Digital Object Identifier 10.1109/ACCESS.2017.DOI

Toward Usable and Accessible Two-Factor Authentication Based on the Piezo-Gyro Channel

YOSSI OREN[1], (Senior Member, IEEE), DAN ARAD[2]
[1]Ben Gurion University of the Negev, Israel (e-mail: yos@bgu.ac.il)
[2]Ben Gurion University of the Negev, Israel (e-mail: arda@post.bgu.ac.il)
Corresponding author: Yossi Oren (e-mail: yos@bgu.ac.il).

ABSTRACT

Two-factor authentication (2FA) is crucial for protecting the security of users authenticating to online servers. Despite its importance, users hesitate to use 2FA, due to usability issues.
In this report we present a prototype implementation of PiGy, a novel system which improves the usability of existing methods, without compromising on security and compatibility.
In PiGy, a one time password is automatically passed from the external token to a smartphone by selectively applying an acoustic stimulus to the phone's microelectromechanical (MEMS) gyroscope, using a piezoelectric transducer. This scheme is much easier to use, requires no additional hardware support on modern phones, and is fully compliant with the time-based one time password (TOTP) standard.
We implement a proof of concept of PiGy, and perform both a functional test and a user study to evaluate it. Through our evaluation we show that this authentication scheme is a viable alternative to existing methods, and that users agree with its usability advantages.

I. INTRODUCTION

FOR many years, providing good authentication solutions has proved to be a difficult challenge. The simple scheme of username and password was shown to be insecure [4], and two-factor authentication has become a norm in many services.

A password is only *"something you know"*. The idea behind two-factor authentication is to strengthen authentication by adding another component, reflecting *"something you have"* or *"something you are."* Simple examples are a cellphone as *"something you have"* (i.e., in SMS one-time password (OTP) based authentication), or your fingerprint as *"something you are."*

A subcategory of two-factor authentication is based on hardware tokens, for example SecurID by RSA. Many of these tokens use the time-based one-time password (TOTP) algorithm [8]. Under this implementation, the token held by the user is responsible for generating OTPs, and displaying them to the user on a screen. In order to generate the correct OTPs, the token only needs to have the user's key, and a clock synchronized with the authentication server.

Each two-factor authentication solution presents its own strengths and weaknesses. SMS based OTPs, for example, do not require the user to carry anything other than their mobile device, but depend on cellular network availability, incur additional costs, and are susceptible to attacks [7]. In contrast, SecurID does not depend on cellular networks, but requires tokens to be deployed to each user. Common to both of these schemes is the requirement to manually copy the OTPs into the login screen, an activity with many usability problems, as we describe below.

The abundance of two-factor authentication solutions relates to the difficulty of balancing these security, deployability and usability constraints. This complexity is the reason that secure and usable authentication remains an open problem.

A. THE PIEZO-GYRO CHANNEL

The gyroscope is a Microelectromechanical (MEMS) sensor containing a mass vibrating at one axis. When angular momentum is applied to the mass, the Coriolis effect causes the mass to vibrate in a direction orthogonal to the direction of rotation, with an amplitude with direct correlation to the angular velocity. The angular velocity can be thus measured by measuring the change in the sensor's capacitance.

Smartphone gyroscopes have been shown to be susceptible to acoustic signals in the resonant frequency of their internal mass [10]. These signals induced a motion of the sensor's

inner mass, translating to artificial angular velocity readings. This phenomenon was originally used as a form of attack on gyroscopic sensors, used to disrupt the operation of drones, remote-operated vehicles, smartphones and similar sensor-equipped devices. Farshteindiker et al. [3], showed this phenomenon can be used for non-attack purposes as well, using it to build a unidirectional communication channel, by using a piezoelectric transducer to affect the gyroscope. This *piezo-gyro channel* was shown to work on multiple phone models with acceptable error rates.

This channel can be an interesting and innovative way to facilitate a two-factor authentication solution.

