Privacy Preserving Transparent Mobile Authentication

Julien Hatin[1,2], Estelle Cherrier[2], Jean-Jacques Schwartzmann[1,2] and Christophe Rosenberger[2]
[1]*Orange Labs, 14000 Caen, France*
[2]*Normandie Univ., ENSICAEN, UNICAEN, CNRS, GREYC, 14000 Caen, France*
{julien.hatin, jeanjacques.schwartzmann}@orange.com, {estelle.cherrier, christophe.rosenberger}@ensicaen.fr

Keywords: Transparent Authentication, Privacy, Cancellable Biometric, Smartphone, Online Authentication.

Abstract: Transparent authentication on mobile phones suffers from privacy issues especially when biometric information is involved. In this paper, we propose a solution to address those two issues using the Biohashing algorithm on behavioral information extracted from a mobile phone. The authentication scenario is tested on a dataset composed of 100 users and shows promising results with a 10% EER in the worst case scenario (i.e when protection key is compromised) and a 1% EER in the best case one. In addition, privacy concerns are discussed and experimentally evaluated both in a quantitative and qualitative ways. This opens new perspectives concerning online authentication using smartphone sensing abilities.

1 INTRODUCTION

Mobile phone security is becoming a predominant issue as more and more online applications and services are now available for smartphones. Often those applications consider having the smartphone is sufficient to access the service (Grosse and Upadhyay, 2013) and use the *remember me* function. When more security is needed, or when it is needed to authenticate on another device than the smartphone, burden is added to the user by asking him to remember a password. To minimize this burden, users often choose the same password for multiple services and easy-to-guess passwords like 123456 (Vance, 2010). An alternative is to use transparent authentication. It permits to seamlessly authenticate an user. To explain what transparent authentication is, we quote Nathan Clarke (Clarke, 2011):

Definition 1 (Transparent Authentication)**.** *Transparent authentication can be achieved by any authentication approach that is able to obtain the sample required for verification non-intrusively.*

Behavioral biometrics is a perfect candidate for transparent authentication. Indeed, in this approach, authentication is based on the way someone does something. The authentication burden is removed when using behavioral biometrics because samples are recorded seamlessly.

In May 2016, Google announced the release of a continuous and transparent authentication mechanism to replace the {login, password} couple (Google, 2016). This solution is announced to be available by the end of the year as an API. However, concerns still occur (Patel et al., 2016) : (i) Often this kind of authentication are outsourced to companies that have the expertize in the field of transparent authentication and identity management; (ii) Unlike passwords, biometrics cannot be cancelled.

Besides, samples can be of any type including geolocation, application usage, web browsing history, emailing and many more, therefore important privacy concerns occur. This is even more true when the authentication task is outsourced. In addition, the cancellability of biometric data is an important security risk in case of theft of a large dataset and is an important brake to the deployment of online transparent authentication services.

Our contribution is a new authentication framework that is privacy preserving and cancellable. In addition, this new scheme does not add any burden to the user and can by applied on multiple samples issued from different smartphone sensors. The system is evaluated on a dataset containing data from 100 users collected during one month.

The organisation of the paper is as follows. We first describe the related works concerning behavioral authentication on mobile devices in section 2. Section 3 proposes a framework that uses behavioral samples and the Biohashing algorithm to protect privacy and make the data cancellable. Section 5 rounds off the paper with a conclusion and indications of future works.

Hatin, J., Cherrier, E., Schwartzmann, J-J. and Rosenberger, C.
Privacy Preserving Transparent Mobile Authentication.
DOI: 10.5220/0006186803540361
In *Proceedings of the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017)*, pages 354-361
ISBN: 978-989-758-209-7

2 RELATED WORKS

This part is dedicated to a short states of the art both for behavioral authentication and mobile authentication.

Behavioral authentication solutions that provide transparent authentication are a fast growing area. This is especially due to the Active Authentication project (Guidorizzi, 2013). The Defense Advanced Research Projects Agency offers to move beyond password by using transparent authentication mecanism. This means most users will authenticate themselves using biometrics sensors.

