network

MDPI

Article

EchoIA: A Cloud-Based Implicit Authentication Leveraging User Feedback

Yingyuan Yang [1], Jiangnan Li [2], Sunshin Lee [1], Xueli Huang [3] and Jinyuan Sun [2,*]

[1] Computer Science Department, University of Illinois, Springfield, IL 62629, USA; yyang260@uis.edu (Y.Y.); slee675@uis.edu (S.L.)
[2] Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN 37996, USA; jli103@utk.edu
[3] Temple Department of Computer & Information Sciences, Temple University, Philadelphia, PA 19122, USA; tuc36161@temple.edu
* Correspondence: jysun@utk.edu; Tel.: +1-(865)-974-0426; Fax: +1-(865)-974-4404

Abstract: Implicit authentication (IA) transparently authenticates users by utilizing their behavioral data sampled from various sensors. Identifying the illegitimate user through constantly analyzing current users' behavior, IA adds another layer of protection to the smart device. Due to the diversity of human behavior, existing research tends to utilize multiple features to identify users, which is less efficient. Irrelevant features may increase the system delay and reduce the authentication accuracy. However, dynamically choosing the best suitable features for each user (personal features) requires a massive calculation, making it infeasible in the real environment. In this paper, we propose EchoIA to find personal features with a small amount of calculation by leveraging user feedback derived from the correct rate of inputted passwords. By analyzing the feedback, EchoIA can deduce the true identities of current users and achieve a human-centered implicit authentication. In the authentication phase, our approach maintains transparency, which is the major advantage of IA. In the past two years, we conducted a comprehensive experiment to evaluate EchoIA. We compared it with four state-of-the-art IA schemes in the aspect of authentication accuracy and efficiency. The experiment results show that EchoIA has better authentication accuracy (93%) and less energy consumption (23-h battery lifetimes) than other IA schemes.

Keywords: implicit authentication; security; cloud computing; edge computing

Citation: Yang, Y.; Li, J.; Lee, S.; Huang, X.; Sun, J. EchoIA: A Cloud-Based Implicit Authentication Leveraging User Feedback. *Network* **2022**, *2*, 190–202. https://doi.org/10.3390/network2010013

Academic Editor: Rajendra V. Boppana

Received: 23 January 2022
Accepted: 17 March 2022
Published: 21 March 2022

1. Introduction

Recent years have witnessed the rapid growth of smart technologies such as smartphones, smart glasses, and smartwatches. On the one hand, people rely heavily on smart devices to share information and gain services, which become primary elements of our daily lives [1]. On the other hand, the security problem raised by smart devices has become more important than ever before [2]. One of the most critical issues is user authentication, especially in cloud and edge computing.

To identify users, most of the existing systems use explicit approaches (explicit authentication), such as passwords, PINs, and draw patterns. However, explicit authentication requires user-system interaction, which could be frustrating, especially when the users possess many different passwords. A recent survey [3] shows 3% percent of people forget a password at least once a week. Explicit authentication can also be circumvented and be broken [4]. Therefore, researchers begin to study new authentication methods to enhance explicit authentication.

Utilizing sensors' data sampled by the smart device, implicit authentication (IA) transparently identifies users by constantly comparing current users' behavioral data with historical legitimate users' behavioral data [5]. The comparing or classification process is usually achieved using various machine learning models, e.g., SVM. At the same time,

most of the calculations and storage are generally offloaded to the cloud due to the energy limitation of smart devices [6]. Furthermore, since users do not need to interact with the system in the authentication, IA can be seamlessly applied to various authentication systems, which adds another layer of protection to the smart device. If suspicious behaviors were detected during the usage, IA will lock the device and ask users to perform a multi-factor authentication, e.g., inputting passwords [7,8]. From the users' aspect, due to IA's transparency, they will not notice IA until the device has been locked. In addition, implicit authentication does not require user–system interaction, which releases users from tedious password inputting and the burden of memorizing the passwords.

In implicit authentication (IA), the features used for the authentication are predefined by the system, which will not be able to change during the usage [9,10]. To achieve better coverage in user authentication, the existing approaches tend to use multiple features [11–13], such as location, touch, and acceleration. However, only a small number of the features (personal features) in the entire feature set are needed for a specific user. Therefore, irrelevant features not only reduce the system's efficiency, but decrease the authentication accuracy as well. Nevertheless, due to the high complexity of human behavior, it requires a massive calculation to derive personal features, which is infeasible in practice [14]. Hence, finding a suitable scheme that dynamically derives personal features is critical for IA implementation.

This work introduces a human-centered implicit authentication (EchoIA), which utilizes user feedback to pinpoint personal features during the usage with a small amount of calculation. By comparing with current IA schemes, the experiment shows that our method can significantly improve the authentication accuracy of traditional IA. In addition, the proposed method is lightweight, which can be easily embedded into existing IA systems as an add-on to achieve efficient authentication. As far as we know, we are the first group that utilizes user feedback to improve authentication accuracy and energy efficiency in IA.

The major advantage of implicit authentication (IA) is its transparency, releasing users from the tedious authentication process. However, it is challenging to gather user feedback in a transparent environment since directly asking users' input will break the transparency. Even though we could have various user feedback, pinpointing the best suitable features is also challenging. To this end, we propose a method that utilizes the correct rate of inputted passwords to implicitly collect user feedback and find personal features. In implicit authentication, the system will lock the device and deem current users illegitimate when their behavior mismatches legitimate users' historical behavior. Legitimate users may be locked out due to the misidentification caused by using unsuitable features, but they can input a correct password to unlock the device. Illegitimate users may also input a valid password to unlock the device after several attempts, but their correct rate of inputted passwords will be lower than legitimate users'. EchoIA can utilize the correct rate of inputted passwords to deduce current users' true identities and further adjust feature sets to better match users' behavior. The detailed procedure is discussed in Section 2.

