Hernández-Álvarez, L., Fuentes, J.M., González-Manzano, L., Hernández-Encinas, L. (2021). SmartCAMPP - Smartphone-based continuous authentication leveraging motion sensors with privacy preservation. *Pattern Recognition Letters*, 147, pp. 189-196.

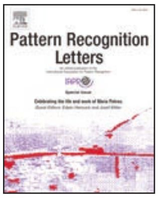
DOI: 10.1016/j.patrec.2021.04.013

Pattern Recognition Letters 147 (2021) 189–196

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



SmartCAMPP - Smartphone-based continuous authentication leveraging motion sensors with privacy preservation[☆]

Luis Hernández-Álvarez[a], José María de Fuentes[b,*], Lorena González-Manzano[b], Luis Hernández Encinas[a]

[a] *Institute of Physical and Information Technologies (ITEFI), Spanish National Research Council (CSIC), Madrid, Spain*
[b] *Computer Security Lab (COSEC), Universidad Carlos III de Madrid, Madrid, Spain*

ARTICLE INFO

Article history:
Received 16 October 2020
Revised 3 February 2021
Accepted 8 April 2021
Available online 27 April 2021

Keywords:
Continuous authentication
Smartphone
Privacy
Gyroscope
Accelerometer

ABSTRACT

Continuous Authentication (CA) approaches are attracting attention due to the explosion of available sensors from IoT devices such as smartphones. However, a critical privacy concern arises when CA data is outsourced. Data from motion sensors may reveal users' private issues. Despite the need for CA in smartphones, no previous work has explored how to tackle this matter leveraging motion sensors in a privacy-preserving way. In this work, a mechanism dubbed SmartCAMPP is proposed to achieve CA based on gyroscope and accelerometer data. Format-preserving encryption techniques are applied to privately outsource them. Our results show the suitability of the proposed scheme, featuring 76.85% of accuracy while taking 5.12 ms. of computation for authenticating each user. Interestingly, the use of cryptography does not lead to a significant impact as compared to a non-privacy-preserving mechanism.

1. Introduction

In the last years, a plethora of connected devices have rocketed worldwide, leading to the so called Internet of Things (IoT). Among them, smartphones are significantly prominent –1.52 bn devices were sold in 2019[1]. Beyond phone calls, they are able to store sensitive information such as pictures or documents. Therefore, their security is paramount.

Beyond their malware-related and advanced persistent threats (e.g., Fancy Bear Android implant [1]), a critical issue is ensuring that the user is the right one. Since smartphones may be physically stolen, it is essential to check this issue at all times. This calls for the inclusion of Continuous Authentication (CA) approaches, such as those based on behavioral biometrics [2].

Smartphone-based CA mechanisms may leverage the growing amount of data provided by its sensors. Currently, an average device counts on a wide array of them such as touch sensors, GPS or accelerometer, to name a few [3].

Given that a great number of CA schemes are based on machine learning techniques, which typically involve a training phase leveraging data from different users, these processes may be outsourced to a cloud–based infrastructure. However, sensorial information may reveal the identity or be used for profiling the user [4]. Therefore, building an outsourced CA mechanism for smartphones requires applying privacy protection techniques.

Several previous works have addressed this particular setting, such as [5] or [6]. However, no previous effort has leveraged on motion sensors, particularly accelerometer and gyroscope data, which have been largely used in other contexts [7–9]. Nevertheless, [10] and [11] study a great number of CA proposals and their analysis show the appropriateness of using accelerometers and gyroscopes for CA purposes. These sensors are those that lead to challenging results, e.g. [2,12], and thus, they are chosen for this work. However, they cannot be directly outsourced –it has already been shown that both sensors are useful to reveal private information such as the user's PIN code [13]. To address this limitation, in this paper SmartCAMPP, an outsourced privacy-preserving CA mechanism, is proposed. SmartCAMPP is designed for scenarios in which the authentication decision is taken on an external, untrusted third party. To foster its adoption in current out-

[☆] Editor: Modesto Castrillón Santana

[*] Corresponding author.

E-mail addresses: luis.hernandez@csic.es (L. Hernández-Álvarez), jfuentes@inf.uc3m.es (J.M. de Fuentes), lgmanzan@inf.uc3m.es (L. González-Manzano), luis@iec.csic.es (L. Hernández Encinas).

[1] https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/, last access sept. 2020

https://doi.org/10.1016/j.patrec.2021.04.013

sourced settings, SmartCAMPP leverages a cryptographic technique, called Format-Preserving Encryption or FPE [14,15], that ensures that transformed data keeps the format of the original one, so the same database and computational components can be applied to operate with this new information. Consequently, this work offers the following contributions:

- A privacy-preserving CA approach which, for the first time, applies machine learning techniques over encrypted sensorial data using FPE.
- Experimental analysis on a large-scale database containing real-world user-related data [16] to allow the comparison between FPE and non–FPE schemes, and establish a baseline for future investigations.
- Experimental material released in GitHub to foster further research and ensure its repeatability.

The remainder of this paper is as follows. Section 2 describes the related work. SmartCAMPP is introduced in Section 3. The assessment is shown in Section 4. Lastly, Section 5 concludes the paper and points out future research lines.

2. Related work

Many proposals have worked towards CA approaches applying assorted features. For instance, gyroscope and accelerometer are combined in [17,18]. [19] further combines them with orientation and magnetometer. On the other hand, [20] prefers other features such as GPS and other proposals like [21] work with screen touches. Specially focused on managing access control in cloud based applications, [22] fuses accelerometer, magnetometer and gyroscope data through a trust manager and privacy protocols are envisioned as a future step. To get deeper in CA approaches, an extensive review about the state of the art of CA, user profiling, and related biometric databases has been recently published [10]. For the sake of brevity, this section focuses on privacy-preserving CA approaches.