The authors of (Hayashi et al., 2013) proved that combining the location with a standard authentication increases the global trust in that authentication. In addition, this article shows that the main locations arising for a user are: (i) Home and (ii) Workplace. This implies to continuously know where the user is and therefore compromises users privacy. The location property and especially the one offered by GPS sensors embedded in modern smartphones represents discriminating features. In reference (Das et al., 2013), the authors record the daily activities (including location) of the user to ask him questions about its past day in order to authenticate himself. This implies to store private data about the user.

The authors in (Li et al., 2013) offer a solution to authenticate users using the geolocation and the phone calls. They obtain an EER of 5.4% with the 6 last phone calls. However, the privacy aspect is not taken into account. In (Saevanee et al., 2014), the authors cumulate different authentication modalities and also include the text message content. To proceed with the text message information, the messages must be read. This implies privacy leakage.

Less sensitive data can be exploited to perform behavioral authentication. This is the case of gait recognition (Derawi and Bours, 2013). However, the authors in (Tanviruzzaman and Ahamed, 2014) have shown that combining location information with gait recognition increases the global performance of the system. By combining those data, they obtained an ERR of 10% on a dataset of 13 users. However, privacy protection is not taken into account. Another approach is to use the swipe gesture patterns as proposed in (Mondal and Bours, 2013). The touch dynamics is one of the most commonly used methods to transparently authenticate users on smartphones.

Besides, mobile sensors are exploited to build frameworks for user authentication purpose as in reference (Witte et al., 2013). A probabilistic approach is performed in (Kayacik et al., 2014). In this paper, the authors propose to store data on the phone. This mitigates the privacy issues but the data saved on the mobile phone become an issue when the device has to be replaced: in this case, data have to be transferred or stored online. The authors in (Fridman et al., 2015) propose a solution that uses multiple informations that can be extracted from the mobile phones to monitor the behavior of the user. This monitoring includes text messages, web browsing habits, application usage and location of the user. The performances are promising with a FAR around 11% and a FRR around 6% for the location modality alone using an SVM classifier. By merging this modality with other behavioral modalities, authors are able to achieve an EER of 5% after 1 minute of user interaction with the device. Those results are obtained on a private database of 100 users. No privacy protection scheme is envisaged in this paper.

The privacy problem was noted in (Jakobsson et al., 2009). In this article, the authors advise to (i) remove unique identifier information, (ii) use pseudonyms, (iii) use aggregated data.

To the best of our knowledge, there are few proposed solutions in the literature to deal with privacy concerns. The authors in (Safa et al., 2014) use an homomorphic encryption scheme. In (Nauman et al., 2013) the authors describe a protocol dedicated to keystroke analysis. In (Chow et al., 2010), the authors address the problem of online authentication using implicit information and store the data directly on the mobile phone, thus delegating the autorization server role to the mobile phone. This permits to mitigate the privacy problem but does not solve the cancellability issue.

The main contribution of this paper is to propose a solution for mobile authentication enabling privacy protection and data cancellation.

3 PROPOSED APPROACH

The proposed solution merges different sensors information from the mobile phone and use the Biohashing algorithm to ensure both cancellability of sensitive data and privacy protection.

3.1 Biohashing

Biometric data are personal data, intrinsically non-revocable (unlike passwords, or tokens) and thus very sensitive data. The BioHashing algorithm is one particular example of cancelable biometrics techniques (Ratha et al., 2001; Bolle et al., 2002). The concept of cancelable biometrics relies on a transformation of the raw biometric data, enabling the transformed data to address both security and privacy protection issues. The general principle consists in the generation of a new biometric template, from the biometric feature vector and a secret random number. Therefore, it can be seen as a two-factor authentication scheme. Mobile authentication is particularly suited to perform cancelable biometrics based authentication: the mobile can easily capture various biometric modalities and at the same time the secret random number can be stored in the secure element (see section 3.2.2). As every biometric system, cancelable ones involve two steps.

- Enrolment step: once transformed, the new template (or transformed template) is stored as reference in the mobile phone, while the original raw biometric vector is discarded and never kept.
- Verification step: a comparison is performed between two transformed templates, namely between the transformed query and the transformed reference.

