under the 10 factors listed in Table 3. Each factor is represented by a separate feature vector consisting of one or several features. The concatenation of these individual feature vectors correspond to a 66-dimentional feature vector or record. Likewise, each session consists of several mouse records, each corresponding to ($n = 30$) consecutive mouse actions.

In Table 3, the factors PER_MAD, PER_DD, PER_MTD, ADD, ASD, AVXD, AVYD, and ATVD are calculated for each of the 8 movement directions (identified above). As a result, each of these factors is represented by eight feature values, each corresponding to a separate direction.

A silence occurrence is identified when the mouse move distance is 0. The percentage of silence occurrence in a sequence of mouse actions is represented by the feature SR.

E. Data Analysis

After extracting the features, noise reduction is performed on keystroke dynamics biometric features and mouse dynamics biometric features as explained in the following.

As mentioned above, flight time is defined as the time between two consecutive keystrokes. For instance, the flight time (down-down) is the time between pressing the first key and pressing the second key. First, the flight time value must be positive since the keys are pressed consecutively.

Second, by analyzing sample collected raw data for 5 random users (from our experiment), we observed that only a small amount of samples involve flight times beyond 3 seconds.

The percentage of records with flight time (down-down) greater than 3 seconds is 5.34% in the corresponding keystroke sample. As a result, we applied a filter by removing the data that is outside the range [0, 3 seconds].

TABLE 3: MOUSE DYNAMICS BIOMETRIC FEATURES

Factor	Acronym	Unit	Number of Features	Description
Average click time	ACT	Second	1	The average of mouse clicks time.
Silence ratio	SR	%	1	The percentage of silence occurrence of a sequence of mouse actions.
Percentage of mouse action per mouse movement direction	PER_MAD	%	8	The percentage of mouse action occurrence of a sequence of mouse actions in each mouse move direction.
Percentage of distance per mouse movement direction	PER_DD	%	8	The percentage of mouse move distance of a sequence of mouse actions in each mouse move direction.
Percentage of mouse move time per mouse movement direction	PER_MTD	%	8	The percentage of mouse move time of a sequence of mouse actions in each mouse move direction.
Average distance per mouse movement direction	ADD	Pixel	8	The average distance in each mouse movement direction.
Average speed per mouse movement direction	ASD	Pixel / Second	8	The average speed in each mouse movement direction.
Average velocity in X axis per mouse movement direction	AVXD	Pixel / Second	8	The average velocity in X axis in each mouse movement direction.
Average velocity in Y axis per mouse movement direction	AVYD	Pixel / Second	8	The average velocity in Y axis in each mouse movement direction.
Average tangential velocity per mouse movement direction	ATVD	Pixel / Second	8	The average tangential velocity in each mouse movement direction.

For mouse dynamics data, we apply two types of filters. First, we apply the moving average filter on the captured mouse move position data: computer screen x-y coordinates, and remove noise data on mouse dynamics feature values.

Using the moving average, we take the means of 5 points as the new values of the center point. We apply the moving average filter on *x*-coordinates and *y*-coordinates separately. The second filter applied on mouse dynamics data is similar to the filter applied on keystroke dynamics. The mouse dynamics data considered are mouse move time and speed. In the mouse dynamics data set collected for the above sample users, we observed the percentage of data with mouse move time less than 1.5 seconds and mouse move speed less than 5,000 pixels per second to be 97.44%. Therefore, we filter out as noise mouse move time and mouse move speed falling out of the range [0, 1.5 seconds] and [0, 5000 pixels], respectively.

As discussed earlier, we learn a Bayesian network to build the user profile, and then use it to classify the monitored samples. In the Bayesian network learning step, it is assumed that variables are discrete and finite. Since most of the features extracted from keystrokes and mouse actions are continuous, we convert them into nominal features through data discretization..

IV. FRAMEWORK EVALUATION

A. *Experimental Method and Setup*

The experimental evaluation was conducted on a simpler version of a social network website through which users share personal information such as family events or photos with friends. The main page is a normal user log on page, using user name and password authentication. After accessing an account, the website allows a user to perform the following actions: Post status, Post pictures, Add comments on account owner's pictures, Browse friends list, Add comments on friends' statuses, and Add comments on friends' pictures.

