Electronic Communications of the EASST
Volume 080 (2021)

Conference on Networked Systems 2021
(NetSys 2021)

Privacy-Preserving and Scalable Authentication based on Network Connection Traces

David Monschein, advised by Oliver P. Waldhorst

5 pages

Guest Editors: Andreas Blenk, Mathias Fischer, Stefan Fischer, Horst Hellbrueck, Oliver Hohlfeld, Andreas Kassler, Koojana Kuladinithi, Winfried Lamersdorf, Olaf Landsiedel, Andreas Timm-Giel, Alexey Vinel
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Privacy-Preserving and Scalable Authentication based on Network Connection Traces

David Monschein[1], **advised by Oliver P. Waldhorst**[1]

[1] Data-centric Software Systems (DSS) Research Group at the Institute of Applied Research
Karlsruhe University of Applied Sciences, Karlsruhe, Germany

Abstract: Since password-based authentication is no longer sufficient for web applications, additional authentication factors are required. Especially in the context of mobile devices and with regard to usability, there is an increasing focus on methods where the user's behavior is used as authentication factor (e.g., touchscreen interactions or sensors). As this typically requires the processing of large amounts of sensitive data, issues related to privacy and scalability arise. Our work addresses the issues by presenting a scalable and privacy-friendly approach for authenticating users of mobile applications based on information about their network connections.

Keywords: Mobile applications, Mobility patterns, Machine learning

1 Research Problem and Objectives

Web-based applications are increasingly being accessed from mobile devices, which offer a broad range of attack possibilities [WŁ20]. Thus, sophisticated security measures such as a comprehensive authentication of users are indispensable to prevent the misuse of digital identities. The sole use of password-based authentication is no longer sufficient, as passwords can fall into the hands of third parties due to a multitude of potential attacks [BLBA12]. One common way to mitigate this problem is to make use of multi-factor authentication (MFA), which requires the user to provide several independent proofs (factors) to confirm his identity [OBM$^+$18]. Examples of additional factors are codes sent via short message service (SMS) or the use of key generators. However, many of these factors are not suitable with regard to mobile applications. On the one hand, confirmations via SMS are pointless if an attacker has physical access to the device (e.g., in the case of theft), because he has also access to the received text messages. The same applies to key generators that are installed on the same device. On the other hand, an external key generator that users must carry with them poses a substantial threat to the usability.

For these reasons, more and more approaches concentrate on factors which are specifically tailored to mobile devices and avoid additional effort for the user. In particular, the analysis of location data [TIK$^+$20], sensor data [AMV$^+$19] and touchscreen interactions [KPC$^+$20] is considered. However, since these data sources reveal very sensitive user information, their processing is associated with significant privacy concerns. Furthermore, establishing such analytics for a large number of users poses challenges in terms of resource consumption and scalability. So far, these issues have not been sufficiently addressed by existing works.

Therefore, we propose a privacy-preserving and scalable methodology for establishing an authentication factor based on analyzing traces of network attachment points of mobile devices.

In order to preserve the privacy of the users, the sensitive data is pre-processed directly on the user's device in such a way that minimal information about the actual network connection and the corresponding location can be derived from it. Subsequently, it is examined whether the pre-processed data is consistent with the behavior previously observed for the user. We argue that this can be achieved by training a single machine learning model which is applied for all users. As a result, the scalability is significantly improved compared to related work.

2 Conception

Our approach enables the establishment of an additional authentication factor for conventional mobile applications that are based on the client-server principle. The basic idea is that the user's mobile device (frontend) provides information about its network connection to the application server (backend). In order to avoid that the server can extract sensitive details, this information is encoded directly on the user's device. After the data has been transferred to the backend, traces of transitions between different network attachment points are formed and compared with previously observed behavior of the user. The quantification of the deviations can then be used to make an authentication decision. Figure 1 visualizes the main components of our approach.

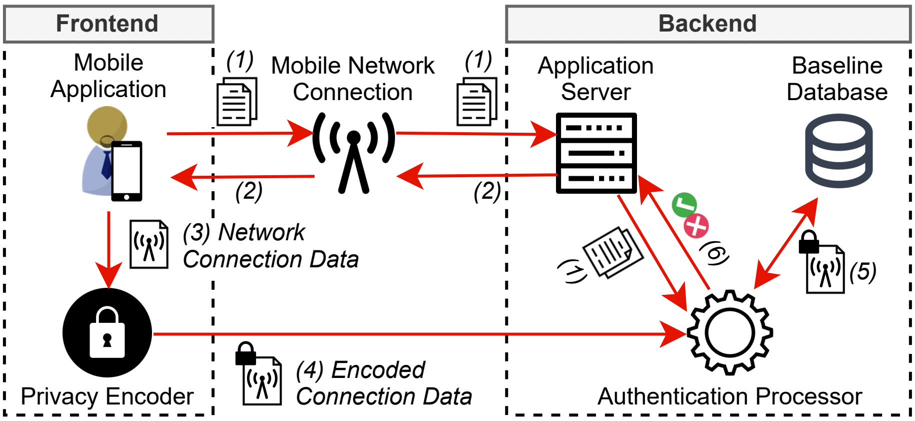

Figure 1: Overview of the main components and data flows of our authentication strategy