This paper makes the following contributions:

- We proposed EchoIA to find the best suitable features (personal features) for each user by utilizing user feedback. EchoIA maintains the transparency of implicit authentication, while can choose personal features for legitimate users based on their recent behavior.
- We implemented EchoIA in a real environment using the Android system and multiple servers. To evaluate the proposed method, we also implemented four state-of-the-art implicit authentication schemes.
- We collected users' behavioral data in the past two years. In addition, we evaluated the proposed method in the aspect of authentication accuracy, computational efficiency, and energy efficiency. In the experiment, EchoIA has a better authentication accuracy and a lower energy cost.

2. Related Work

To improve explicit authentication mechanisms such as PIN and passlocks, various implicit authentication schemes have been proposed as secondary authentication mechanisms [5,6,11,13,15–21]. Among them, leveraging different features, Shi scheme [6], MultiSensor scheme [11], Gait scheme [19], and SilentSense scheme [20] are four different schemes that represent four research directions of state-of-the-art implicit authentications [22,23]. In addition, current implicit authentication research tends to adopt all the available features to achieve better authentication accuracy [11,14,24]. On one hand, due to the high complexity of users' behavior, utilizing only a specific behavior metric is not sufficient to identify them in practice. On the other hand, to identify a specific user, only a small portion of the total behavior metrics is needed [6,11,13,22,24]. Reducing the number of features can also exponentially decrease the system's energy and time consumption [25]. However, to find personal features requires additional calculation [11,14], which increases time and energy consumption. Leveraging user feedback, EchoIA can select the best suitable features for different users. During the usage, the legitimate users can also notify the system to update personal features if their behavior changed, e.g., injury.

Most of the existing research in implicit authentication utilizes the support vector machine (SVM) [11,20,26–28] to identify users. Other classifiers, e.g., Gaussian mixture model (GMM [25]), have also been used in implicit authentication [6]. We mainly adopted SVM to achieve user authentication in this work.

3. The System Overview

Implicit authentication (IA) identifies users by constantly comparing current users' behavioral data with legitimate users' historical behavioral data. If current users' behavioral data is different, IA will lock the device and ask the users to input passwords. Due to the noise and behavioral change, it is common that legitimate users are falsely blocked by the device [14]. In implicit authentication, if users input correct passwords, they can continue to use the device, where the setting of the original IA system will not change. In EchoIA, however, if users input correct passwords and prove their identities, the system will unlock the device and adjust the features to align with legitimate users' behavior.

As shown in Figure 1, EchoIA contains two phases, *Initialization* and *Authentication*. The *Initialization* phase only takes place for the first time of the usage. Meanwhile, we assume the users at the *Initialization* phase are legitimate. In the *Initialization* phase, candidate features are sent to legitimate users, who may rank the features based on their behavior. The combination of top-ranking features and system-default features is used as personal features to identify users. Note that the personal features are dynamically adjusted according to different users and devices. In the *Authentication* phase, EchoIA utilizes the correct rate of inputted passwords to adjust personal features. Specifically, user feedback is implicitly obtained through the process of inputting passwords (Section 3.1), which keeps IA's transparency. However, to prevent illegitimate users from taking advantage of the system, it only updates personal features after users entered a correct PIN number, which must be different from passwords used for unlocking the device. A secured channel is established to transmit data between client and server. We adopted a Wind Vane module [14] to optimize the data transmission efficiency.

From the system's point of view, inputting incorrect passwords will enhance its confidence in using existing personal features; inputting correct passwords will reduce its confidence in using existing personal features and encourage it to choose different features. Since most of the time, the system is running at the *Authentication* phase, the users will not be able to notice the existence of EchoIA during the usage. The following sections will discuss the detail of the *Initialization* phase and the *Authentication* phase.