B. ALTERNATIVE CHANNELS

Many suggested authentication solutions rely on smartphone sensors, such as the camera and microphone, and communication channels can also be established via NFC, light sensors or the accelerometer, to name a few.

The camera, microphone and NFC sensors all require extra app permissions, which the user might be reluctant to supply. In contrast, the gyroscope does not require permission to access, enabling a simpler workflow for the user, and maximizing the chance that users will use this additional authentication option. In contrast to accelerometer-based signals, the gyroscope channel, requires very low power to operate. This allows the token to be smaller and more power-efficient.

We also wanted the product to be as secure as possible, and be resistant to shoulder surfing attacks. While sensors such as the light sensor can be eavesdropped on from a distance, the piezo-gyro channel requires physical contact with the device, preventing such attacks.

C. CONTRIBUTIONS

In this article we present PiGy, a two-factor authentication scheme that transmits the OTP to the device using the piezo-gyro channel. As can be seen in **Figure 1**, this scheme only requires that the token is in proximity to the user's cellphone; there is no need for the user to type anything but his/her password, and there is no chance for anyone else to copy the code.

We compare PiGy to existing research in **Table 1**. We have chosen schemes that focus on authentication to external services using sensors available on mobile devices.

As the Table shows, most solutions require dedicated apps and extra permissions, as they use either the microphone or camera. PiGy can be used from the browser without any permissions, since it uses the gyroscope, which is not considered a sensitive component. Our solution also has authentication times and acceptance rates comparable to proposed schemes. Most importantly, it is based on the TOTP standard, a widely accepted secure method of authentication, while other schemes each rely on custom algorithms.

In summary, this research makes the following contributions:

- We propose an innovative technological scheme for two-factor authentication, based on the piezo-gyro channel, for OTP transmission. The piezo-gyro channel lets the scheme offer usability advantages, while requiring no extra security-sensitive permissions.
- We provide a proof of concept implementation for our design, and perform functional evaluation to assess its security and performance. We demonstrate the full authentication process, and achieve low insult and fraud rates, with a reasonable authentication time.
- We investigate how users perceive the scheme through a user study. We observe that our implementation is intuitive, easy to use, and is preferred by many users over existing solutions.

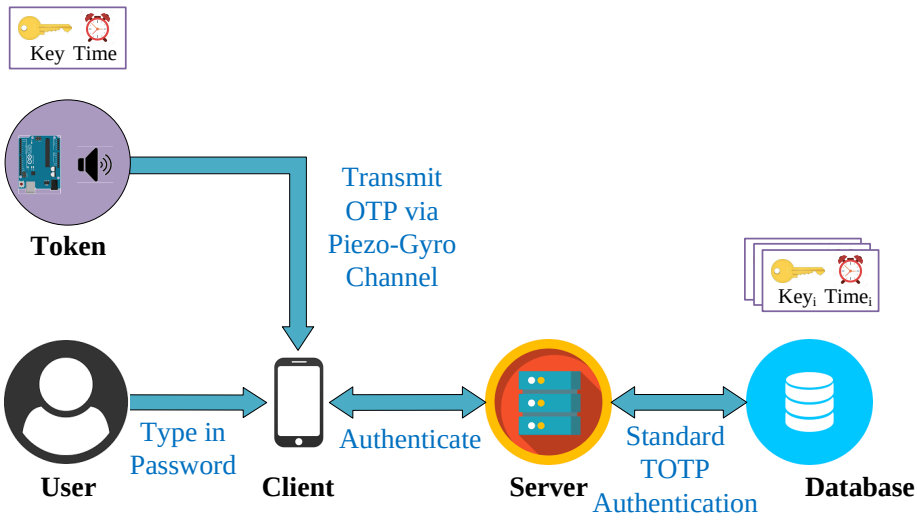


Figure 1. The Piezo-Gyro authentication scheme.

Table 1. Comparison with Related Studies on Smartphone Sensor-Based Authentication.

