A Practical Real-Time Authentication System with Identity Tracking Based on Mouse Dynamics

Xiao-jun Chen[1) 2)], Fei Xu[2)], Rui Xu[2)], S.M. Yiu[3)], Jin-qiao Shi[2)]
[1)] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2)] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[3)] Department of Computer Science, The University of Hong Kong, Hong Kong
{chenxiaojun, xufei,shijinqiao}@iie.ac.cn, xurui@nelmail.iie.ac.cn, smyiu@cs.hku.hk

***Abstract*—It is relatively easier for an insider attacker to steal the password of a colleague or use an unattended machine (logged in by other users) within a trusted domain to launch an attack. A simple real-time authentication by password may not work if they have the password. A promising direction is to consider how a user uses the mouse (referred as mouse dynamics). By comparing the stored mouse behavioral profile of the valid user, the system automatically authenticates the user. However, existing approaches are impractical due to long verification time (several minutes on average). An attack may only take seconds (e.g. copying a file, sending an email). In this abstract, we propose a practical real-time authentication method via mouse dynamics, called PAITS (Practical Authentication with Identity Tracking System). The real-time authentication can be done in only 5 seconds with similar accuracy as existing methods. The novelty of PAITS comes from tracking the behavior of users using "short-lived interventional scenarios" (in which the cursor is out of control, e.g. cursor stops, disappears, or slows down) with 71 carefully chosen attributes as the profile.**

I. Introduction

Insider threats to organizational security have been one of the most difficult challenges to address and are receiving increased attention in academic, commercial and government research communities. It is relatively easier for an inside attacker to steal passwords from colleagues or make use of an unattended machine (the valid user may forget to lock the screen or forget to log out) to launch an attack (this is usually referred as identity theft). Since these inside attackers may already get hold of the passwords or some kinds of credentials of other valid users, a simple real-time authentication procedure using these credentials may not be effective. Recently, researchers started to explore whether human-computer interaction (HCI) behavior [1] can be used to detect identity theft. HCI behavioral biometrics mainly consists of human interaction with input devices such as keyboard and mice which is believed as individual biometrics that are more difficult to imitate.

Among different input devices, mouse is a good choice for capturing this HCI behavior. Existing authentication schemes based on mouse dynamics (mouse movements) have demonstrated some promising results and achieved low false rejected ratio (FRR) and false accept ratio (FAR) [2 – 4]. For example, both FRR and FAR is around 2% in [2]. However, these approaches have a common limitation that makes them impractical. Most of them require several minutes on average (some even require more than 10 minutes) to collect enough mouse movement for verification [5]. In real situation, attacks from a malicious insider may only take seconds with very few mouse actions (e.g. copying a file with sensitive information, sending out an email with virus attached, or just putting a Trojan horse in the machine).

The general approach used by existing methods is very similar and is described as follows. They collect mouse movements of valid users, extract features from these movements, train a model to do classification, and store a profile for every user. For examples, back-propagation neural network is used in [2] and C5.0 decision tree is used in [4]. In normal usage, in order to capture enough mouse movements for classification, around 1,000 – 3,000 mouse actions are needed. This would require several minutes or even more than 10 minutes depending on what kind of mouse actions can be captured in that particular session. This makes the verification time unacceptable. Whether the verification can be done in seconds is a challenge.

In this extended abstract, we introduce a practical real-time authentication method via mouse dynamics, called PAITS (Practical Authentication with Identity Tracking System). From our preliminary results, the real-time authentication can be done in only 5 seconds with similar accuracy as existing methods (with 2.86% FRR and 4% FAR). The novelty of PAITS comes from the followings. Instead of capturing the mouse actions in normal operation, we propose to use "short-lived interventional scenarios" (in which the cursor is out of control, e.g. cursor stops, disappears, or slows down, see [6] for an example). With this abnormal situation, it is easier to capture enough mouse actions within a short period of time to do the classification. We also carefully select 71 attributes under these abnormal scenarios to form the profile. Our preliminary experiments show that PAITS performs very well.

II. Details of PAITS

A. System Architecture of PAITS

PAITS consists of: (i) Data capture component which implements three short-lived interventional scenarios (Cursor-Stopping, Cursor-Disappearance and Cursor Slowing) and collect user behavior based on these scenarios. This module will be triggered according to some security measures (e.g. the

978-1-4799-3360-0/14/$31.00 ©2014 IEEE
Corresponding Author: Fei Xu (xufei@iie.ac.cn)

machine has been idle for a few minutes) and (ii) a back-end system which receives mouse movement data and extracts abstract features, and feeds features into the probability neural network (PNN) for training or testing. When an anomaly is detected, data capture component will record current user's operation sequence. Figure1 describes the details of *PAITS*.

In Cursor-Stopping scenario, the cursor is fixed on the center of the screen for five seconds. In Cursor-Disappearance scenario, the cursor gets invisible for five seconds. In Cursor-Slowing scenario, the cursor move shorter distance with one mouse moving action than the normal situation.