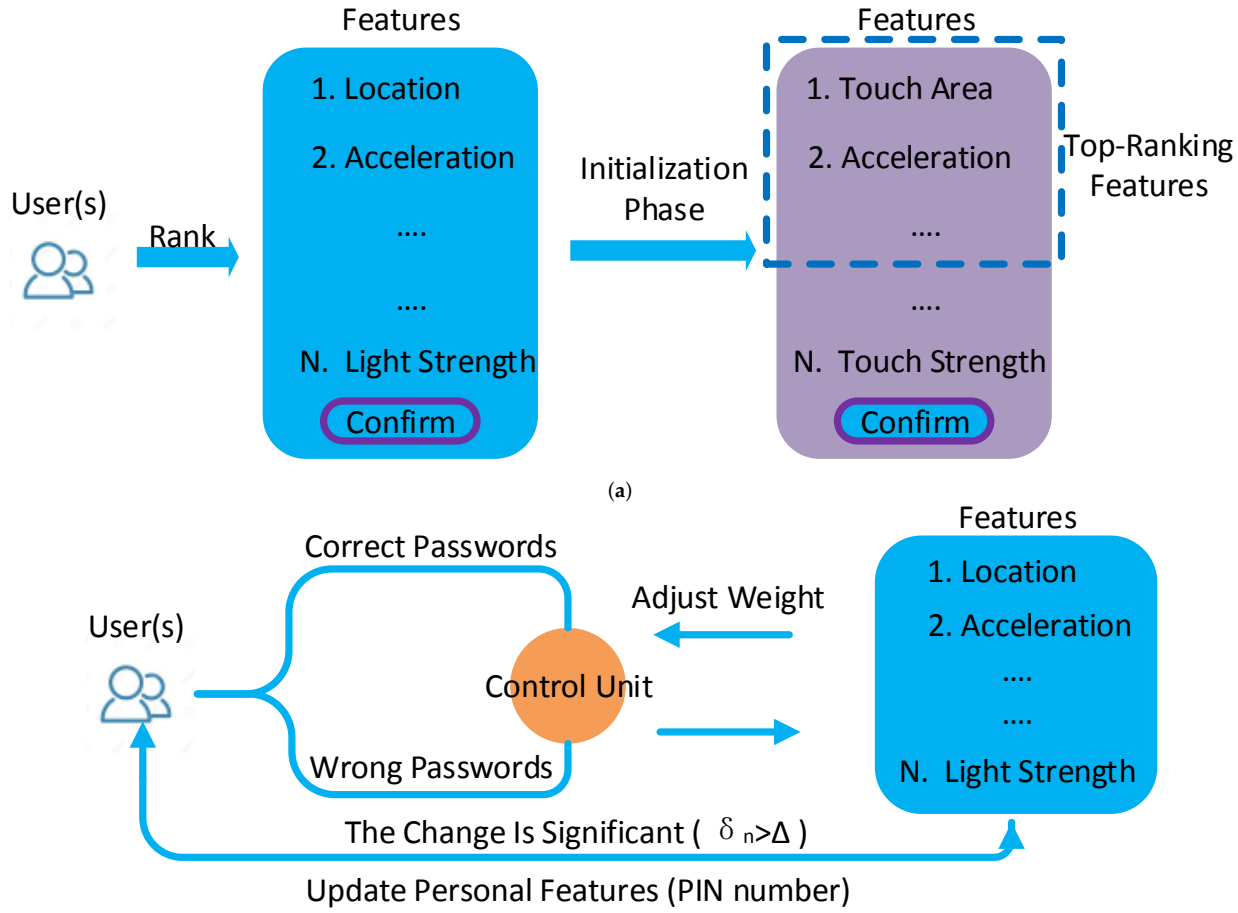


Figure 1. EchoIA overview. (**a**) The *Initialization* phase. (**b**) The *Authentication* phase.

3.1. The Initialization Phase

As its name suggests, the *Initialization* phase mainly focuses on initializing personal features and associated system settings. The users will spend a short time in this phase in order to help the system to prepare the authentication.

As shown in Figure 1a, at the *Initialization* phase, EchoIA will send a message contains all candidate features to the users. Based on their own behavior, the users will rank candidate features, where the result will be sent back to the remote servers for further processing. For each feature, there is an associated weight parameter, which will be initialized at this phase. The total available features in the smart device are F. Note that the elements in F are various for different devices, and can be updated during the usage once new features are introduced.

$$F = \{f_1, f_2, f_3, ..., f_n\} \tag{1}$$

To ensure reliability, some of the features are system preserved, which is not shown in F. For example, a touch trajectory feature is preserved since it has high accuracy when identifying most of the users. For each feature in F, the corresponding weight is predefined in W.

$$W = \{w_1, w_2, w_3, ..., w_n\} \tag{2}$$

The users may rank the features based on their routine. To this end, EchoIA will renew the weight for each feature based on the users' ranking.

$$w_n = \frac{1}{r_n}, \tag{3}$$

where w_n is the weight of the nth feature; and r_n is the associated ranking of the feature. The top-ranking features only contain a part of elements in F and are dynamically changed during the usage. For example, at some moments, the top-ranking features, F_{top}, may only contain 5 different features $F_{top} = \{f_2, f_4, f_5, f_7, f_8\} \subset F$. Personal features in this example will have both F_{top} and system reserved features.

In real usage, users may change their behavior, which is common in practice. To better identify the users, the system also needs to adjust personal features according to the behavioral change. The details of adjusting personal features will be discussed in the next section.

3.2. The Authentication Phase

To dynamically adjust personal features, the system can directly send the message to users to request new features, but this approach will break IA's transparency. In addition, it is difficult for the system to decide the "right time" to send the request, since the system will not know the behavioral change unless it analyzed the data. In EchoIA, instead of analyzing users' behavioral data, the system leverages the correct rate of inputted passwords to dynamically adjust personal features.

At the *Authentication* phase, as shown in Figure 1b, the system will reduce the weights of each feature in F_{top} if users input correct passwords. Since users only need to input passwords when IA locks the device, correct passwords indicate current users have a large chance of being legitimate. Similarly, the system will increase the weight of each feature in F_{top} if users input incorrect passwords. The new weight is updated by δ_n.

$$\delta_n = \delta_n^{(C)} - \delta_n^{(I)}, \tag{4}$$

where $\delta_n^{(I)}$ is the amount of weight increased for the feature n in F_{top}; $\delta_n^{(C)}$ is the amount of weight decreased for the feature n in F_{top}. The new weight of the feature n is calculated by $w_n + \delta_n$ ($\delta_n^{(C)}$ indicates the system has less confidence in the current data samples).

In EchoIA, a predefined threshold Δ is used to measure the significance of the weight change. In practice, we choose Δ by using k-fold cross-validation. If the change is significant, $\delta_n > \Delta$, the system will challenge the users to input a PIN number, which is the number different from passwords used to unlock the device. The users can choose the PIN number at the *Initialization* phase. If the users type a correct PIN and agree with personal features' change, EchoIA will update F_{top} and personal features according to the new weights in F. In this process, decayed features will be replaced by new features. The system will use the updated personal features to identify the users until $\delta_n > \Delta$ again. We adopted the support vector machine (SVM) to achieve the user classification and to identify legitimate users. The parameters in the model are optimized by using k-fold cross-validation. We have summarized the *Authentication* phase in Algorithm 1.