Despite the significant amount of CA approaches, privacy preservation has not received much attention in this context. Privacy-preserving approaches can be divided into a pair of categories [23], namely those in which an external server stores templates to be used in the authentication process, and those in which the transformed version of the template is stored within the authentication device.

Concerning the former type and in line with the proposed approach, homomorphic encryption is a common technique. It enables the computation on encrypted data without accessing the secret/private key. Data can be outsourced while being encrypted and distances, e.g., Hamming distance [5], or mathematical operations, e.g. addition [24], are computed to spot illegitimate users.

With respect to template protection techniques, their goal is to securely store the users' template generated in the enrollment phase of the authentication process [25]. The use of biometric hashes (called biohash) is a template protection technique. A hash of the biometric/behavioral data, commonly using a key, is computed and stored in the external server to be compared with the users' one in the authentication process, for instance, through a classifier [26]. Multi-party computation is other privacy-preservation approach and garbled circuits have been used in CA [23]. In this case, a circuit (i.e., a function that can be executed in a privacy-preserving way) is generated from the input data and both server and client verify its correctness using, for instance, the Euclidean distance. Finally, data anonymization could be useful for users' privacy protection [27], but the lack of data granularity and its possible de-anonymization should be studied.

Table 1 shows a comparison of privacy-preserving CA approaches in smartphones. Assorted features have been applied, like GPS [24] or keystroke dynamics [27]. Algorithms to achieve privacy are also varied, being homomorphic encryption the most common [5,24]. [27] proposes the simplest approach, where anonymization based on removing sensitive data is applied. However, given the amount of de-anonymization techniques, the probability of recovering private data from publicly available information should be carefully studied. This is the case, for instance, of WiFi data [28]. In terms of continuously authenticating users, classifiers are commonly used [11], but more alternatives appear when privacy is at stake. They are quite assorted, being the comparison between the stored template and the received one through some distance metric [5,23] or a threshold value [27] a common choice. Finally, the size (i.e., time span and amount of users) of the CA dataset in the evaluation process is specially relevant due to the continuity of the process. Just [26] and [5] carry out experiments on appropriate datasets for CA, though [26], in spite of the amount of users, does not really specify the time span per user, and [27] points out that, on average, 516.04 feature vectors are used, which could be enhanced to have a higher dataset.

As compared to existing works, SmartCAMPP is the first privacy-preserving approach for CA that applies gyroscope and accelerometer, features which have already been significantly studied in non-privacy-preserving CA, using format-preserving encryption for protecting users' data and a classifier in the authentication process.

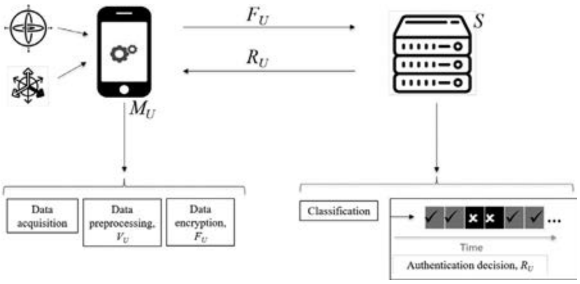


Fig. 1. SmartCAMPP overview.

3. Proposed mechanism

This Section introduces the proposed mechanism. In particular, Section 3.1 includes the problem formalization and Section 3.2 presents the stakeholders of the mechanism and its threat model. Afterwards, Section 3.3 describes its goals, whereas Section 3.4 describes its messages and interactions.

3.1. Problem formalization

The formalization of the problem includes an overview of the proposed mechanism and a description of the data features to use. The general scheme is shown in Fig. 1, where U is the user, M_U his mobile device, S the server, F_U the encrypted data sent to S, and R_U the authentication decision.

Input data, V_U, is collected from gyroscope and accelerometer and defined by three spatial coordinates (x, y, z) whose ranges are shown in Table 4. Such data is preprocessed, due to the characteristics of the used FPE system (see Section 3.4). Float numbers turn into integers removing the decimal point and then, FPE is applied leading to F_U. In Table 2, as an example, the first feature before and after the encryption process for the first two users is presented. After receiving the encrypted data, S analyzes it and generates R_U which is sent to M_U. A more detailed description is presented in the following sections.

Table 1
Related work comparison.

	Biometric feature	Data protection alg.	Authentication technique	Dataset size
[24]	GPS, WiFi session duration, time start charging the battery, num. users interactions with device	Homomorphic and Order Preserving encryptions	Homomorphic operations	-
[26]	Location, call information	Biohashing	SVM, KNN	100 users/ 1 month data
[5]	Screen touches	Homomorphic encryption	Euclidean and Manhattan distance	41 users/ 21,158 data vectors in total
[27]	Keystroke dynamics and mouse movements	Data anonymization	Similarity score	15 users/1 session 45min per user
[23]	Battery	Garbled circuit	Circuit evaluation (Manhattan or Euclidean distance) and oblivious transfer	1 smartphone
SmartCAMPP	**Accelerometer and gyroscope**	**Format-preserving encryption**	**SVM**	**50 users/ 120,000 samples, 500 h. per user**

Table 2
Example of user- and provider-based key encryption.

Original Data	Key	Encrypted Data
71.2604	6h0nh06diqq32ugqkakl	434858
180.6605	6h0nh06diqq32ugqkakl	9326781
1184.2775	6h0nh06diqq32ugqkakl	7708607
71.2604	ebnic0adefju9otfeane	742147
180.6605	7fyl8imtggs5z9zy3n9w	6115133
1184.2775	faew3gek1l025jcacm5h	91757017

3.2. Model

There are a pair of stakeholders in SmartCAMPP, whose capabilities are described as follows:

- The user, U, whose behavioural biometric information is collected for authentication purposes. Such information is gathered from his mobile device, M_U, in particular from its sensors, i.e. accelerometer and gyroscope.
- A server, S, processes data received by M_U in a continuous way, and provides an authentication decision to allow U to get access (or not) to M_U.