Each user was required to log in the web site as themselves (genuine user) or other users (intruder). For each login session, the logging type was recorded. Users have access to the usernames and passwords of other users, in order to allow impersonation attacks.

The architecture of the website consists of a web server and a database server. The web server and the database server were set up on a computer with Dual CPU 3.2 GHz and memory 2.00 GB RAM in our lab. The server was Windows Server 2008. The database server was MySQL server. Users used their own desktops, laptops or handheld devices (i.e. iPhone) to access the website. Requests originated from four different geographic locations, including Victoria (British Columbia, Canada), Toronto (Ontario, Canada), Oslo (Norway), and Shanghai (China).

B. *Collected Data*

In total, 24 users with different background and computer skills participated in the experiment. The experiment lasted for about 8 weeks. In total, 193 legitimate visits and 101 intrusive visits were contributed by the test users. Table 4 provides a breakdown of the sample keystroke and mouse data collected.

In Table 4, the data collected under each of the different modalities (keystroke, mouse actions) were grouped by sessions. Here, a session corresponds to a regular login session which spans from the time the user logs in to when he/she logs out. The samples collected within one session were grouped by a unique session ID.

In the experiment, every test user contributed different numbers of sessions. Although the total numbers of samples provided by test users were high, the number of samples for each session was low. For example, 62.6% of genuine keystroke sessions contain less than 100 keystrokes.

TABLE 4: COLLECTED SAMPLES STATISTICS

		Keystrokes		Mouse Actions	
		Genuine Data	Attack Data	Genuine Data	Attack Data
Number of Samples per user	Minimum	11	42	303	371
	Maximum	6,337	1,268	84,503	22,093
	Average	1,264	417	19,007	6,205
Total number of samples (all users)		22,585	6,696	344,005	103,620

C. Evaluation Method

For each of the users, a reference profile was generated for each individual modality based on a training set consisting of positive and negative records. Only genuine records were used in the training sets. Genuine data is divided into enrolment data and test data based on the timeline. The earliest data (received in time) was used for enrolment while the data collected subsequently was used for testing. For each of the legal users, the positive records in the training set consisted of genuine enrolment samples for that user while the negative training records consisted of enrolment data from other randomly selected legal users. The numbers of selected users (for training per user) varied for each modality: the number of selected legal users for keystroke dynamics was 8 whereas the number of selected users for mouse dynamics was 4.

By analyzing the sample data, we found that the minimum number of positive records required to effectively train the Bayesian network for each of the different modalities varies; the minimum number of keystroke dynamics records was 200 and the minimum number of mouse dynamics records was 2,500.

To test for false rejection, for each of the genuine users, we compared one-by-one the rest of their genuine sessions (not involved in building their profiles) against their own profile. The FRR for each of the genuine user was obtained as the ratio between the number of false rejections and the total number of trials. The overall FRR was obtained as the average of the individual FRRs obtained for all the genuine users.

To test for false acceptance, for each of the genuine users, we compared one-by-one against his/her profile the attack sessions generated for this user. The individual FAR was computed for each user as the ratio between the number of false acceptances and the number of test trials. The overall FAR was computed as the average of the individual FAR over all the genuine users.

D. User Enrollment and Verification

We used stratified 10-fold cross validation to train the Bayesian network corresponding to a user profile. The validation steps are briefly explained in the following. First, randomize the records and divide them into 10 equal size subsets (or 10 folds). Each fold has similar class distribution.

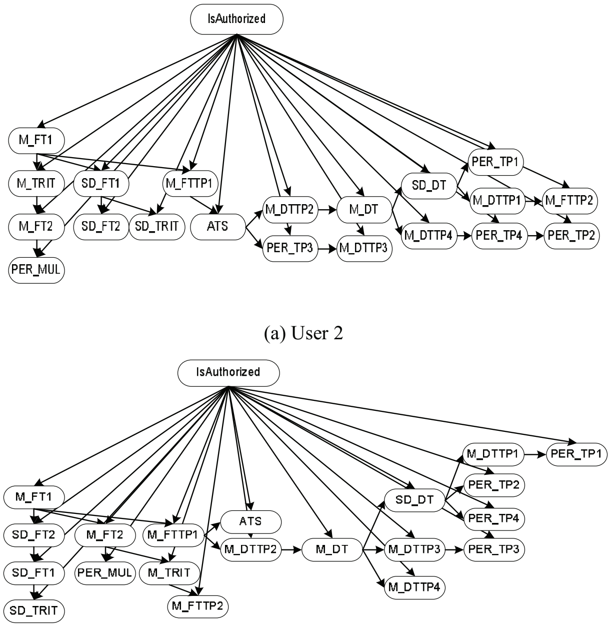
TABLE 5: BAYESIAN NETWORK TRAINING RECORDS AND VALIDATION RESULTS FOR LEGAL USERS (PCCR STANDS FOR THE PERCENTAGE OF CORRECTLY CLASSIFIED RECORDS).