Algorithm 1: EchoIA (Simplified).

Input: A Password Input, δ_n in the Previous Step
Output: δ_n in the Current Step

initialize psw:=A Password Input ;
initialize $D[n]_{pre}$:=δ_n in the Previous Step ;
/* $D[n]_{pre}$ denotes the weight change amount of f_n in the previous step. */
initialize $D[n]_{cur}$:= $D[n]_{pre}$;
/* $D[n]_{cur}$ denotes the weight change amount of f_n in the current step. */
/* The system will not update F_{top} if $D[n]_{cur} \leq \Delta$. */
if *(psw==PSW_LEGITIMATE_USER)* **then**
    /* The current user inputs the correct password. */
    f[n]=f[n]-C ;
    /* Reducing the weight of corresponding feature. */
    temp_change=C ;
    /* The amount of change in the current step (Usually a positive value). */
else
    /* The current user inputs the wrong password. */
    f[n]=f[n]+I ;
    /* Increasing the weight of corresponding feature. */
    temp_change=-I ;
    /* The amount of change in the current step (Usually a negative value). */
$D[n]_{cur}$=$D[n]_{cur}$+temp_change ;
/* The total amount of weight changes until the current step. */
return $D[n]_{cur}$;
/* The system will update F_{top} if $D[n]_{cur} > \Delta$. */

4. Implementation

We implemented EchoIA by using the Android system and multiple servers. The system architecture is shown in Figure 2. The data collection is achieved at the user-end, in which an application is created to collect users' behavioral data. The user-end application constantly samples users' behavioral data from various sensors and sends it to the Control Server by using a secured channel. As shown in Figure 2, multiple users can connect with the Control Server at the same time. The Control Server contains three main components, Control Unit, Authentication Unit, and Message Unit. As mentioned in Section 3, the Control Unit is responsible for updating the weight parameter associated with each feature. The Authentication Unit leverages implicit authentication to constantly monitor users' behavior and compare it with legitimate users' historical behavior. In order to compare EchoIA with other IA schemes, we also implemented four state-of-the-art IA schemes in the Authentication Unit, called Shi-IA [6], Multi-Sensor-IA [11], Gait-IA [19], and SilentSense-IA [20]. Finally, the Message Unit is used to communicate with users during the *Initialization* phase and the *Authentication* phase. All users' data is formatted and stored in the Database Server.

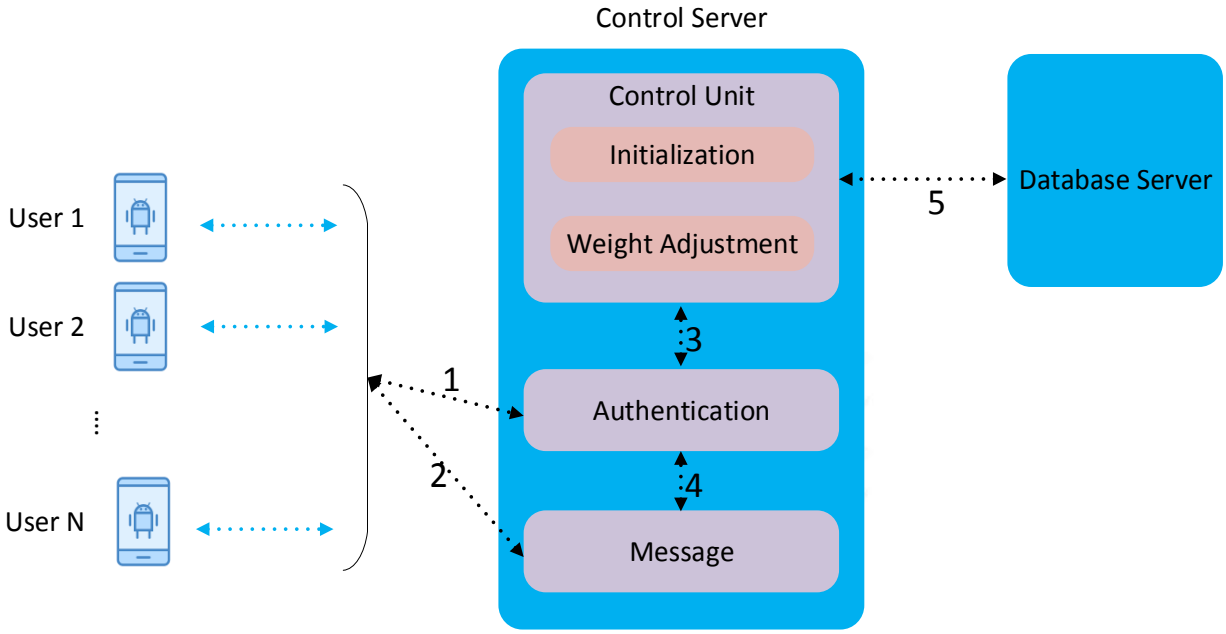


Figure 2. The system architecture of EchoIA. Data flow: 1. Implicit authentication is achieved by comparing current users' data with their historical data. IA will lock the device and asks users to input passwords if it fails to identify the current user. 2. The system utilizes the inputted passwords to adjust personal features. It will request users' PINs to update weights for F_{top} and personal features by exchanging messages with legitimate users. 3. Feature' weights are initialized and used to authenticate users. 4. After inputting a correct PIN, users can update F_{top}. 5. Users can reinitialize the system when inputting a correct PIN. 6. The Control Unit will constantly update the Database Server if the current users are deemed legitimate.