Concerning the threat model, the following attackers are pointed out:

- An external attacker who has the intention of stealing information. It could be collected through the robbery of M_U or sniffing data sent to S.
- Honest-but-curious server, that is S, which can be understood as a legitimate participant that will follow the established protocol but will try to learn all possible information from received messages [29].

3.3. Goals

The proposed mechanism has to fulfill three main goals:

- **Privacy preservation**. Sensor data cannot be accessed by any party except from U.
- **Efficiency**. The authentication decision, R_U, has to be taken with short delay. Therefore, SmartCAMPP must be suitable to be run in a continuous fashion.

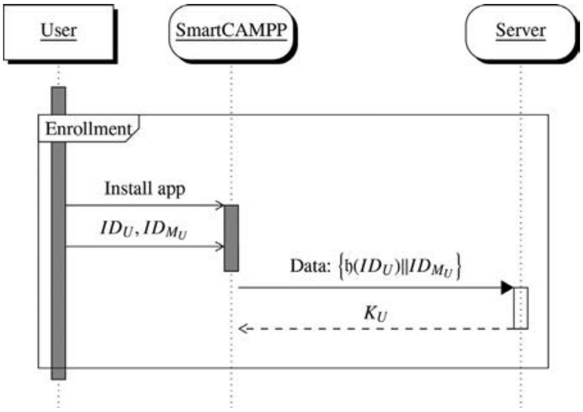


Fig. 2. Enrollment Protocol.

- **Accuracy**. The provision of the two aforementioned goals must not substantially increase (e.g., $< 10\%$ of difference) the amount of authentication errors, as compared to a non-privacy-preserving approach.

3.4. Description

SmartCAMPP is a mechanism that is continuously and interactively run between M_U and S, as depicted in Fig. 2. Their interaction involves a set of exchanged messages, which will be sent over an encrypted channel by means of usual network protection techniques (such as TLS [30]). As other applications, S only knows a pseudo-identifier of the user, $\mathfrak{h}(ID_U)$, where $\mathfrak{h}$ is a secure hash function, and ID_U the identifier of M_U (e.g., Android Advertising ID). SmartCAMPP is then run after the user enrolls in the system, for example by downloading the corresponding app.

Before its proper operation, it is necessary to train the classifier in S. In particular, several steps are carried out (see Fig. 3). Note that this approach is focused on the use of the cloud and then, in case of unavailability, other authentication technique, e.g. password or fingerprint, should be used instead.

Data acquisition. Firstly, M_U retrieves sensor data from accelerometer and gyroscope and stores them for further processing.

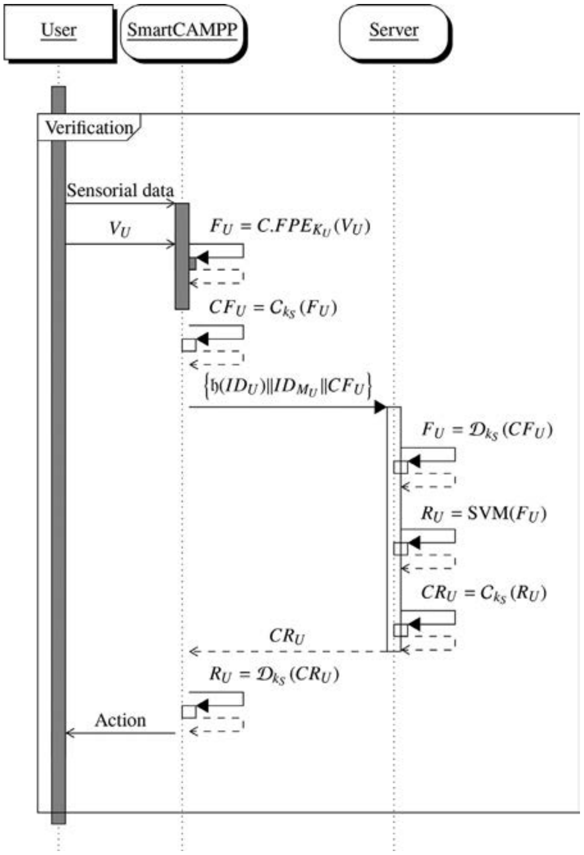


Fig. 3. Verification Protocol.

This step is carried out regularly, following the sampling rate of sensors at stake.

Data preprocessing. Each sensorial reading is composed by a set of features. Thus, in this step the selection of f features $V_U = \{v_1, \ldots, v_f\}$ is carried out per reading.

Data encryption. The set of selected features, V_U, is encrypted by means of FPE, a type of algorithm that ensures that both output (ciphertext) and input (plaintext) are in the same format, that is, both belong to the same domain [14,15]. Bellare *at al.* provide a comprehensive treatment of the FPE problem, using what they called type-1 and type-2 Feistel networks [14]. Based on their work, this paper applies FFX mode of operation for FPE [15], introduced herein. The encryption algorithm takes a key K, a plaintext X, and a tweak T. The plaintext is taken over an arbitrary alphabet *Chars*. Assuming that $n = |X|$ is a supported length, the encryption produces a ciphertext $Y = FFX.EncryptK_K^T(X) \in Chars^n$. The recovery of X from Y is computed by $X = FFX.DecryptK_K^T(X)$. For this application the use of FPE is preferred, as any other symmetric cryptosystem would near-randomly transform the original data into a binary file, denaturalizing the information and complicating its use in artificial intelligence techniques. Moreover, with this technique there is no need to modify the structure of the database to be used and, therefore, the computational processes are simplified, and allows the use of a cloud architecture (if necessary). Additionally, as it is proven with this proposal, the use of FPE can be added to an already existing CA procedure.