Study	Requires Dedicated App	Sensors Used	Requires Permissions	Authentication Time	User Study Size	Compatible with RFC 6238	TAR	FAR
Karapanos et al.(2015) [6]	Yes	Microphone	Yes	3 Seconds	32	No	99.8%	-
Schneegass et al.(2016) [9]	Yes	Microphone	N/A	1 Second	10	No	-	10.2%
Azimpourkivi et al. (2017) [1]	Yes	Camera	Yes	0.5 Seconds	42	No	95.75%	0.02%
Zhang et al. (2018) [12]	Yes	Microphone	Yes	-	N/A	No	∼99%	-
Zhou et al. (2018) [13]	Yes	Microphone & Camera	Yes	< 0.5 Seconds	45	No	-	-
PIGY (Our Contribution)	No	Gyroscope	No	5 Seconds	34	Yes	∼90%	∼2%

II. OUR SCHEME

In this Section, we list the requirements and processes that make up our proposed scheme. We rely on those later in our implementation and evaluation.

A. REQUIREMENTS

Following the work by Bonneau et al. [2], we list our requirements under the categories of *Security*, *Deployability*, and *Usability*.

In the **security** category, our OTP generation must comply with the TOTP standard. This standard was proven secure, and is common in other existing solutions. Our token should also be resilient to attacks where an adversary has physical access. It should minimize the attack surface, by disabling inputs on any existing hardware interface. It should also have a short transmission range, in order to prevent other devices from receiving the OTP sequence. Our scheme is not required to be secure against attackers running on the device, or those with access to the supply chain of the tokens.

In the **deployability** category, the scheme must allow for a single service provider to authenticate many users to many services, given each user's identity (secret key) is shared between the services. Service providers may even share tokens, if they manage user's identities through a third party. The scheme should also require as few changes as possible on the server. The token should be cheap to manufacture, and easy to attribute to a user.

Finally, in the **usability** category, it should be easy to use the token for authentication, particularly for motion and vision impaired users. More specifically, authentication time should be short, false positives should be rare, and the token should have a long battery life.

B. PROCESSES

The authentication scheme involves two separate processes: enrollment and authentication.

The **enrollment** process is performed only once in the token's lifetime, when the token is associated with a user. The goal of enrollment is to give the token the information necessary to generate correct OTPs for the user. This is done by transferring the user's key, as well as initializing the token's internal clock, through an input interface that is only available during enrollment. When enrollment concludes, the token should disable this interface, ignoring any additional inputs, to minimize its attack surface.

The goal of the **authentication** process is to actually authenticate to the service provider. It is initiated by the user every time they want to log in. To authenticate, the user opens the provider's login screen, enters a username and password, and places the token next to their device. The token then generates and transmits OTPs generated using the standard TOTP algorithm.

The server retrieves the username's password and key, reconstructs the received OTP, and generates an OTP sequence. Then, it compares the received password to its own stored copy, and the received OTP to the one it generated locally.

We note that the piezo-gyro authentication scheme intentionally has many similarities to existing token-based schemes. This is because it is built on the same principles as these schemes, and mainly aims to improve on usability.

The uniqueness of the scheme is derived from the OTP transmission mechanism, which uses the piezo-gyro channel. This changes both the way the user interacts with the token, and the way the OTP is received and is later reconstructed. In particular, since the piezo-gyro channel is prone to corruption by environmental noise, the sequence received by the client might not be exactly identical to the correct one expected by the server. To balance between the security and usability needs of specific deployments, our scheme can optionally accept OTPs that are not identical to the expected sequences, but rather similar enough according to some similarity function, as explained below.

III. TECHNICAL IMPLEMENTATION

A. COMPONENTS

Our prototype token was implemented using an Arduino Nano board. To generate the piezo-gyro signal, we use a PUI audio APS2509S-T-R piezoelectric transducer, as suggested by Farshteindiker et al. [3]. Our implementation of the transmission is made using the built-in pulse width modulation (PWM) mechanisms of the Arduino for modulation. To conserve battery, the token shuts down unused peripherals and transmits only when a button connected to the microcontroller is pressed.