5. Evaluation

In the real experiment, we tracked the usage of 17 participants during the past two years. Each user was in turn selected as the legitimate user, while another user was deemed as illegitimate users. Illegitimate users were required to use the device at least 10% of the total usage time. Utilizing our system, we gathered rich information from all users under different environments. In the experiment, we mainly use the following 12 features to achieve user authentication: accelerometer, orientation, magnetometer, gyroscope, touch, light, pressure, temperature, GPS, microphone, battery usage, and WiFi status.

To compare EchoIA with other state-of-the-art IA schemes [22], we implemented Shi-IA [6], Multi-Sensor-IA [11], Gait-IA [19], and SilentSense-IA [20]. We used the recommended settings of the original papers [6,11,19,20] in the experiment. In addition, the feature selection strictly follows the descriptions of the original works, while the parameters were optimized by using k-fold cross-validation. Finally, we tested the performance of different schemes by using the same data.

In the experiment, we evaluated the authentication accuracy, CPU utilization, memory utilization, and energy consumption for each IA schemes, including EchoIA. The experiment details are described in the following sections.

We first compared the authentication accuracy of different IA schemes, where results are shown in Figure 3. In the figure, users' data was divided into ten parts based on the timeline. The first part of the data was used to train the models (The first parts contain 15% of the total data since the training data of this size optimizes the authentication accuracy across all five IA schemes). We then calculated the authentication accuracies for different schemes by using the other parts of the data. The authentication accuracy was calculated by $ACC = \frac{TR+TA}{TR+TA+FR+FA}$, where true accept ($TA$) indicates the legitimate user has been correctly identified; the false reject (FR) indicates the legitimate user has been incorrectly identified to be the illegitimate user; the true reject (TR) indicates the illegitimate user has been correctly identified; the false accept (FA) indicates the illegitimate user has been incorrectly identified to be the legitimate user. The ultimate goal of the authentication is to

reduce both *FR* and *FA* to 0, where the system will achieve 100% accuracy. We also adopted the retraining techniques discussed in our previous work [7] to improve the authentication accuracy for all five schemes.

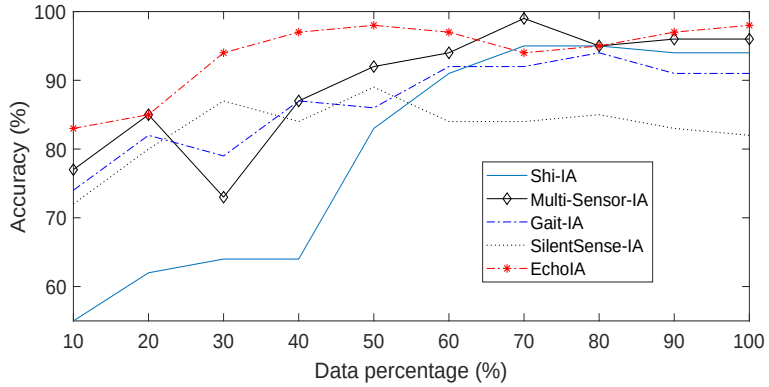


Figure 3. Accuracies for different IA schemes.

5.1. Authentication Accuracy

EchoIA has the highest authentication accuracy in most of the tests, as shown in Figure 3. Multi-Sensor-IA also has a high authentication accuracy compared to other IA schemes. In the experiment, most of the schemes reach to more than 90% accuracy after using 80% of the data except SilentSense-IA. Note that there are some fluctuations in different IA schemes due to behavior changes, where both Multi-Sensor-IA and Gait-IA have an accuracy drop at the point of 30%. We analyzed the data of Multi-Sensor IA at that point, which shows that most users traveled to different places that did not appear in the training phase. The machine learning model cannot separate users based on the given training data. Since, for some users, the traveling and staying time is non-negligible, it becomes harder for Multi-Sensor IA to make a decision based on the previous training data. After the behavioral data in this new location is collected and stored in training data, the accuracy of Multi-Sensor IA eventually increases and becomes similar to other IA schemes. However, this behavior change does not affect the proposed EchoIA since the system automatically updates users' personal features during usage. In such a case, the accuracy curve for EchoIA is much smoother than the other schemes. As literature usually does in template updating related research [22,29,30], we measured the EERs and ROCs for EchoIA, Multi-Sensor-IA, Gait-IA, SilentSenseIA, and Shi-IA, which provide more reliable results. The EERs for EchoIA, Shi-IA, Muti-Sensor-IA, Gait-IA, and SilentSense-IA are 0.1428, 0.2200, 0.1635, 0.2700, and 0.2720, respectively; and corresponding AUCs are 0.8991, 0.8588, 0.9244, 0.8103, and 0.8013, respectively.

In EchoIA, we calculated an average authentication accuracy for 17 users by utilizing all the data spanned two years. The result is shown in Figure 4, where the average accuracy across all users is 93.23%.