In this work, V_U is encrypted via FPE, producing $F_U = C.FPE_K(V_U)$. In order to implement SmartCAMPP in a real-world scenario, two alternative settings may come into play concerning the key for FPE. In the first one, called *provider-based keying*, the encryption key is the same for all users (K). Such key would be distributed by a key distribution center *KDC* using a secure keying protocol [31]. *KDC* is a trusted third party in charge of distributing encryption keys to all M_U. In the real world, this can be implemented through certification authorities or as part of smartphone app stores (e.g., Google Play). By contrast, in the second case, called *user-based keying*, each user sets its own key (K_U). Thus, $F_U = C.FPE_K(V_U)$ in the first case, whereas $F_U = C.FPE_{K_U}(V_U)$ in the second one.

Classification. The result of the previous encryption, F_U, is then encrypted again by M_U for S with a shared key, k_S, obtained by, for example, a hybrid cryptosystem and gets $CF_U = \mathcal{C}_{k_S}(F_U)$. S decrypts the received data, CF_U, with the shared key, k_S, and obtains F_U. After its normalization, it will be the input to a classifier, that will be used either for training (while setting up the CA mechanism) or for actually taking a decision R_U, that is, either S determines that M_U is being held by U or by an intruder. This decision is sent back to M_U leveraging the previously established encrypted channel.

Although a plethora of alternatives are available, in this study three classifiers are explored: Support Vector Machine (SVM), Random Forest (RF), and Logistic Regressor [32], [33]. These are well-known machine learning models that provide several implementations options in their learning process, and their utility for CA has been demonstrated, specially with motion and biometric features (see [11], [10] and the references included therein).

4. Evaluation

This Section focuses on the assessment of the proposal. In this regard, Sections 4.1 and 4.2 present the experimental settings and the data preparation issues. For the sake of comparison against SmartCAMPP, Section 4.3 shows the performance of a non-privacy-preserving CA mechanism, which serves as a baseline. Lastly, the achievement of SmartCAMPP goals are addressed in Sections 4.4 (privacy), 4.5 (performance) and 4.6 (accuracy), respectively.

4.1. Experimental settings

This subsection describes the dataset used in the experiments as well as the developed prototype implementation.

Data from 52 users was retrieved from the Sherlock database [16]. In what comes to motion sensors, the information of each subject is composed by 90 features, 48 from the accelerometer and 42 from the gyroscope. Data was collected every 15 seconds for a period of 3 years. However, due to acquisition issues (i.e., void readings) and the lack of continuity for all users, the data involved in the evaluation consists of 120,000 samples per user, i.e., 500 hours, for 50 users.

All experiments have been carried out on Intel Core i7 at 2.00 GHz and 16 GB of RAM. Cryptographic libraries (pyffx[2] [15]) were run on Python 3.7. For encryption, keys composed of 20 random alphanumeric symbols, namely 10 characters and 10 digits with repetitions were considered. To foster further research in this direction, our prototype implementation has been released[3]. It must be noted that all experiments have been repeated three times for the sake of soundness. In each one, a different test set was built to prevent over-fitting.

4.2. Data and classifier preparation

This Section describes the feature selection process and the applied classifier settings.

[2] https://pypi.org/project/pyffx/

[3] https://github.com/luishalvarez/Smart-CAMPP.git

Table 3
Training Features Analysis.

No. features (acc + gyr)	$10(5+5)$	$15(8+7)$	$20(10+10)$
Average Accuracy (%)	82.12	82.79	81.43
Standard Deviation	7.88	10.19	10.58
Max Accuracy (%)	88.79	93.45	93.49
Min Accuracy (%)	61.44	59.54	58.03

4.2.1. Feature selection

To reduce computational costs, authentication time, and the quantity of transmitted data, the number of sensorial features used in the classification process was reduced. The most relevant features V_U were chosen based on a Chi-square statistical test. For this analysis, a subset of 10 users, with training and test sets sizes of 10,000 and 1,000 samples per label, respectively, and a SVM model with Radial Basis Function (RBF) kernel and default configuration are used. For the sake of brevity, Table 3 shows results for 10, 15 and 20 features. The best average accuracy in this study was achieved with 15 features, from which 8 belong to the accelerometer and 7 to the gyroscope (see Table 4).

4.2.2. Classifier settings

In these experiments, the three machine learning techniques (SVM, RF and LR, recall Section 3.4) have been considered to assess the non-privacy-preserving mechanism (see Section 4.3). Afterwards, the best one was selected to evaluate SmartCAMPP. In the following, we describe the choices made in what comes to classifier parameters and the training and testing period.

Parameterization. Two parameters were cross-validated for each classifier: $C = \{0.1; 1; 10; 100; 1,000\}$ and $gamma = \{"auto"; "scale"; 0.001; 0.01; 0.1\}$, for SVM; $max\_depth = \{10; 30; 50; 70; 100\}$ and $n\_estimators = \{200; 500; 1,000\}$, for RF; and $C = \{0.1; 1; 10; 100; 1000\}$ and $solver = \{"newton-cg"; "lbfgs"\}$ for LR [32].

Train and test periods. For the sake of real-world suitability, a train-test distribution of 15:85% and training times of 48 and 72 hours were set. These settings are in line with the expected usage of the system, which should be ready to operate after a reduced period (i.e., shortly after the user buys the device). Moreover, both the training and test sets contain as much positive samples (from the user under study) as negative ones (from any of the other users, taken at random).