B. TOKEN PROXIMITY

Our setup transmits to a very short distance, requiring the token to physically touch the phone. When moving the token more than a few millimeters from the device, the signal-to-noise ratio plummets and reliable communication cannot

be established. This is the optimal setting for a two-factor solution, as it prevents eavesdropping by nearby devices.

C. OTP RECONSTRUCTION

Our system uses 20-bit OTP sequences, for a security level equivalent to the 6-digit numeric code commonly used by other tokens.

The server plays a dual role, serving as both a token enrollment station and an authentication gateway to the service. It is implemented in Python using the Flask framework, and is connected to a MySQL DB, to store user data and other configuration.

The reconstruction process can be roughly split into two parts. In the first part, the gyroscope sample is converted to a bit sequence; in the second part, the OTP is decoded from this sequence.

The input to the conversion process is a gyroscope sample that contains data of the three independent rotation axes. We start by reducing it to one dimension, by computing the root-mean-square of the three axes. We then trim the start and end of the signal, to avoid high amounts of noise that happen when the user presses the authentication button. Next, we shift the signal's phase to be 0. We do so by averaging the phase of the signal relative to the sampling start point. Finally, we split the signal into windows the size of a single bit ($1/bps$ seconds). For each window, we count how many points cross a given threshold, and decide if it should be a logical 1 or a logical 0.

In order to convert the bit sequence to an OTP, we add a unique preamble to the transmission, and pad our sequence to make sure it's unique. Once we have the bit sequence we simply scan until we find the preamble, and then remove the padding bits to reconstruct the original OTP. We've chosen the length of the preamble to minimize transmission overhead, and achieved a sequence length of 31 bits for each OTP.

D. SIMILARITY FUNCTION

In our implementation, we use the absolute value of the Pearson correlation test as a similarity function:

$$|\rho| = \left| \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \right|$$

This value is between zero and one, with zero indicating no correlation, and one indicating a perfect match with equal or opposite signs. The threshold value can be fine-tuned, to provide the best trade-off of usability and security: Setting a low threshold will make it easier for users to authenticate, but will also increase the chance that a user will be able to authenticate with an incorrect token. Standard OTP-based schemes require strict equality, and thus are equivalent to the special case of choosing a threshold of 1.

Another possible approach, which we did not evaluate, would be to use error-correcting codes (ECCs) to tolerate errors in the digital domain. This effectively reduces the security level of the scheme, by accepting additional bit sequences as valid tokens, and incurs a time penalty, as more bits need to be transmitted per OTP.

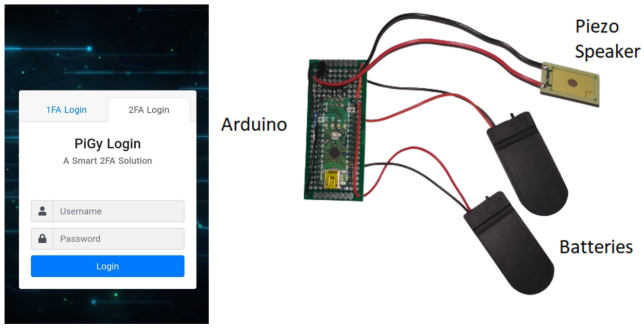


Figure 2. The User Interface (left) and Token (right)

E. USER INTERFACE

As shown in Figure 2, we designed the user interface of our native client to be similar to a standard login form.

The user fills in his or her credentials, and then attaches the token to the back of the phone. Once the login button is pressed, the client samples the gyroscope, and sends the credentials and samples to the server.

We sample for twice the length of an OTP to make sure we have at least one full OTP contained in the sequence. This requires around 5 seconds at a data rate of 12 bps, satisfying our usability constraint.

IV. EVALUATION

We use two separate steps in our evaluation. First, we carry out a system test to extract quantitative parameters relating to the security and performance of our implementation. Next, we perform a user study to gauge its potential as a viable two-factor authentication alternative.