For an overview of the other existing cancelable biometrics schemes, we refer the reader to many survey papers (Rathgeb and Uhl, 2011), (Patel et al., 2015). BioHashing has been first described in (Goh and Ngo, 2003) and (Teoh et al., 2004), it has been respectively applied to face recognition and fingerprint recognition. The principle is as follows. The transformation function in BioHashing combines a user-specific secret key K with the biometric feature expressed as a fixed-length vector $x = (x_1, \dots, x_n) \in \mathbb{R}^n$. For more protection, the key K is stored in the secure element of the mobile phone (see section 3.2.2). The BioHashing process is divided into two steps:

- Random projection: the key K is used as a seed to generate m random vectors $r_j \in \mathbb{R}^n$, $j = 1, \dots, m$ and $m \leq n$. After orthonormalization by the Gram-Schmidt method [37], these vectors are gathered as the column of a matrix $O = (O_{i,j})_{i,j \in [1,n] \times [1,m]}$. A projection of the biometric feature vector $(f_1, \dots, f_n)$ is then computed.
- Quantization: This step is devoted to the transformation in a binary-valued vector of the previous real-valued vector using a simple thresholding. More precisely, a binary vector $B = (B_1, \dots, B_m)$ called BioCode is obtained from the previous projected vector by thresholding. The goal of this step is to guarantee the irreversibility of the whole process. It requires the specification of a threshold to compute the final BioCode.

Thus, by combining the high confidence of the key to the biometric data, the inter-class variation increases while the intra-class distance is preserved. Hence, good performances can be achieved, see (Belguechi et al., 2013a) for example.

We insist on the fact that BioHashing is privacy-preserving. Indeed, with BioHashing algorithm, as well as other cancelable biometrics techniques, original raw biometric data does not have to be stored. Therefore the users privacy is guaranteed. Moreover, revocability property is also automatically ensured: if the transformed template is compromised, the secret random key can simply be replaced to generate a new BioCode. Concerning the remaining properties among the aforementioned ones, unlinkability is also guaranteed together with the diversity, since distinct secret keys can be chosen for distinct applications, with no link between the generated BioCodes. Again, these properties have been deeply analyzed (for fingerprints) in (Belguechi et al., 2013a) and (Belguechi et al., 2013b).

In the context of the present paper, *i.e.* for transparent online mobile authentication, details are given in the following section 3.2 about the architecture, i.e, how we propose to implement BioHashing.

3.2 Architecture

The global architecture is composed of a client server application. In this paper, the mobile phone collects data about the user behavior when this one is calling. The proposed approach can be extended to other samples from different sensors from the one used in this work. It particularly well fits for merging geolocation samples with any other sensors or group of sensors.

3.2.1 Client Architecture

Smartphones have a great penetration into the market (Sophos,) and possesses heavy sensing capabilities. Within those sensors, we can find: accelerometers, gyroscopes, light sensors, GPS and many others. In addition, features can be extracted using the software capabilities of the phone such as the call duration or the applications usage.

In this work, we focus on the association of the location with call information. Location can be computed from the networks informations, the wifi networks available and the GPS of the mobile phone. Call information can be extracted from the software part of the phone and from hardware part too. For instance, on the Android system it is possible to extract:

- The duration of the call
- The position of the phone (accelerometer, gyroscope)
- The location of the user

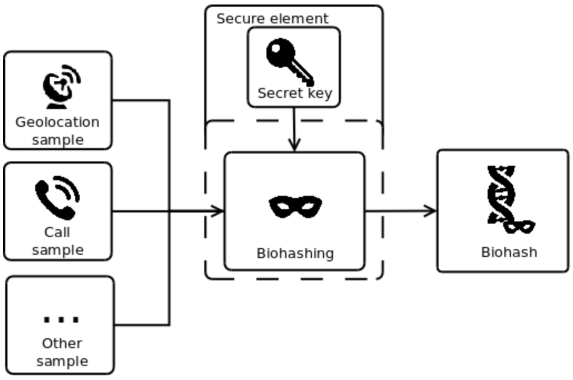

Figure 1: Client architecture.

- The phone number of the callee
- Noise informations from the environment

For this article, we have the following information available in an dataset described in section 4:

- The location of the cell from where the call as been made
- The phone number of the callee

Those data were chosen because they already offer interesting performances in the literature (Li et al., 2013). The generated Biocode size is dependent of the input vector (see section 3.1. As a consequence, the longer the input vector is, the better it is to avoid collision between Biocodes. In this paper we only use geolocation and phone number data because of the available dataset. In a real use case, additional data can be used.