To determine the most suitable training time, an analysis was conducted for all users in a one-vs-all configuration. Thus, the dataset was processed to assign a label to the user at stake and a different one (always the same) to all remaining users. The experiments of this study were carried out leveraging a RBF kernel-based SVM considering the parameters stated above. The results are presented in Table 5 and demonstrate that 48 hours is the best training size.

Other works [2] justify the use of one-class classifiers for CA. However, this option was discarded due to three reasons: 1) One-class classifiers are used when the positive samples do not present a defined structure, as this complicates the establishment of a class boundary. For the biometric features used in this study we expect a pattern along time for each user [34]. 2) The use of external users' data motivates the construction of a cloud-based infrastructure, which is appropriate to study the classification performance with encrypted data. 3) The experiments conducted with a one-class SVM revealed that more training data is required to obtain the accuracy provided by the baseline SVM (see Table 6). In fact, training data with 50,000 samples per user produces an accuracy of 73.02%, and this implies more memory resources and an increased training time of 37 min. per subject.

4.3. Baseline – Non-privacy-preserving CA

Opposed to SmartCAMPP, a non-privacy-preserving CA mechanism, referred as baseline, does not apply any kind of encryption. Features V_U are sent in the clear to S both at training time and during the authentication process.

Table 6 presents accuracy, standard deviation, and Equal Error Rate (ERR) of the baseline mechanism leveraging the three classifier techniques (SVM, RF and LR). The EER is defined as the value of False Positive Rate (FPR, access allowed to unauthorized user) when it is equal to the value of False Negative Rate (FNR, access denied to an authorized user), and defines the optimal point of the ROC (Receiver Operating Characteristic) curve. For clarity purposes, data from the three repetitions is shown, considering the optimal parameters for each classifier after cross correlation (recall Section 4.2.2). RF provides better outcomes than LR, but both are worse than SVM. In particular, the average accuracy of SVM is 1.31% better than RF, and 8.35% better than LR, while the average EER of SVM is 1.89% better than RF, and 9.22% better than LR. In terms of standard deviation, LR presents the smallest one, followed by SVM and RF. However, assuming a worst case considering the standard deviation values, SVM still achieves the best accuracy results, that is 70.53%, while RF reaches 67.97%.

In the light of these results, SVM is the best option so it will be used to confront against SmartCAMPP (see Section 4.4). However, it is also interesting to assess the results of the cross–validation process of C and $gamma$. Table 7 shows the amount of users with optimal results for each configuration parameter. Thus, the best values of these parameters are the greatest possible ones, i.e., $C = 1,000$ and $gamma = 0.1$. This indicates that, for this application, it is beneficial to define a severe model, focused on avoiding false positives, and interpreting each training example with a high influence [32,33]. This fact makes sense, as for CA purposes, it is preferable to fail with a false rejection than with a false acceptance.

4.4. Privacy preservation analysis

SmartCAMPP is suitable to protect the users privacy leveraging FPE. Considering the search space of all keys (recall Section 4.1), the number of possible keys is $VR_{36}^{20} = 36^{20} \approx 2^{103}$. This value is equivalent to the number of keys of 103 bits, which implies an adequate security for the considered application [35,36].

It is important to note that this key size is a measure of the required brute-force effort that S should carry out to reveal M_U's sensor readings. For external attackers, the effort would be higher as all communications between S and M_U take place leveraging an encrypted channel (recall Section 3.4). Therefore, they would need to break both encryption keys to succeed.

With respect to the privacy offered by each of the proposed keying schemes, namely *user-based* or *provider-based*, the former is more convenient as users will not have to share the same key and so attackers would need to guess K_i for each user. This also prevents S from getting access to K, which would not always be realistic in all settings.

4.5. Performance analysis

There are two resource-intensive tasks in SmartCAMPP, namely the encryption and the classifier operation. Therefore, each one is studied in the following.

Concerning the encryption time, data from 10 users was encrypted using random keys. Given that this task is carried out in a smartphone, the laptop was instructed to lower down the processor speed to 1.6 Ghz., which is in line with current mid-range

Table 4
Accelerometer and gyroscope selected features.

Range (Hz)	Feature Name	Meaning
0–10042.01 0–10041.70 0.003–7295.92	acc_stat_x_dc_fft acc_stat_y_dc_fft acc_stat_z_dc_fft	The DC component of the FFT on the *x*-axis (resp. *y*-axis and *z*-axis)
0–87766.28 0–150522.03 0–108098.34	acc_stat_x_var_fft acc_stat_y_var_fft acc_stat_z_var_fft	The variance of the FFT values obtained from *x*-axis (resp. *y*-axis and *z*-axis) frequencies
0–166.60	acc_stat_z_mean_fft	The average energy across the FFT components on the *z*-axis
0–4453-317	acc_stat_cov_y_x	The *y*-*x* covariance of the sampled values
0–256 0–256	gyr_stat_x_fourth_idx_fft gyr_stat_z_fourth_idx_fft	The index (frequency) of the FFT with the fourth most energy on the gyroscope *x*-axis (resp. *z*-axis)
0–28538.37 0–15909.89 0–17680.36	gyr_stat__ x_ var __ fft gyr_stat__ y__ var__ fft gyr_stat_ z_ var_ fft	The variance of the FFT values obtained from *x*-axis (resp. *y*-axis and *z*-axis) frequencies
0–3041.09 0–1899.45	gyr_stat__ y__ dc__ fft gyr_stat__ z__ dc__ fft	The DC component of the FFT on the gyroscope *y*-axis (resp. *z*-axis)

Table 5
Training Size Analysis.

Training Set (hours; samples/label)	$48; 11,520$	$72; 17,280$
Test Set (hours; samples/label)	$272; 65,280$	$408; 97,920$
Average Accuracy (%)	81.39	79.08
Standard Deviation	12.29	14.07
Max Accuracy (%)	97.74	96.33
Min Accuracy (%)	51.94	50.71

smartphones available in the market[4]. On average, each encryption operation requires 5.05 ms. per sensorial reading.