A. SYSTEM TESTING

In this part of the evaluation, we wanted to see how our similarity function threshold affects the insult and fraud rates under various conditions. The insult rate specifies how often will a benign user fail authentication, while the fraud rate specifies how often will a malicious actor succeed. Specifically, we wanted to see how different transmission bit rates, phone models, and activity profiles affect these values.

1) Experiment Design

The experiment is built of different sets, each testing a different combination of parameters. Each set consists of 50 benign and 50 malicious authentication attempts. We simulate a benign authentication by having a registered user use his own token, and a malicious authentication by using the same token for a different user.

To test different conditions and data rates we used a Xiaomi Redmi 4X. We tested it while being still on a table,

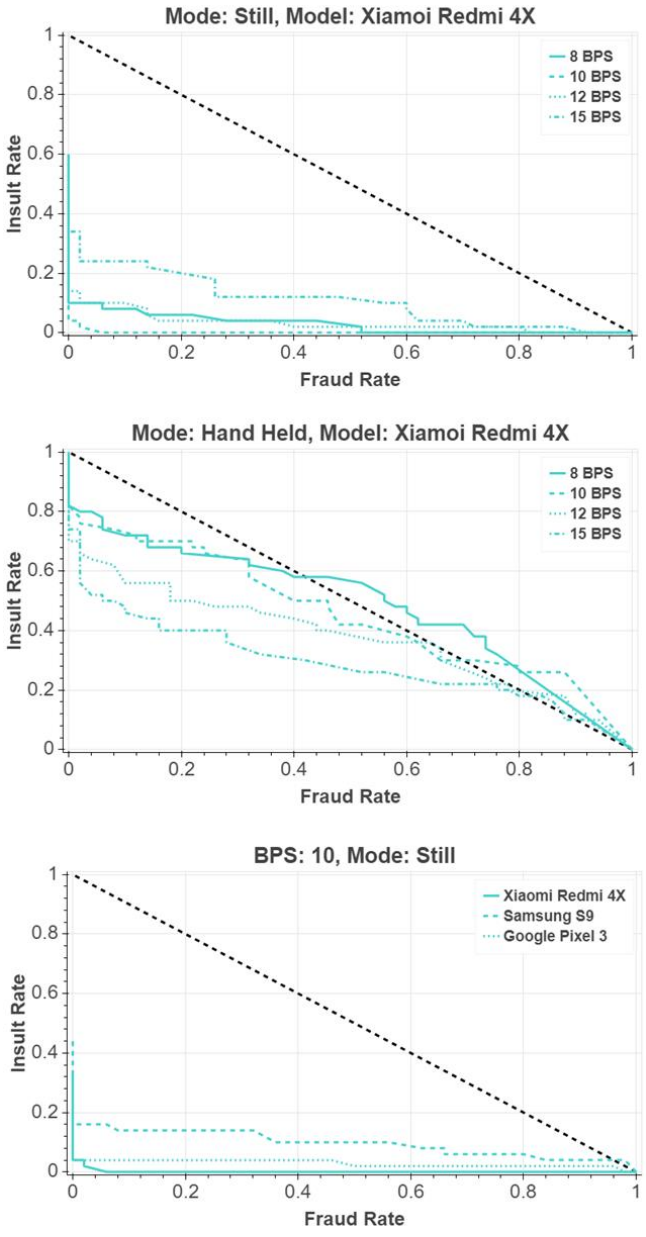


Figure 3. Experiment results shown as receiver operating characteristics (ROC) curves. Top: different transmission rates when device is still. Middle: different transmission rates when device is hand-held. Bottom: different phone models.

and hand held, each with bit rates of 8, 10, 12 and 15 bps. We also evaluated the Samsung Galaxy S9, and Google Pixel 3 while still and with a bitrate of 10 bps, to compare between phone models.

In all the tests, the token was physically attached to the device.

2) Experiment Results

Looking at **Figure 3**, we can see that while the transmission rate has some impact over the results, the greatest impact is caused by device motion.