Initially, the user's device monitors its network connection and tracks transitions to other network attachment points. In particular, the type of network, the ID of the network attachment point (e.g., base station ID) and the amount of transmitted data are considered. As soon as the user initiates a request (1), the authentication procedure is started by the *Application Server*. The *Application Server* requests information about the user's latest network connections (2). Before the frontend responds with the requested data, it is pre-processed by the *Privacy Encoder* (3).

The *Privacy Encoder* is responsible for masking sensitive information. In our case, this mainly concerns the ID of the network attachment point, from which the user's location can typically be determined with considerable accuracy. In order to prevent this, the client device selects a random key once, which is stored permanently and not revealed at any point in time. Subsequently, this key is used by the Privacy Encoder to encrypt data. For data without an underlying order, e.g., structureless identifiers, we use Keyed-Hashing [KBC97] and for ordered data, e.g., transmitted data, we apply Order-Preserving Encryption (OPE) [BCLO09]. Keyed-Hashing ensures a higher level of security, as it is only susceptible to bruteforce attacks [KBC97]. OPE, on the other hand, does not provide such strong security guarantees, but assuming that the attacker does not know the key and cannot map arbitrary encrypted (decrypted) values to the corresponding decrypted (encrypted) ones, the recovery of the raw data is unlikely [BCLO09]. As in our case the backend neither has access to the key nor has access to the frontend on which the encryption takes place, both methods can be considered as very reliable in terms of privacy protection.

After the data has been processed by the *Privacy Encoder*, the transformed network connection details are sent to the *Authentication Processor* (4). First, the *Authentication Processor* retrieves previous data points of the same user from the *Baseline Database*, referred to as *history frame* (5). The *history frame* is used as reference to evaluate whether the received data conforms to the expected user behavior. If no information about this user is known, our system cannot make an authentication decision yet. Consequently, conventional procedures must be conducted as fallback. In contrast, if reference data is available, the *history frame* and the currently received information, also called *observation frame*, are used as input for a machine learning model.

An attacker who does not know the client's key is not able to generate a valid *observation frame* and even if the key was obtained (e.g., by stealing the device), the attacker would also need to know the user's behavior to generate a valid *observation frame*, because we ensure that collected records about the network connections on the user's device are periodically deleted. Note that if the user's device changes, the respective key changes as well and the data already collected can no longer be used as a reference for the affected user. Nevertheless, as we expect that a user rarely changes his device, this is negligible. In our implementation, we applied gradient boosted decision trees [CG16] to decide whether the two frames are consistent. A fundamental aspect of our approach is that we train only a single model, which ensures scalability for high numbers of users. The training procedure is described in the course of our experimental analysis in Section 3. Based on the output of the model, an authentication decision can be made (6) and if approved, the contents of the observation frame are persisted in the Baseline Database.

3 Preliminary Results

By means of an existing dataset, we investigated how well our approach performs in a real-world setting. It contains the network requests issued by 1000 different users when interacting with mobile applications [YLX+18]. All users were located in the same city during the observation period of one week. After removing users with an unusually high or low number of interactions, 783 users remained. We performed the training based on the data of 607 users and the remaining 176 were used for evaluation. From the network requests, we inferred transitions between base stations and assembled them to fine-grained network connection traces for each user.

Subsequently, we used the combination of an observation frame and a history frame as input for our authentication model as described in Section 2. The size of the obervation frame was set to 5 and the size of the history frame was set to 25. In other words, the model should determine if the 5 passed network transitions (observation) originate from the same user as the given 25 ones (history), with each transition consisting of the associated base station IDs and the amount of data transmitted until the transition happened. The parameters reflect realistic values for deployment in practice, considering the dataset used. Because we need two classes of training samples (valid and invalid), we first built valid pairs of observation frames and history frames (i.e., data within both frames comes from the same user) and then artificially added invalid pairs by recombining the history frames with observation frames from other users. By using this method, the authentication system can learn to distinguish between the behavior of different users. Furthermore, we trained three different models, one that works on raw data, one that works on data that has been encrypted using OPE, and finally a model that works on data that has been transformed using Keyed-Hashing. As usual for the evaluation of authentication methods, we used the Equal Error Rate (EER) to assess the quality of the models, which is defined as the common value where the False Acceptance Rate equals the False Rejection Rate [AMVF20].