With respect to the classifier performance, there are two times at stake: training and operation. The time needed to train a model in the baseline (i.e., non-privacy-preserving) approach is 8 min. per subject, while it increases to 26 min. per subject in SmartCAMPP. These results follow expectations, since when data is encrypted, it loses part of its significance and becomes quasi-random. On the other hand, the time taken to authenticate a user is 0.13 ms. per sensorial reading in the baseline mechanism and 0.2 ms. in SmartCAMPP.

Thus, the overhead introduced by SmartCAMPP in operational mode (that is, after training) is 5.12 ms. per reading. Therefore, the performance overhead is negligible for the proposed context. On the contrary, the training time is significantly higher, but it is irrelevant as a period of 48 hours has been set for the training phase.

Finally, the efficiency of SmartCAMPP is compared with [24], for being, to the authors knowledge, the only CA privacy-preserving proposal focused on reducing computation time. In [24] the authentication, in an optimized setting, takes 54 ms. in the device side, which is far from the 5.25 ms required in SmartCAMPP, where authentication involves 5.05 ms. in M_U and 0.2 ms. in S.

4.6. Accuracy analysis

SVM is the applied technique to assess SmartCAMPP, as it was the best alternative in the non-privacy-preserving scenario. Table 8 depicts accuracy results when data is encrypted with a key for each user (*user-based keying*) and with a single key for all of them (*provider-based keying*). The best accuracy is achieved in the first alternative (76.85%), thus distinguishing both in 5.18% and less than 2 in terms of standard deviation.

None of the settings of SmartCAMPP is able to reach the baseline accuracy (82.28%, recall Section 4.3). This is because increasing the security of the data, by encrypting it, produces a decrease in the authentication performance. This trade-off will always be present, since the encryption process distorts the original information. However, the difference in accuracy between the baseline and the *user-based keying* SVM is only of 5.4%, which is not a dramatic reduction considering that it is a privacy-preserving approach. Besides, as in the baseline case, the best values of the parameters C and *gamma* are 1,000 and 0.1, respectively (cf. Table 7).

As a result, SmartCAMPP *user-based keying* is the most appropriate alternative and not just in terms of accuracy and EER, but also considering that any additional entity (e.g. *KDC*) has to be involved in the model. Then, a detailed accuracy analysis is carried out constructing ROC curves and studying EER for each user in the baseline mechanism (Fig. 4) and in the SmartCAMPP *user-based keying* one (Fig. 5). These curves were built by increasing the authentication decision threshold from 0 (all users considered owners) to 1 (none user considered owner) in steps of 4×10^{-5}, and the EERs were calculated by intersecting each ROC curve with the line that joins the points (0,1) and (1,0). On average, EER and standard deviation are 17.16% and 13.30% in the baseline mechanism and 23.24% and 10.26% in the *user-based keying* one. Results between both pair of mechanisms differ in 6.08%, being the standard deviation lower in the latest. Comparing the ROC curves of each mechanism, some specific users present a strange behavior in

[4] https://nanoreview.net/en/soc-list/rating, last access October 2020.

Table 6
Baseline authentication results.

Model	SVM			RF			LR		
	Avg. Acc (%)	St. D	EER (%)	Avg. Acc (%)	St. D	EER (%)	Avg. Acc (%)	St. D	EER (%)
1st	82.24	11.74	17.18	80.99	12.99	19.58	73.92	8.86	26.36
2nd	82.25	11.75	17.17	80.98	12.98	19.56	73.94	8.84	26.32
3rd	82.30	11.71	17.13	80.88	12.97	19.72	73.87	8.85	26.46
Overall	82.26	11.73	17.16	80.95	12.98	19.62	73.91	8.85	26.38

Table 7
Cross–validation results for SVM.

C	# Subjects (Baseline)	# Subjects (SmartCAMPP)	*gamma*	# Subjects (Baseline)	# Subjects (SmartCAMPP)
0.1	1	0	auto	14	12
1	1	0	scale	6	13
10	4	5	0.001	0	0
100	8	22	0.01	2	1
1000	36	23	0.1	28	24

Table 8
SmartCAMPP accuracy.

Model	User-based keying			Provider-based keying		
	Avg. Acc (%)	St. D	EER (%)	Avg. Acc (%)	St. D	EER (%)
1st	76.77	10.88	23.31	71.71	8.96	29.33
2nd	76.64	11.05	23.38	71.61	8.96	29.34
3rd	77.12	10.66	23.03	71.67	9.06	29.41
Overall	76.84	10.86	23.24	71.66	8.99	29.36

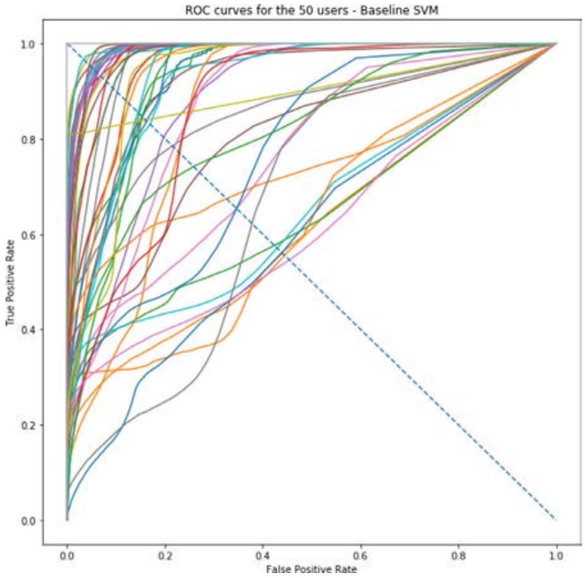


Fig. 4. ROC curves of each user for baseline SVM.