When still, we achieve low insult and fraud rates, and our data suggests we can pick a threshold value between 0.5 and 0.9 to optimize for usability or security based use-cases, respectively: Setting a low threshold will make it easier for users to authenticate, but will also increase the chance that a malicious actor will be able to authenticate by providing a wider range of accepted OTPs. Using a transmission rate of 12 bps will provide an authentication time of 5 seconds, comparable to existing solutions.

When in motion, the curves are almost indistinguishable, and we can't find a threshold that will provide a good enough trade off between insult and fraud rates. We note this as a major issue that requires additional research.

Testing on different phone models shows that the method can work for multiple devices. We predict that it will work for any device with a MEMS gyroscope, based on previous research and our results.

B. USER STUDY

In this user study we gather information from potential users in order to evaluate our solution, based on criteria from the work by Bonneau et al. [2].

Out of the eight usability criteria, we focus on the *Nothing-to-Carry* (U3), *Physically-Effortless* (U4), *Easy-to-Learn* (U5), and *Efficient-to-Use* (U6) criteria, which are the most related to our scheme's characteristics.

1) Ethical Considerations

This study asks participants to share their personal opinions, as well as information regarding their health. To address potential ethical issues, we made sure to collect data anonymously, and minimize our inquiries regarding the participants themselves as much as possible.

Each participant was given a consent form, which clarified how the data will be stored, who will have access to it, and that participants are not compensated in any way. Signing the consent form was mandatory for participation in the research.

Prior to conducting the experiment, we sought and received approval from the Ben-Gurion University's Institutional Review Board (IRB).

2) Protocol

The study starts with a brief explanation of the authentication method. Then, each participant is asked to fill in a questionnaire with background questions, followed by a set of questions regarding their experience of existing two-factor solutions. Next, the participant is introduced to the system and asked to perform three authentication attempts. Finally, the participant is requested to fill the last part of the questionnaire, in which he is asked about the experience of using the solution, according to the usability criteria described above.

3) Questionnaire Results

a: User Study Participants

The user study included 34 participants, all in the age group of 18-35. Out of the 34, 27 (79.4%) were male and 7 (20.6%)

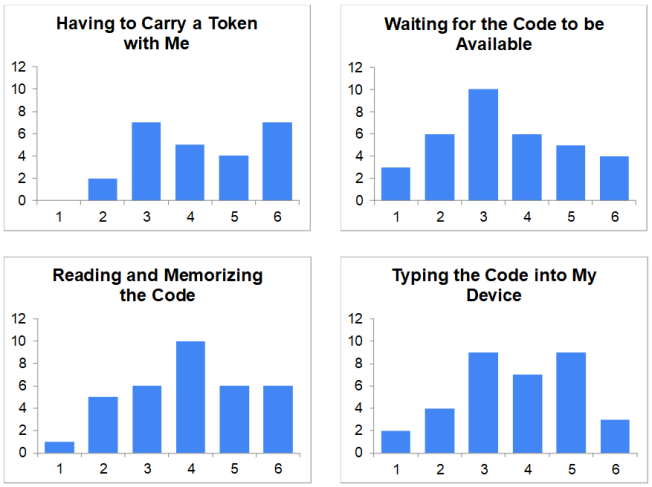


Figure 4. Results of the first part of the questionnaire. 1 signifies "least annoyed" and 6 signifies "most annoyed."