		Positive Training		Negative Training		PCCR (%)
		Number of Sessions	Number of Records	Number of Sessions	Number of Records	
User 2	Keystroke Dynamics	2	579	32	2,235	91.61
	Mouse Dynamics	3	8,407	14	35,889	95.70
User 3	Keystroke Dynamics	3	597	34	1,711	97.44
	Mouse Dynamics	4	9,632	28	28,985	92.57
User 5	Keystroke Dynamics	8	395	29	1,470	93.03
	Mouse Dynamics	3	4,900	12	32,682	97.85
User 7	Keystroke Dynamics	2	671	17	1,785	90.35
	Mouse Dynamics	3	6,809	11	32,222	98.91
User 10	Keystroke Dynamics	11	369	26	1,876	92.69
	Mouse Dynamics	10	2,794	7	20,893	98.43
User 11	Keystroke Dynamics	2	474	45	2,611	97.21
	Mouse Dynamics	2	6,761	12	33,066	96.42
User 12	Keystroke Dynamics	3	237	26	2,570	95.48
	Mouse Dynamics	5	3,963	20	26,294	98.41
User 18	Keystroke Dynamics	2	215	30	2,331	96.11
	Mouse Dynamics	3	7,772	12	21,004	98.83
User 20	Keystroke Dynamics	11	199	21	1,845	94.23
	Mouse Dynamics	3	6,128	21	23,703	94.27
User21	Keystroke Dynamics	4	271	30	2217	95.70
	Mouse Dynamics	5	6,096	23	33,313	98.64
User22	Keystroke Dynamics	3	512	34	2,771	94.15
	Mouse Dynamics	2	9,853	21	24,178	93.60
User 24	Keystroke Dynamics	4	44	33	2,382	99.22

	Mouse Dynamics	4	5,155	14	29,220	96.84

This type of validation is called stratified validation. Second, repeat the run tests for 10 times. In each round i ($1 \le i \le 10$), the i^{th} subset is removed from the training set and is used as a test set. We then obtain the correctly classified records for 10 tests. The correctly classified records are the records whose predicted class probability is over 50%. The total number of correctly classified records is the sum for 10 tests. The percentage of correctly classified records (PCCR) is the total number of correctly classified records divided by the total number of records. Table 5 lists the numbers of sessions and records in each training set and the PCCR corresponding to legal users' profiles.

Figure 2 displays the trained keystroke Bayesian Networks for two different users; User 2 and User 7. In the verification stage, we apply sample keystroke dynamics records on the Bayesian network profile of a given user to compute the probability that the records were generated by the user.



(a) User 2

(b) User 7

Figure 2: Keystroke Bayesian Network for two different users: User 2 and User 7

E. Evaluation Results

Keystrokes and mouse actions were collected from a total of 24 test users. Due to limited numbers of sessions contributed by some test users, the total number of users having enough data for enrolment was 12. These users represent our legal users in calculating the FAR and FRR. We evaluated the performance of each individual modality for each user separately. Averaging users' FRR and FAR yielded the overall FRR and FAR for that modality. Figure 3 shows the Receiver Operating Characteristic (ROC) curves corresponding to the individual modalities. From this figure, it can be observed that the mouse dynamics model has slightly lower equal error rate (EER) at 22.41%, while keystroke dynamics yields an EER at 24.78%.

Figure 4 shows the ROC curve corresponding to the fusion of both modalities. The overall EER is 8.21%. This is lower than either the EER of keystroke dynamics or the EER of mouse dynamics, and can be considered as very encouraging considering the limited amount of free text and free mouse movements available in many web sessions.