The verification process is done online. Before being sent to the server, the content of the data must first be anonymized. This step is performed using the Biohashing algorithm. The privacy protection depends on the security of the secret key used to perform the Biohashing. To ensure this security, the key is stored inside the secure element of phone. It can either be an online secure element or a physical one.

Depending on the computing capabilities of the secure element, the Biohashing algorithm could be computed directly inside it. This way, the secret key is never revealed. This is described in figure 1

Once the Biohashing is performed, the data can be sent online. To avoid an interception of the biohashed samples by an attacker, the data must be sent through a secure channel. This could be done using an TLS(Dierks, 2015) connection.

3.2.2 Server Architecture

The server receives BioCodes continuously each time an user calls. The first step is to store the BioCode

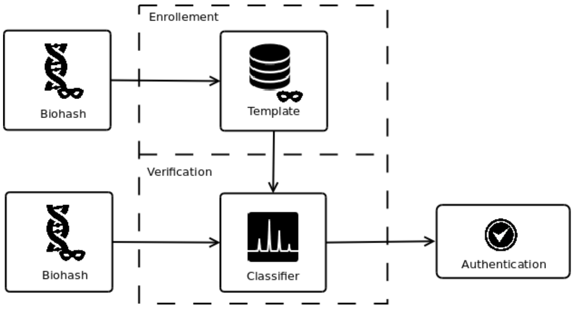

Figure 2: Server architecture.

in the database. This database can be centralized because each user have its one key. The stored data permits to construct a template. When enough data are enrolled for an user, the server switch to verification mode. This step is described in figure 2.

When in verification mode, the server keeps receiving BioCode from the mobile phone. When receiving a BioCode, the server uses a classifier to determine whether the received sample is genuine or not.

In the next section, we describe the experiments we lead to evaluate the proposed approach.

4 EXPERIMENTS

4.1 Dataset

The collected dataset is a private real dataset containing the communication behavior of 100 users during one month extracted from the network information. The available data are:

- The latitude and longitude of cell
- The phone number of the caller
- The phone number of the callee
- The type of communication (Text message or phone call)

We use those properties to perform a static and transparent authentication. The idea is to use the behavioral data extracted from a phone communication to authenticate a user. To perform this authentication, we choose not to use impostors data. This constraint reflects real life industrial datasets. Even if those datasets can contain millions of customers, impostors data are not necessarily inside the dataset.

Each entry of the dataset is composed of a call or a text message associated to the position of the user. Table 1 presents the general statistics of the dataset. We split the one-month dataset into two sets: the first set, corresponding to the 15 first days is dedicated to the creation of the template, while the remaining data are used for test purpose.

Table 1: Size of the communication dataset.

	Enrollment data	Verification data
Maximum	666	666
Minimum	16	14
Mean	157.5	109

In the following section, we describe the experimental protocol used to evaluate the proposed solution both in terms of performances and in terms of privacy protection.

4.2 Protocol

In the proposed approach, we evaluate both the system performance and the privacy protection.

To evaluate the performance of the system we use the False Match Rate and the False Non Match Rate. Two biometric samples of the same person are not exactly the same. As a consequence, an error can occur if a collected sample is too different from the one stored. This error is called the False Non Match Rate (FNMR) (Jain et al., 2004). Oppositely, two persons could have a similar behaviour and as consequence the collected samples could be so close that the recognition system accepts the impostor as the genuine user. This error is called the False Match Rate (FMR) (Jain et al., 2004).

To evaluate the privacy protection of the proposed solution we use a measure based on the Shannon entropy. The theoretical entropy of a perfect BioCode is equal to its length. When computing the real entropy a dataset of BioCode, we observe that the results are inferior to the theoretical entropy. The difference between these two measures is called the privacy leakage and express how much information is available concerning the original data. This value is expressed in bits.