For better depicting the characteristic of mouse moves when users are impatience with the short-lived interventional scenario, we add features on movement range and distribution of angle between moves based on previous work, and then form 71 features grouped into three categories: (i) Features on movement range. The range of cursor movement is different between individuals. We divide the full screen into 8 zones and count the distance and number of moves occurred in these zones (zone coordinates will be given in the full paper). (ii) Features on movement direction and speed. We classify the direction of moves into 8 zones in a counterclockwise. An interesting but common observable is that user usually moves the cursor alone the horizontal direction that is directions (1,4,5,8) more, and moves alone the vertical direction that is directions (2,3,6,7) less. So in fact we define direction I as 1,4,5,8 and direction II as 2,3,6,7. We divide the travel distance on direction I into 7 zones and divide the travel distance on direction II into 5 zones. (iii) Distribution of angles between two successive moves. It describes the degree of angle transformation during mouse movement and can be used to distinguish if a user is moving in a circle or moving back and forth.

The back-end system uses PNN model generated based on the total 71 inputs of training data samples to compare normal user mouse behavior and suspected masquerader behavior.

B. Preliminary Evaluation

After 12 volunteers used our system for over one month, we collected 1038 mouse sessions in total, an average of 86 sessions per user. Among all these sessions, there are 757 sessions in Cursor-Stopping scenario (530 sessions for training and 227 sessions for testing), 144 sessions in Cursor-Disappearance scenario (100 sessions for training and 44 sessions for testing) and 137 sessions in Cursor-Slowing scenario (100 sessions for training and 37 sessions for testing).

Environment variables may have huge influence on the experiment results. We make sure that computer environment, pointer velocity and distance is consistent in all participants in order to minimize the effect. We calculated the FRR and FAR in each case. Figure 2 shows the receiver operating characteristic (ROC) curve, represent different FRR and FAR under different spread values (an important parameter in PNN training, which determines the size of the receptive field of the Gaussian kernel). As shown in Figure 2, *PAITS* achieves the best FRR and FAR which are 2.86 and 4 percent under the spread value between 0.36 and 0.38.

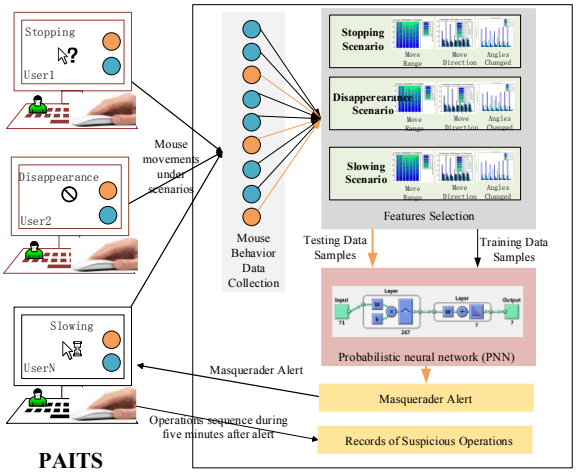

Fig. 1 System architecture of *PAITS*

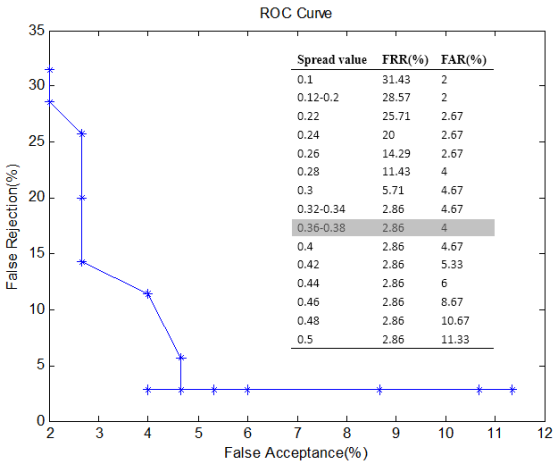

Fig. 2 ROC curve

REFERENCES

[1] Yampolskiy R V. Human computer interaction based intrusion detection[C]//Information Technology, 2007. ITNG'07. Fourth International Conference on. IEEE, 2007: 837-842.

[2] Ahmed, Ahmed Awad E., and Issa Traore. "A new biometric technology based on mouse dynamics." Dependable and Secure Computing, IEEE Transactions on 4.3 (2007): 165-179.

[3] Gamboa H, Fred A. A behavioral biometric system based on human-computer interaction[C]//Defense and Security. International Society for Optics and Photonics, 2004: 381-392.

[4] Pusara M, Brodley C E. User re-authentication via mouse movements[C]//Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security. ACM, 2004: 1-8.

[5] Jorgensen Z, Yu T. On mouse dynamics as a behavioral biometric for authentication[C]//Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. ACM, 2011: 476-482.

[6] Wei Z, Chen X, Pu Y, et al. A Real-Time Authentication Method Based on Cursor-hidden Scene[C]//1st International Workshop on Cloud Computing and Information Security. Atlantis Press, 2013.

978-1-4799-3360-0/14/$31.00 ©2014 IEEE
Corresponding Author: Fei Xu (xufei@iie.ac.cn)