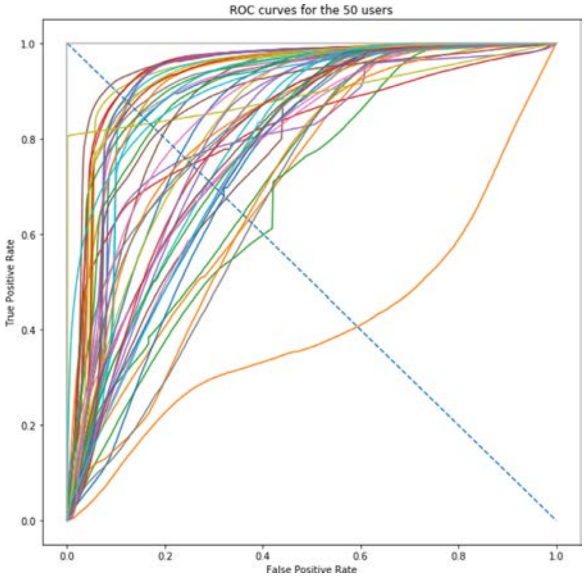


Fig. 5. ROC curves of each user for *user-based keying* CA.

both cases, slightly mitigated in the *user-based keying* mechanism. For example, there is an orange curve in the *user-based keying* case whose behavior is clearly different from the others, but discarding this user the accuracy and EER are improved to 77.32% and 22.51%, respectively. Therefore, regardless of the approach, the behaviour follows the same pattern, though with slightly higher values when encryption takes place, but still showing the feasibility of using FPE.

Finally, SmartCAMPP *user-based keying* is compared to [26] and [5] in terms EER, for being the only proposals, to the authors knowledge, that offer privacy-preserving CA and compute this metric (see Section 2). Note that datasets and authentication features are not analogous, but the comparison is useful for illustration purposes given their common goal. In [5] the lowest EER is 22.50% and 45.77% the highest using Euclidean and Manhattan distance. Results are comparable or worse than in SmartCAMPP where the best EER is 23.03% and the worst 29.41%. This is a sensible result as [5] applies discretization and distances which may lead to data variations specially in continuous processes. On the contrary, in [26] EER is 10% on average and thus, better than in SmartCAMPP. This proposal works with biohashing generated from location and phone calls data and due to the hash property of generating a different output just with a small change in the input, this technique may not be feasible in behavioural features with extremely assorted values, e.g. gyroscope and accelerometer.

5. Conclusion. Future work

Continuous Authentication techniques are attracting attention of the research community. Given the growing relevance of smartphones, in this work we have proposed SmartCAMPP, a CA mechanism leveraging gyroscope and accelerometer information collected by the device. Although with slightly lower accuracy rate than in a non-privacy-preserving setting, SmartCAMPP achieves promising accuracy rates involving a negligible performance overhead.

SmartCAMPP is the first attempt to building a privacy-preserving CA approach for smartphones leveraging motion sensors. Hence, a wide array of alternatives exist for future works. As such, exploring the use of other sensorial information collected by smartphones, or determining the impact of other encryption techniques and artificial intelligence tools (Recurrent or pre-trained Neural Networks), are envisioned work directions. Also, the use of other databases, like HMOG [2], to confirm results, as well as the analysis of SmartCAMPP performance with special attention to users' activities (e.g. running, sleeping, etc.) are limitations of our proposal and interesting future research lines.

Declaration of Competing Interest

No.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments. This work was partially supported by Spanish MINECO, AEI and European Regional Development Fund (ERDF), through grants TIN2017-84844-C2-1-R (COPCIS) and PID2019-111429RBC21 (ODIO); by Comunidad de Madrid (Spain) through grant P2018/TCS-4566-CM (CYNAMON), co-funded with ERDF, and also jointly with Univ. Carlos III de Madrid, grant CAVTIONS-CM-UC3M. Lorena González and José María de Fuentes would like to thank the Excellence Program for University Researchers. Luis Hernández-Álvarez would like to thank CSIC Project 202050E304 (CASDiM).

References

[1] Crowdstrike, Use of fancy bear Android malware in tracking Ukranian field artillery units, 2017, (Crowdstrike Global Intelligence Unit).

[2] Z. Sitová, J. Sedenka, Q. Yang, G. Peng, G. Zhou, P. Gasti, K. Balagani, Hmog: new behavioral biometric features for continuous authentication of smartphone users, IEEE Trans. Inform. Forens. Secur. 11 (5) (2015) 877–892.

[3] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, Deep learning for sensor-based activity recognition: a survey, Pattern Recognit. Lett. 119 (2019) 3–11.

[4] M. Sun, W.P. Tay, Decentralized detection with robust information privacy protection, IEEE Trans. Inform. Forens. Sec. 15 (2019) 85–99.

[5] S. Govindarajan, P. Gasti, K.S. Balagani, Secure privacy-preserving protocols for outsourcing continuous authentication of smartphone users with touch data, in: 2013 IEEE Sixth Intl. Conf. on Biometrics: Theory, Applications and Systems (BTAS), 2013, pp. 1–8.

[6] M. Al-Rubaie, Towards privacy-aware mobile-based continuous authentication systems, Iowa State University, 2018 Ph.D. thesis.

[7] C. Shen, Y. Tianwen, S. Yuan, Y. Li, X. Guan, Performance analysis of motion-sensor behavior for user authentication on smartphones, Sensors 16 (3) (2016) 345.

[8] G. Wu, J. Wang, Y. Zhang, S. Jiang, A continuous identity authentication scheme based on physiological and behavioral characteristics, Sensors 18 (1) (2018) 179.