Two classifiers are evaluated. The first classifier used is a Classical Support Vector Machine (Boser et al., 1992) answers to two class classification problems. In the studied case, we model each user's behavior with a one-class SVM. As a consequence, we use a One Class Support Vector Machine. The aim of a One Class SVM is to decide if new samples are within the previously enrolled ones. The implementation used the libSVM(Chang and Lin, 2011) for Matlab. Additionally, we compute the sum of the distance to the distance of the nearest entries to the samples collected in the user template. The approach was preferred over the Kmeans algorithm because of the dataset, because the geolocation are already gathered by cell.

In order to evaluate the performance of the dataset, we first compute the verification without any privacy protection. In this first case, the data are sent in clear. Then, we use the best case scenario where impostors try to perform a zero effort attack. In this attack, they simply try to authenticate with their own BioCodes. Finally, the worst case scenario is envisaged, in this case, the phone is stolen and the attacker try to perform usual phone calls. Those results are exposed in the following section.

4.3 Experimental Results

We evaluate our results both in terms of performance and and of privacy protection for the differents scenarios.

4.3.1 Without Privacy Protection

The first testing scenario is to authenticate without using any privacy protection on the data. This permits to evaluate the basic performance of classical algorithms on our dataset.

We use the caller number as an identifier to verify our result. The only difference in this scenario is that the biohashing algorithm is not applied on the data. We evaluate two classifiers to determine the best approach for the current data. The first one is a One Class SVM with $nu = 0.0001$ and the second one is based on the KNN classifier. Results are summarized in tables 2 and 3.

Table 2: One Class SVM without privacy protection.

FRR (%)	FAR (%)
29.54	1.23

Table 3: KNN without privacy protection.

Number of neighbors	EER (%)	Corresponding threshold
1	8.39	0.15
2	8.39	0.30
3	9.15	0.49
4	9.28	0.67
5	9.77	0.86

Results using the KNN classifier are close to those we can found in the literature (Li et al., 2013). We observe a small difference that can be explained by the fact only one class is used for classification. However, the One Class SVM does not offer sufficient performance to be used as an authentication system. Similar results were observed using the One Class SVM with the privacy protection. About the privacy constraints, none is respected because a potential attacker can access all the point of interest usable for this authentication system. This means all sensitive information are leaked. In the next section, we evaluate how Biohashing improves those results.

4.3.2 Best Case Scenario

The best case scenario represents the normal utilization of the authentication protocol. In this scenario, a zero effort attack is performed. The Biohashing algorithm was applied as follows : the latitude and longitude were wrote in digits using respectively a 8 digits precision and a 6 digits precision, the phone number was concatenated to obtain a 24 digits tuple. This tuple is then converted to a 104 bits binary vector. This vector is then Biohashed to obtain a 100 bits Biocode. Table 4 and table 5 summarize the results obtained using a OCSVM and KNN.

Table 4: One Class SVM in the best case scenario.

FRR (%)	FAR (%)
35.19	0

Table 5: KNN in the best case scenario.

Number of neighbors	EER (%)	Corresponding threshold
1	1.04	0.30
2	1.10	0.62
3	1.09	0.95
4	1.16	1.29
5	1.19	1.63

The privacy protection is evaluated by looking at how much bits of information are leaked when using this protection. The evaluation process measures the difference between a perfect dataset of random vectors and our results. We obtain an entropy of 96 bits. It means a privacy leakage of 4 bits. This represents a great advance in terms of privacy preservation compared to the previously available solutions which leaves the data in clear.

Results obtained with using the privacy protection are greatly improved. With an EER as low as 1% and the privacy protection it can be envisaged to use this solution in an online dataset.

4.3.3 Worst Case Scenario

In the worst case scenario, the attacker knows the key. The only possible solution is that the attacker has stolen the mobile phone because the key is stored inside a secure element. The performances are shown in table 6 and table 7.

Table 6: One Class SVM in the worst case scenario.

FRR (%)	FAR (%)
34.60	2.68

Table 7: KNN in the worst case scenario.

Number of neighbors	EER (%)	Corresponding threshold
1	10.45	0.23
2	10.16	0.47
3	10.65	0.72
4	10.69	0.98
5	10.76	1.24

Even in this scenario, the secret key is not accessible to an attacker. This protects the data contained in the dataset. Assuming an attacker possess both the mobile phone and the dataset, an attacker can only guess location to see if the Biocodes are similar to obtain information about the location of an user. Of course because the attacker possess the mobile phone, he has access to the call history and the called numbers.