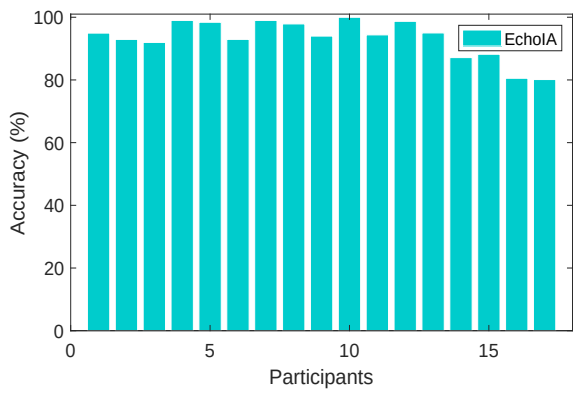
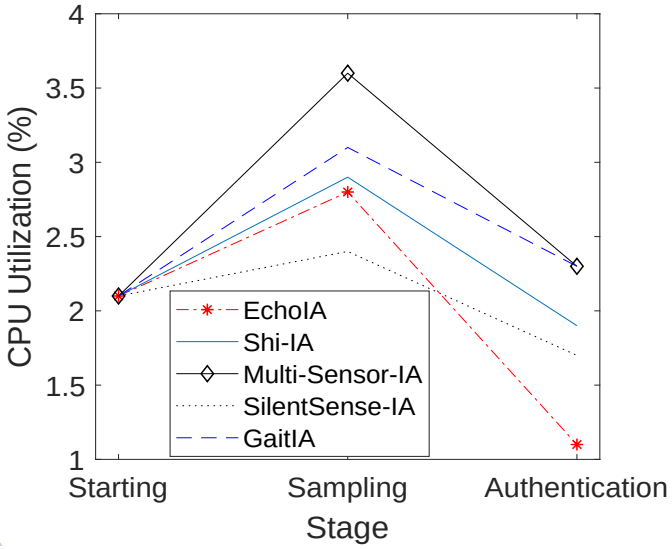


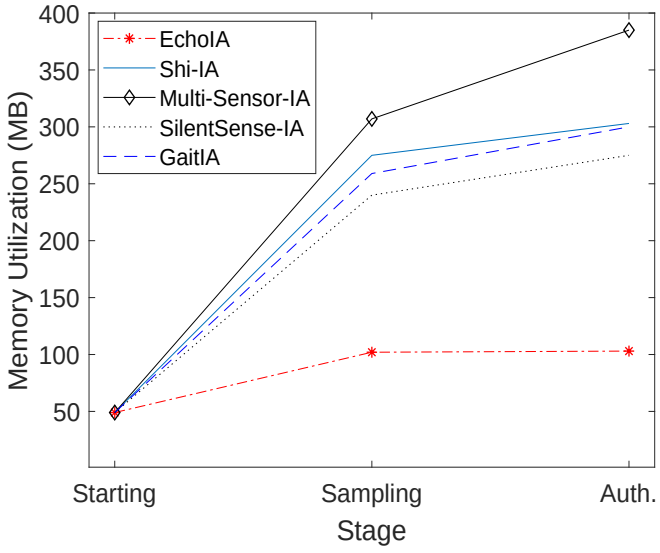
Figure 4. The accuracy for each user.

5.2. CPU Utilization and Memory Utilization

In addition, we evaluated the CPU usage and memory usage for different schemes. In the experiment, we recorded the CPU utilization of different IA schemes at three stages, Start, Sampling, and Authentication. The experiment results are shown in Figure 5a. At the Sampling stage, the CPU utilization of EchoIA is the second-lowest for all five schemes. At the Authentication stage, the CPU utilization of EchoIA is the lowest among all the schemes. Since most of the time the users are at the Authentication stage, the total amount of CPU utilization of EchoIA is the smallest for all the schemes. In the experiment, Multi-Sensor-IA has the highest CPU utilization, but it also has a high authentication accuracy similar to EchoIA.



(**a**)

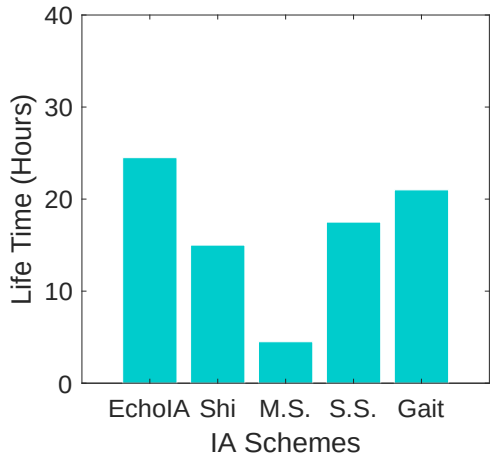


(**b**)

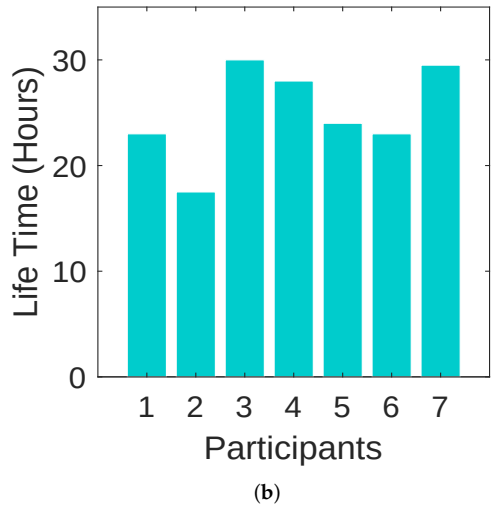
Figure 5. CPU Utilization and Memory Utilization (User-End). (**a**) CPU utilization. (**b**) Memory utilization. At the Starting stage, the user-end application and services begin to launch. At the Sampling stage, sensors' data is sampled and periodically uploaded to the server. The *Initialization* phase is at the Sampling stage. At the authentication stage, utilizing the model returned from the server, the current user will be classified into two categories, legitimate or illegitimate. The *Authentication* phase is at this stage. Note that we connected different stages using lines in the figure to improve the readability. The stages are actually independent data points.

We recorded the memory utilization of various schemes at different stages. The result is shown in Figure 5b, in which the EchoIA has the lowest memory utilization among all the five schemes. Since EchoIA only uses a small portion of features to train the model and to authenticate users, the total amount of memory used to store the data is smaller than other schemes. As shown in Figure 5b, the memory utilization of Multi-Sensor-IA is the highest since it uses all the features and associated sensors' data on the device.