[9] M.N. Malik, M.A. Azam, M.E. ul Haq, W. Ejaz, A. Khalid, ADLAuth: Passive authentication based on activity of daily living using heterogeneous sensing in smart cities, Sensors 19 (11) (2019) 2466.

[10] L. Hernández-Álvarez, J.M. De Fuentes, L. González-Manzano, L. Hernández Encinas, Privacy-preserving sensor-based continuous authentication and user profiling: a review, Sensors 21 (2021) 92.

[11] L. Gonzalez-Manzano, J.M.D. Fuentes, A. Ribagorda, Leveraging user-related internet of things for continuous authentication: a survey, ACM Computing Surveys (CSUR) 52 (3) (2019) 1–38.

[12] M. Ehatisham-ul Haq, M.A. Azam, U. Naeem, Y. Amin, J. Loo, Continuous authentication of smartphone users based on activity pattern recognition using passive mobile sensing, Journal of Network and Computer Applications 109 (2018) 24–35.

[13] D. Berend, S. Bhasin, B. Jungk, There goes your pin: Exploiting smartphone sensor fusion under single and cross user setting, in: 13th Intl. Conf. on Availability, Reliability and Security, 2018, pp. 1–10.

[14] M. Bellare, T. Ristenpart, P. Rogaway, T. Stegers, Format-preserving encryption, in: Intl. Workshop on Selected Areas in Cryptography, 2009, pp. 295–312.

[15] M. Bellare, P. Rogaway, T. Spies, The FFX Mode of Operation for Format-Preserving Encryption, Technical Report, Submited to NIST, 2010.

[16] Y. Mirsky, A. Shabtai, L. Rokach, B. Shapira, Y. Elovici, Sherlock vs Moriarty: A smartphone dataset for cybersecurity research, in: 2016 ACM Workshop on Artificial Intelligence and Security, 2016, pp. 1–12.

[17] Y. Li, H. Hu, G. Zhou, Using data augmentation in continuous authentication on smartphones, IEEE Internet Things J. 6 (1) (2018) 628–640.

[18] W.-H. Lee, R.B. Lee, Implicit smartphone user authentication with sensors and contextual machine learning, in: 2017 47th Annual IEEE/IFIP Intl. Conf. on Dependable Systems and Networks (DSN), IEEE, 2017, pp. 297–308.

[19] C. Shen, Y. Li, Y. Chen, X. Guan, R.A. Maxion, Performance analysis of multi-motion sensor behavior for active smartphone authentication, IEEE Transactions on Inf. Forensics and Security 13 (1) (2017) 48–62.

[20] L. Fridman, S. Weber, R. Greenstadt, M. Kam, Active authentication on mobile devices via stylometry, application usage, web browsing, and GPS location, IEEE Syst. J. 11 (2) (2016) 513–521.

[21] R. Rocha, D. Carneiro, P. Novais, Continuous authentication with a focus on explainability, Neurocomputing (2020).

[22] G. Fenu, M. Marras, Controlling user access to cloud-connected mobile applications by means of biometrics, IEEE Cloud Comput. 5 (4) (2018) 47–57.

[23] P. Gasti, J. Šeděnka, Q. Yang, G. Zhou, K.S. Balagani, Secure, fast, and energy-efficient outsourced authentication for smartphones, IEEE Trans. Inform. Forens. Sec. 11 (11) (2016) 2556–2571.

[24] S.F. Shahandashti, R. Safavi-Naini, N.A. Safa, Reconciling user privacy and implicit authentication for mobile devices, Computers & Security 53 (2015) 215–233.

[25] A.K. Jain, K. Nandakumar, A. Nagar, Biometric template security, EURASIP J. Adv. Signal Process. 2008 (2008) 1–17.

[26] J. Hatin, E. Cherrier, J.-J. Schwartzmann, C. Rosenberger, Privacy preserving transparent mobile authentication, in: Intl. Conf. on Inf. Systems Security and Privacy, 2, 2017, pp. 354–361.

[27] Y. Sun, S. Upadhyaya, Secure and privacy preserving data processing support for active authentication, Inf. Syst. Front. 17 (5) (2015) 1007–1015.

[28] A. Di Luzio, A. Mei, J. Stefa, Mind your probes: De-anonymization of large crowds through smartphone WiFi probe requests, in: 35th Annual IEEE Intl. Conf. on Computer Communications, 2016, pp. 1–9.

[29] A. Paverd, A. Martin, Modelling and Automatically Analysing Privacy Properties for Honest-but-Curious Adversaries, Technical Report, University of Oxford, 2014.

[30] E. Rescorla, T. Dierks, The transport layer security (TLS) protocol, version 1.3, 2018, (Request For Comments 8446).

[31] Y. Slimane, K. benahmed, Y. Benslimane, Efficient end-to-end secure key management protocol for internet of things, Int. J. Electr. Comput. Eng. 7 (6) (2017) 3622–3631.

[32] C.M. Bishop, Pattern recognition and machine learning, Springer, 2006.

[33] T. Hastie, R. Tibshirani, J. Friedman, The elements of statistical learning: Data mining, inference, and prediction, 2nd, 2009.

[34] A. Fernández, S. Garcia, M. Galar, R. Prati, B. Krawczyk, F. Herrera, Learning from Imbalanced Data Sets, Springer, edition, 2018. 10.1007/978-3-319-98074-4.

[35] ISO/IEC, Information technology – Security techniques – Lightweight cryptography – Part 1: General, Technical Report, Intl. Organization for Standardization/Intl. Electrotechnical Commission, 2014.

[36] W.J. Buchanan, S. Li, R. Asif, Lightweight cryptography methods, J. Cyber Secur. Tech. 1 (3–4) (2017) 187–201.