5 CONCLUSION AND FUTURE WORKS

The ROC curves for the three evaluated scenarios are exposed in figure 3. We observe a great improvement when using the Biohashing algorithm in the nominal case (best case scenario). A limitation to this study is the number of available samples. But even with this limitation due to the available datasets, it shows great opportunity in privacy preserving mobile authentication. In the performance evaluation we focus on the call and location information. Those information can also be used combined with any other sensors and also

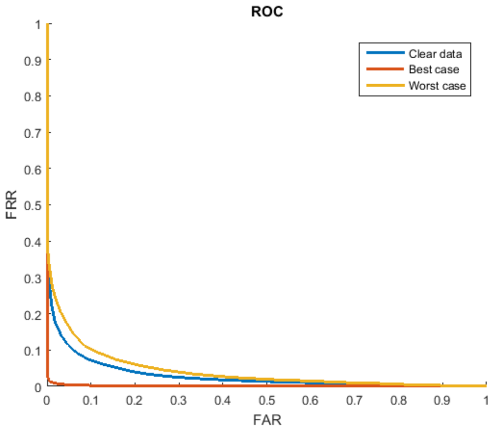

Figure 3: ROC curves in the different scenarios.

Table 8: Comparison with other solutions.

Ref.	No. of Users	Features	Privacy	Performance	Revocable
Li et al.(Li et al., 2013)	71	Location & call	None	EER:8.8% with 1 sample and EER:5.3% with 6 samples	✗
Savaenee et al.(Saevanee et al., 2014)	30	linguistic analysis, keystroke dynamics and behavioral profiling	None	EER:3.3%.	✗
Tanviruzzaman et al.(Tanviruzzaman and Ahamed, 2014)	13	gps location and gait	None	EER:10%	✗
Fridman et al.(Fridman et al., 2015)	200	GPS	None	FAR:11% and FRR:6%	✗
Fridman et al.(Fridman et al., 2015)	200	text message content, gps location, applications, web browsing	None	ERR:5% after 1 minute and EER:1% after 30 minutes	✗
Chow et al.(Chow et al., 2010)	50	phone, browser, sms, gps	Data are stored only on the phone	Legitimate user can perform around 90 actions before being rejected by the system while an impostor can only performed 10 actions before being locked out	✗
Safa et al.(Safa et al., 2014)	Not provided	calls, location, Wi-Fi access, visited websites	3-round protocol between the device and carrier	Not provided	✓
Proposed framework	100	calls, location	Biohashing : 4 bits privacy leakage	EER:1.04% best case scenario, EER:10.45% worst case scenario	✓

using the time (Kayacik et al., 2014). In this case, a similar approach to both allow revocation and privacy protection can be applied.

In a future work, we wish to combine this modality with others to add new modality to a Single Sign On environement.

The proposed approach is compared to the literature in the table 8. In this table both the different privacy protection mechanisms and the performances of the system are compared.

REFERENCES

Belguechi, R., Cherrier, E., Rosenberger, C., and Ait-Aoudia, S. (2013a). An integrated framework combining bio-hashed minutiae template and PKCS15 compliant card for a better secure management of fingerprint cancelable templates. *Computers & Security*, 39:325–339.

Belguechi, R., Cherrier, E., Rosenberger, C., and Ait-Aoudia, S. (2013b). Operational bio-hash to preserve privacy of fingerprint minutiae templates. *IET Biometrics*, 2(2):76–84.

Bolle, R., Connell, J., and Ratha, N. (2002). Biometric perils and patches. *Pattern Recognition*, 35(12):2727–2738.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27.

Chow, R., Jakobsson, M., Masuoka, R., Molina, J., Niu, Y., and Song, Z. (2010). Authentication in the clouds: A framework and its application to mobile users. In *Pro-*

ceedings of the 2010 ACM workshop on Cloud computing security workshop.

Clarke, N. (2011). *Transparent User Authentication Biometrics, RFID and Behavioural Profiling*. Springer.

Das, S., Hayashi, E., and Hong, J. l. (2013). Exploring capturable everyday memory for autobiographical authentication. In *Proceedings of the 2013 ACM international joint conference on UbiComp '13*.