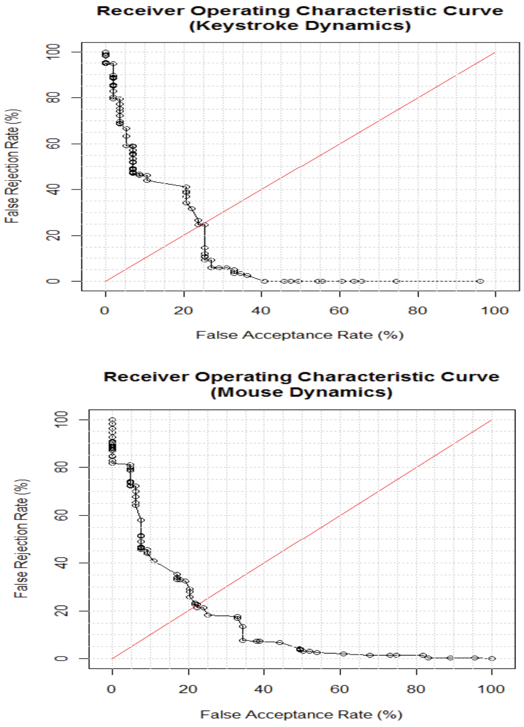


Figure 3: ROC Curves for Individual Modalities

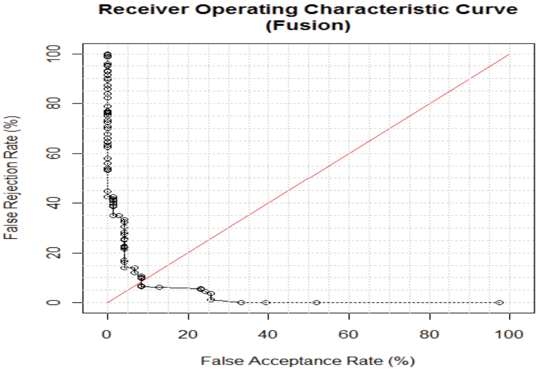


Figure 4: ROC Curve for Mouse dynamics and Keystroke dynamics fusion

V. Conclusion

This paper introduces a new risk-based authentication system for web environments that combines mouse and keystroke dynamics biometrics using Bayesian network models. Web environments are characterized by the limited amount of keystrokes and mouse actions involved in many sessions. This makes detection hard, especially for free samples produced without any predefined baseline. The proposed approach achieves an EER of 8.21%, which is encouraging. Risk-based authentication can be applied from two different perspectives: proactively and reactively. When applied proactively, risk-based authentication can be integrated with the login process and used to block from the beginning access to users flagged as risky. In contrast, reactive risk-based authentication can be used to identify and revert ongoing or completed transactions considered as risky.

Although proactive risk-based authentication may be considered as more desirable than reactive risk-based authentication, the cost of a misclassification error is far greater in the former than in the latter. In other words, more stringent accuracy requirements underlie proactive approaches compared to reactive ones.

Actually, each category is adequate for specific scenarios. While proactive risk based authentication is important in situations where confidentiality is essential such as in military or intelligence transactions, reactive risk-based authentication may be enough in situations where integrity is the primary concern. For instance, in online banking transactions, malicious transactions (e.g. illegal transfer between accounts) can be reverted (immediately) by the end of the session if the user is classified as risky.

As shown above, the experimental evaluation of our proposed risk-based authentication scheme yields an EER of 8.21%. Although such performance can be considered relatively low for proactive risk-based authentication, we believe that it is adequate for reactive risk-based authentication. In this case, the goal is not to prevent the user from using the system, but rather to identify malicious sessions and trigger appropriate risk mitigation measures.

In our future work, we will focus on improving the performance of our proposed system by studying alternative machine learning techniques such as neural networks and artificial immune systems. We will also expand our experimental dataset by involving more participants.

REFERENCES

[1] Diep N. N., S. Lee, Y.-K. Lee, H.J. Lee, "Contextual Risk-based Access Control", *Security and Management*, pp. 406-412, 2007.

[2] Tubin G., "Emergence of Risk-Based Authentication in Online Financial Services: You Can't Hide Your Lyin' IPs", *Whitepaper #V43:15N, TowerGroup,* May 2005.

[3] Obaidat M.S. and Macchairllo D. T. , "An On-line Neural Network System for Computer Access Security", IEEE Transactions on Industrial Electronics, Vol. 40, No.2, pp.235-242, April 1993.