In the experiment, we evaluated the battery consumption for different schemes. The result is shown in Figure 6. We measure the battery usage of different schemes by calculating the average working hours of battery after fully charged. As shown in Figure 6a, EchoIA has the longest battery lifetime, 23 h on average. The Multi-Sensor-IA has the shortest battery lifetime, four hours on average.



(**a**)



(**b**)

Figure 6. Battery Utilization. (**a**) Battery utilization of different schemes. M.S. denotes Multi-Sensor-IA. S.S. denotes SilentSense-IA. (**b**) Battery utilization of different participants.

We also calculated the average battery lifetime of each participant by using EchoIA. Figure 6b shows the battery lifetime derived from the data of seven different participants. They are randomly selected. We calculated an average battery lifetime across all participants' data for comparison purposes, which is 23 h. There are large differences in the battery utilization between users. As shown in the figure, participant 2 has the shortest battery lifetime, which is 16 h. Participant 7, however, has the longest battery lifetime, which is 30 h.

5.3. Energy Consumption of the User-End Application

We also tracked the performance of the user-end application. In the experiment, we compared EchoIA with popular applications, such as Instagram, Facebook, Twitter, eBay, and LinkedIn. We continuously tracked CPU utilization and memory utilization for different applications during the usage. The result is shown in Table 1. Please note that in Figure 5, we calculated the CPU and battery utilization only based on the data at the sampling and authentication stages. In Table 1, we also gathered data from other stages, e.g., switching to a different application.

Table 1. CPU and memory consumption.

	Instagram	**Facebook**	**Twitter**	**LinkedIn**	**eBay**	**EchoIA**
CPU Avg%	7.1	11.6	6.0	5.9	5.3	1.3
CPU Max%	10.0	16.5	10.3	9.6	8.6	3.9
Mem. Max(MB)	121.2	157.0	101.4	138.2	111.6	103

Note that the purpose of this table is to demonstrate that EchoIA will not cost too much energy and calculation in daily usage. We do not claim that EchoIA is better than any of those social media apps listed above. The result may vary for different users. Please refer to Figure 5 for an accurate evaluation.

Table 1 shows the average and maximum CPU consumption for each application. EchoIA has the lowest average CPU consumption compared to other applications, which is 1.3%. Similarly, EchoIA also has the lowest maximum CPU consumption, which is 3.9%. EchoIA also consumes a small amount of memory in real usage, which only occupies a maximum of 103 MB memory.

6. Conclusions

We proposed EchoIA to find the best suitable features (personal features) for different users by utilizing user feedback. To achieve better coverage, the existing works in implicit authentication tend to use many different features to identify users, which is less efficient and may decrease the authentication accuracy. Without using additional calculations, it is difficult to dynamically choose personal features due to the transparency of IA. Leveraging the correct rate of inputted passwords, EchoIA implicitly gathers user feedback to choose personal features, while maintaining the transparency of IA. To evaluate the proposed method, we implemented EchoIA and four state-of-the-art IA schemes by using the Android system and multiple servers. The results show that EchoIA has better authentication accuracy (93%) and less energy consumption (23-h battery lifetimes) than other IA schemes. EchoIA can automatically update the personal feature for each user when their behavior changes, which significantly boosts IA's accuracy and usability. In addition, the low energy consumption makes EchoIA suitable for most smart devices, such as smartphones, smartwatches, and smart glasses. In the future, to benefit associated research, we will share the system's source code, parameter settings, and dataset on our lab website.