Derawi, M. and Bours, P. (2013). Gait and activity recognition using commercial phones. *Computers & Security*.

Dierks, T. (2015). The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246.

Fridman, L., Weber, S., Greenstadt, R., and Kam, M. (2015). Active authentication on mobile devices via stylometry, application usage, web browsing, and gps location. *arXiv preprint arXiv:1503.08479*.

Goh, A. and Ngo, D. (2003). Computation of cryptographic keys from face biometrics. In *Communications and Multimedia Security*, pages 1–13. LNCS 2828.

Google (2016). Google Abacus project. http://www.androidcentral.com/project-abacus-atap-project-aimed-killing-password. [Online; accessed 10-July-2016].

Grosse, E. and Upadhyay, M. (2013). Authentication at scale. *Security & Privacy, IEEE*, 11(1):15–22.

Guidorizzi, R. P. (2013). Security: Active authentication. *IT Professional*, 15(4):4–7.

Hayashi, E., Das, S., Amini, S., Hong, J., and Oakley, I. (2013). Casa: Context-aware scalable authentication. In *SOUPS '13 Proceedings of the Ninth Symposium on Usable Privacy and Security*.

Jain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(1):4–20.

Jakobsson, M., Shi, E., Golle, P., and Chow, R. (2009). Implicit authentication for mobile devices. In *HotSec'09 Proceedings of the 4th USENIX conference on Hot topics in security*.

Kayacik, H. G., Just, M., Baillie, L., Aspinall, D., and Micallef, N. (2014). Data driven authentication: On the effectiveness of user behaviour modelling with mobile device sensors. *CoRR*, abs/1410.7743.

Li, F., Clarke, N., Papadaki, M., and Dowland, P. (2013). Active authentication for mobile devices utilising behaviour profiling. *International Journal of Information Security*.

Mondal, S. and Bours, P. (2013). Continuous authentication using mouse dynamics. In *Biometrics Special Interest Group (BIOSIG), 2013 International Conference of the*, pages 1–12. IEEE.

Nauman, M., Ali, T., and Rauf, A. (2013). Using trusted computing for privacy preserving keystroke-based authentication in smartphones. *Telecommunication Systems*, 52(4):2149–2161.

Patel, V. M., Chellappa, R., Chandra, D., and Barbello, B. (2016). Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61.

Patel, V. M., Ratha, N. K., and Chellappa, R. (2015). Cancelable biometrics: A review. *IEEE Signal Processing Magazine*, 32(5):54–65.

Ratha, N., Connell, J., and Bolle, R. (2001). Enhancing security and privacy in biometrics-based authentication system. *IBM Systems J.*, 37(11):2245–2255.

Rathgeb, C. and Uhl, A. (2011). A survey on biometric cryptosystems and cancelable biometrics. *EURASIP J. on Information Security*, 3.

Saevanee, H., Clarke, N., Furnell, S., and Biscione, V. (2014). Text-based active authentication for mobile devices. In *ICT Systems Security and Privacy Protection*, pages 99–112. Springer.

Safa, N. A., Safavi-Naini, R., and Shahandashti, S. F. (2014). Privacy-preserving implicit authentication. In *IFIP International Information Security Conference*, pages 471–484. Springer.

Sophos. Mobile usage. https://www.sophos.com/en-us/press-office/press-releases/2013/03/mobile-security-survey.aspx. [Online; accessed 10-July-2016].

Tanviruzzaman, M. and Ahamed, S. I. (2014). Your phone knows you: Almost transparent authentication for smartphones. In *Computer Software and Applications Conference (COMPSAC), 2014 IEEE 38th Annual*, pages 374–383. IEEE.

Teoh, A., Ngo, D., and Goh, A. (2004). Biohashing: two factor authentication featuring fingerprint data and tokenised random number. *Pattern recognition*, 40.

Vance, A. (2010). If your password is 123456, just make it hackme.

Witte, H., Rathgeb, C., and Busch, C. (2013). Context-aware mobile biometric authentication based on support vector machines. In *Emerging Security Technologies (EST), 2013 Fourth International Conference on*, pages 29–32. IEEE.