Figure 2 shows the EERs for the individual users in the test set when applying different privacy protection methods. It can be seen that OPE (average EER 17.04%) does not cause a significant performance drop compared to the evaluation on raw data (average EER 15.78%), but Keyed-Hashing does (average EER 27.82%). It can be concluded that the base station IDs have an underlying order and because this structure is suppressed due to Keyed-Hashing, the accuracy is reduced. It can be noted that the average EERs are low enough that the models can serve as an authentication factor [AMVF20]. In addition, we investigate how this approach can be combined with other factors to establish a standalone continuous authentication setting [MW21], without relying on conventional strategies such as SMS confirmation codes as fallback.

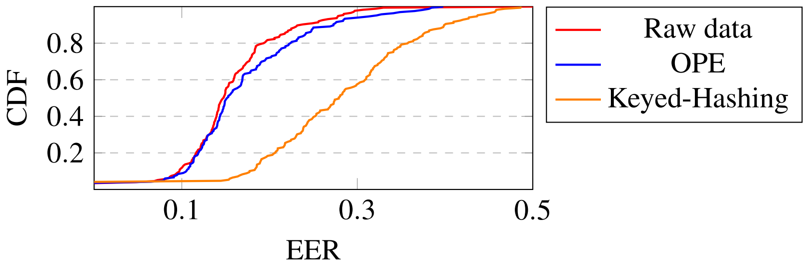

Figure 2: Cumulative distribution function (CDF) of the EERs of the individual users

4 Conclusion and Outlook

We presented a strategy to identify users based on information about their network connections. Our approach combines a profound privacy protection with a special focus on scalability, which enables the establishment of a user-friendly additional factor for existing authentication systems. Using an experimental analysis, we have shown that our methodology works as intended and evaluated the effect of the privacy protection measures on the accuracy of the authentication. The ultimate goal is to embed this strategy, along with other previous works [MW21], into a comprehensive framework to establish robust protection against various attack scenarios.

Acknowledgements: Funded by the German Federal Ministry of Education and Research(BMBF), RefNr. 16KIS1142K

Bibliography

[AMV+19] A. Acien, A. Morales, R. Vera-Rodriguez, J. Fierrez, R. Tolosana. MultiLock: Mobile Active Authentication Based on Multiple Biometric and Behavioral Patterns. In *Proc. 1st Int. Workshop on Multimodal Understanding and Learning for Embodied Applications (MULEA)*. P. 53–59. Nice, France, 2019.

[AMVF20] A. Acien, A. Morales, R. Vera-Rodriguez, J. Fierrez. Smartphone Sensors for Modeling Human-Computer Interaction: General Outlook and Research Datasets for User Authentication. In *Proc. 44th IEEE Annual Computers, Software, and Applications Conf. (COMPSAC)*. Pp. 1273–1278. Madrid, Spain, 2020.

[BCLO09] A. Boldyreva, N. Chenette, Y. Lee, A. O'Neill. Order-Preserving Symmetric Encryption. In *Proc. 28th Annual Int. Conf. on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*. Pp. 224–241. Cologne, Germany, 2009.

[BLBA12] Y. Bang, D.-J. Lee, Y.-S. Bae, J.-H. Ahn. Improving information security management: An analysis of ID–password usage and a new login vulnerability measure. *Int. Journal of Information Management* 32(5):409–418, 2012.

[CG16] T. Chen, C. Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. P. 785–794. San Francisco, California, USA, 2016.

[KBC97] H. Krawczyk, M. Bellare, R. Canetti. RFC2104: HMAC: Keyed-Hashing for Message Authentication. USA, 1997.

[KPC+20] T. Karanikiotis, M. D. Papamichail, K. C. Chatzidimitriou, N.-C. I. Oikonomou, A. L. Symeonidis, S. K. Saripalle. Continuous Implicit Authentication through Touch Traces Modelling. In *20th IEEE Int. Conf. on Software Quality, Reliability and Security (QRS)*. Pp. 111–120. 2020.

[MW21] D. Monschein, O. P. Waldhorst. SPCAuth: Scalable and Privacy-Preserving Continuous Authentication for Web Applications. In *2021 IEEE 46th Conference on Local Computer Networks (LCN 2021)*. virtual, Oct. 2021.

[OBM+18] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, Y. Koucheryavy. Multi-Factor Authentication: A Survey. *Cryptography* 2(1), 2018.

[TIK+20] T. P. Thao, M. Irvan, R. Kobayashi, R. S. Yamaguchi, T. Nakata. Self-enhancing GPS-Based Authentication Using Corresponding Address. In *Proc. IFIP Data and Applications Security and Privacy XXXIV*. Pp. 333–344. 2020.

[WŁ20] P. Weichbroth, Ł. Łysik. Mobile Security: Threats and Best Practices. *Mobile Information Systems* 2020:8828078, Dec 2020.

[YLX+18] D. Yu, Y. Li, F. Xu, P. Zhang, V. Kostakos. Smartphone app usage prediction using points of interest. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1(4):174, 2018.