[4] Enokido, T.; Takizawa, M., "Purpose-Based Information Flow Control for Cyber Engineering", IEEE Transactions on Industrial Electronics, Vol. 58, No.6, pp.2216-22225, June 2011.

[5] Bergadano, F., Gunetti, D., and Picardi C., "User Authentication through Keystroke Dynamics", *ACM Transactions on Information and System Security*, Vol. 5, No. 4, Nov. 2002, pp. 367-397.

[6] Obaidat, M.S., Sadoun, B., "Verification of Computer Users Using Keystroke Dynamics", *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, Vol. 27, No. 2, pp. 261-269, 1997.

[7] Gunetti D., and C. Picardi, "Keystroke Analysis of Free Text", *ACM Transactions on Information and System Security,* Vol. 8, No. 3, Aug., pp. 312-347, 2005.

[8] Aksarı, Y. and Artuner, H., "Active Authentication by Mouse Movements", *In Proc. of the IEEE 24th Intl. Symposium on Computer and Information Sciences (ISCIS 2009)*, Metu, Northern Cyprus, Sept. 2009, pp.571-574.

[9] Bours, P., Fullu, C.J., "A Login System Using Mouse Dynamics", *In Proc. of the 5thIntl. Conference onIntelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2009)*, Kyoto, Japan, Sept. 12-14, 2009.

[10] Dimmock N., Bacon J., Ingram D., and Moody K., "Risk Models for Trust–Based Access Control", *In Proc. of 3rd Annual Conference on Trust Management (iTrust 2005)*, *Series LNCS*, Vol. 3477, Springer, May 2005, 426 pages.

[11] Cheng P.-C., P. Rohatgi, C. Keser, P. A. Karger, G. M. Wagner, and A. S. Reninger, "Fuzzy Multi–Level Security: An Experiment on Quantified Risk–Adaptive Access Control", *IBM Research Report RC24190,* 2007

[12] Jiang C.-H., Shieh S., and Liu J.-C. 2007, "Keystroke Statistical Learning Model for Web Authentication", *In Proc. of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS'07),* Singapore, March 2007, pp. 359–361.

[13] Legget, J, Williams, G., "Dynamic Identity Verification via Keystroke Characteristics", *International Journal on Man-Machine. Studies*, Vol. 35, pp. 859-870, 1988.

[14] Dowland, P., Furnell, S., and Papadaki, M., "Keystroke Analysis as a Method of Advanced User Authentication and Response", *In Proc. of the 17th Intl. Conference on Information Security: Visions and Perspectives (IFIP TC11)*, The Netherlands, May 07-09, pp. 215-226, 2002.

[15] Dowland, P., Singh, H., and Furnell, S., "A Preliminary Investigation of User Authentication Using Continuous Keystroke Analysis", In Proc. of the 8th *IFIP Annual Working Conference on Information Security Management and Small System Security*, Las Vegas, Nevada, 2001.

[16] Villani, M., Tappert, C., Giang, N., Simone, J., Fort, H. St., Sung-Hyuk C., "Keystroke Biometric Recognition Studies On Long-Text Input Under Ideal and Application-Oriented Conditions", *In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06),* New York, USA, June 17-22, pp. 39, 2006.

[17] Monrose, F., Rubin, A., "Authentication Via Keystroke Dynamics", In Proc. of the 4th *ACM Conference on Computer and Communications Security*, Zurich, Switzerland, April 01-04, 1997, pp. 48-56, 1997.

[18] Revett, K., Jahankhani, H., de Magalhaes, S., and Santos, H., "A Survey of User Authentication Based On Mouse Dynamics", *In Proc. of the 4th Intl. Conference on Global E-Security (ICGeS 2008),* London, UK, June 23-25, pp. 210-219, 2008.

[19] Bours, P., Fullu, C.J., "A Login System Using Mouse Dynamics", *In Proc. of the 5thIntl. Conference onIntelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2009)*, Kyoto, Japan, Sept. 12-14, 2009.

[20] Ahmed, A. A and Traore, I., "A New Biometric Technology Based On Mouse Dynamics", *IEEE Transactions on Dependable and Secure Computing* 4, 3 (July), pp. 165-179, 2007.

[21] N. Friedman, D. Geiger, M. Goldszmidt, "Bayesian Network Classifiers", *Machine Learning*, Vol. 29, pp.131-163, 1997.