Author Contributions: Data curation, S.L.; Formal analysis, Y.Y., X.H. and J.S.; Investigation, Y.Y. and J.L.; Writing—original draft, Y.Y. and J.S.; Writing—review & editing, Y.Y. and J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the US National Science Foundation (NSF) grant number CNS-1422665 and the Army Research Office (ARO) grant number 66270-CS.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was partially supported by the US National Science Foundation (NSF) under grant CNS-1422665 and the Army Research Office (ARO) under grant 66270-CS.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ceci, L. Average Time Spent Daily on a Smartphone in the United States 2021. Available online: https://www.statista.com/statistics/1224510/time-spent-per-day-on-smartphone-us/ (accessed on 16 December 2021).
2. Zinkus, M.; Jois, T.M.; Green, M. Data security on mobile devices: Current state of the art, open problems, and proposed solutions. *arXiv* **2021**, arXiv:2105.12613.
3. ROSIE TAYLOR. What IS My Password. Available online: https://www.dailymail.co.uk/news/article-6892327/What-password-Britons-forget-security-information-fortnight-researchers-say.html (accessed on 9 March 2020).
4. Brian Donohue. Lock Screen Bypass Flaw Found in Samsung Androids. Available online: http://threatpost.com/lock-screen-bypass-flaw-found-samsung-androids-030413/77580 (accessed on 21 November 2020).
5. Yang, Y.; Huang, X.; Guo, Y.; Sun, J.S. Dynamic multi-level privilege control in behavior-based implicit authentication systems leveraging mobile devices. In Proceedings of the 2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Delhi, India, 10–13 December 2020; pp. 229–237.
6. Shi, E.; Niu, Y.; Jakobsson, M.; Chow, R. Implicit authentication through learning user behavior. In *International Conference on Information Security*; Springer: Berlin/Heidelberg, Germany, 2011.
7. Yang, Y.; Sun, J.; Li, P. Model retraining and dynamic privilege-based access control for implicit authentication systems. In Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Dallas, TX, USA, 19–22 October 2015.
8. Lee, W.H.; Lee, R.B. Multi-sensor authentication to improve smartphone security. In Proceedings of the Conference on Information Systems Security and Privacy, Angers, France, 9–11 February 2015.
9. Shen, C.; Chen, Y.; Guan, X. Performance evaluation of implicit smartphones authentication via sensor-behavior analysis. *Inf. Sci.* **2018**, *430*, 538–553. [CrossRef]
10. Castelluccia, C.; Duermuth, M.; Golla, M.; Deniz, F. Towards Implicit Visual Memory-Based Authentication. In Proceedings of the Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 26 February 2017.
11. Yang, Y.; Huang, X.; Li, J.; Sun, J. BubbleMap: Privilege Mapping for Behavior-based Implicit Authentication Systems. *arXiv* **2020**, arXiv:2006.08817.
12. Bello, A.A.; Chiroma, H.; Gital, A.Y.; Gabralla, L.A.; Shafii, M.A.; Shuib, L. Machine learning algorithms for improving security on touch screen devices: A survey, challenges and new perspectives. *Neural Comput. Appl.* **2020**, *32*, 13651–13678 . [CrossRef]
13. Xu, X.; Yu, J.; Chen, Y.; Hua, Q.; Zhu, Y.; Chen, Y.C.; Li, M. TouchPass: Towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. In Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, London, UK, 21–25 September 2020; pp. 1–13.
14. Yang, Y.; Sun, J.; Guo, L. PersonaIA: A Lightweight Implicit Authentication System based on Customized User Behavior Selection. *IEEE Trans. Dependable Secur. Comput.* **2016**, *16*, 113–126. [CrossRef]
15. Ravi, N.; Dandekar, N.; Mysore, P.; Littman, M.L. Activity Recognition from Accelerometer Data. In Proceedings of the AAAI, Pittsburgh, PA, USA, 9–13 July 2005; Volume 5, pp. 1541–1546.
16. El-Soud, M.W.A.; Gaber, T.; AlFayez, F.; Eltoukhy, M.M. Implicit authentication method for smartphone users based on rank aggregation and random forest. *Alex. Eng. J.* **2021**, *60*, 273–283. [CrossRef]
17. Cheung, W.; Vhaduri, S. Context-Dependent Implicit Authentication for Wearable Device Users. In Proceedings of the 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, London, UK, 31 August–3 September 2020; pp. 1–7.
18. Zhu, T.; Weng, Z.; Song, Q.; Chen, Y.; Liu, Q.; Chen, Y.; Lv, M.; Chen, T. ESPIALCOG: General, Efficient and Robust Mobile User Implicit Authentication in Noisy Environment. *IEEE Trans. Mob. Comput.* **2020**, *21*, 555–572. [CrossRef]
19. Frank, J.; Mannor, S.; Precup, D. Activity and Gait Recognition with Time-Delay Embeddings. In Proceedings of the AAAI, Atlanta, GA, USA, 11–15 July 2010.
20. Bo, C.; Zhang, L.; Li, X.Y.; Huang, Q.; Wang, Y. Silentsense: Silent user identification via touch and movement behavioral biometrics. In Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, Miami, FL, USA, 30 September–4 October 2013; pp. 187–190.
21. Wei, F.; Vijayakumar, P.; Kumar, N.; Zhang, R.; Cheng, Q. Privacy-Preserving Implicit Authentication Protocol Using Cosine Similarity for Internet of Things. *IEEE Internet Things J.* **2020**, *8*, 5599–5606. [CrossRef]
22. Khan, H.; Atwater, A.; Hengartner, U. A comparative evaluation of implicit authentication schemes. In *International Workshop on Recent Advances in Intrusion Detection*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 255–275.
23. Mehrabi Koushki, M.; Obada-Obieh, B.; Huh, J.H.; Beznosov, K. Is Implicit Authentication on Smartphones Really Popular? On Android Users Perception of Smart Lock for Android. In Proceedings of the 22nd International Conference on Human-Computer Interaction with Mobile Devices and Services, Oldenburg, Germany, 5–8 October 2020; pp. 1–17.
24. Yang, Y.; Sun, J. Energy-efficient W-layer for behavior-based implicit authentication on mobile devices. In Proceedings of the INFOCOM 2017-IEEE Conference on Computer Communications, Atlanta, GA, USA, 1–4 May 2017; pp. 1–9.
25. Bishop, C.M. Pattern recognition. *Mach. Learn.* **2006**, *128*, 1–58.

26. Karanikiotis, T.; Papamichail, M.D.; Chatzidimitriou, K.C.; Oikonomou, N.C.I.; Symeonidis, A.L.; Saripalle, S.K. Continuous Implicit Authentication through Touch Traces Modelling. In Proceedings of the 2020 IEEE 20th International Conference on Software Quality, Reliability and Security (QRS), Macau, China, 11–14 December 2020; pp. 111–120.
27. Shi, D.; Tao, D.; Wang, J.; Yao, M.; Wang, Z.; Chen, H.; Helal, S. Fine-Grained and Context-Aware Behavioral Biometrics for Pattern Lock on Smartphones. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **2021**, *5*, 1–30. [CrossRef]
28. Abuhamad, M.; Abusnaina, A.; Nyang, D.; Mohaisen, D. Sensor-based Continuous Authentication of Smartphones Users Using Behavioral Biometrics: A Contemporary Survey. *IEEE Internet Things J.* **2020**, *8*, 65–84. [CrossRef]
29. Pisani, P.H.; Mhenni, A.; Giot, R.; Cherrier, E.; Poh, N.; Ferreira de Carvalho, A.C.P.d.L.; Rosenberger, C.; Amara, N.E.B. Adaptive biometric systems: Review and perspectives. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 1–38. [CrossRef]
30. Pisani, P.H.; Giot, R.; De Carvalho, A.C.; Lorena, A.C. Enhanced template update: Application to keystroke dynamics. *Comput. Secur.* **2016**, *60*, 134–153. [CrossRef]