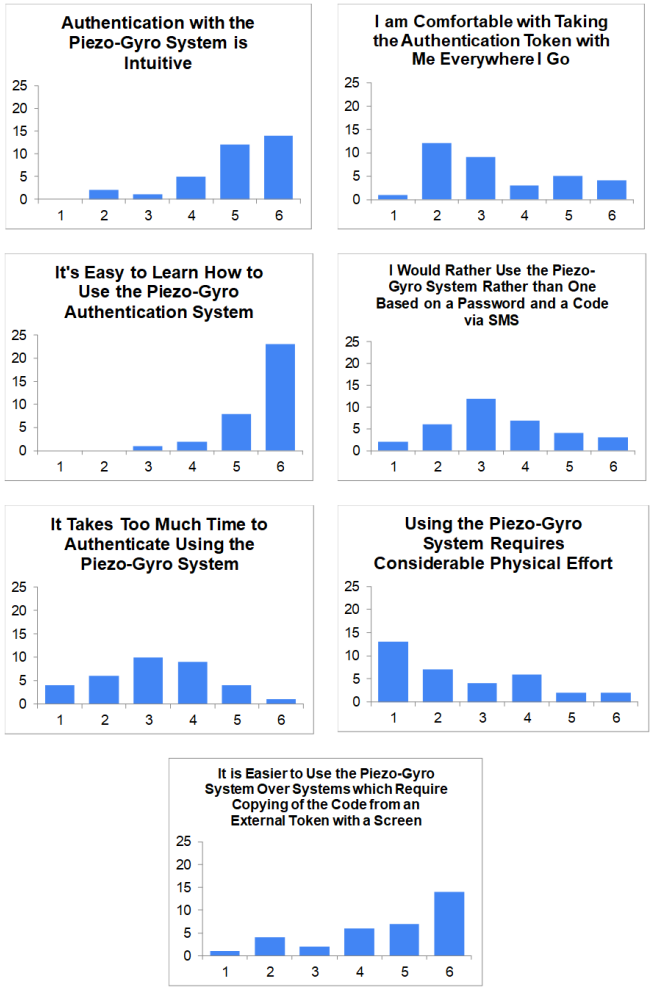


Figure 5. Results of the second part of the questionnaire. 1 signifies "strongly disagree" and 6 signifies "strongly agree."

were female. All participants claimed strong familiarity with using computers for daily tasks (5 or 6 on the scale of 1-6) and have used two-factor authentication solutions before.

Due to COVID-related limitations imposed on our human study, we were unable to reach participants suffering from motor impairments, and only studied one participant suffering from visual impairments that hinder his capability to use computers and mobile devices. We hypothesize that, compared to the general population, our user study group is more biased toward trying out new technologies, less limited in their visual and motor capabilities, and more comfortable following complex technical instructions. We hope that these biases can be addressed in future work.

b: Pre-experiment Questionnaire

In this part of the questionnaire we focused on "pain points" in existing two-factor authentication solutions. We asked each participant to rate several statements by how annoying they are, on a scale of 1 to 6. The results are presented in **Figure 4**.

Based on these results, we believe that our authentication system may help to reduce burden on users, since it spares them from the need to memorize and write the code by themselves (statements 3 and 4). Our scheme still involves carrying a token and waiting for the code to be transmitted (statements 1 and 2), a limitation it shares with similar token-based schemes.

c: Post-experiment Questionnaire

Participants answered these questions after experimenting with our implementation. The results are presented in **Figure 5**.

Looking at our first criterion *Nothing-to-Carry* (U3) we see that many users dislike carrying our token with them. This was also observed during the first part of the user study. The next criterion is *Physically-Effortless* (U4). Most participants felt the use of the system did not require considerable physical effort. This may be especially advantageous to people with accessibility needs. The third criterion is *Easy-to-Learn* (U5). User feedback indicates that the method is simple and intuitive. We note that during the interactive session users quickly understood how to use the token for authentication. The last criterion is *Efficient-to-Use* (U6). We configured the system to work in 10 bps, resulting in an authentication time of about 6 seconds. Some users felt this was too long to wait for authentication. We consider this as an issue to be improved in future work.

Our final two questions were aimed at measuring how participants perceive our solution. Indeed, many participants felt the solution is a viable and more comfortable alternative to existing schemes.

V. CONCLUSION

We have shown a **mature design** for a two-factor authentication system using the piezo-gyro channel. We explored the security, usability and deployability requirements the scheme

must meet, and designed enrollment and authentication processes that are secure and standards-compliant.

The system does not require the user neither to read out the one-time pad, memorize it, nor to type it in. As such, it can be useful in situations where users are temporarily or permanently unable to perform these tasks. Examples include vision-impaired users, motion-impaired users, users who need to authenticate while driving or while wearing eye and hand protection, and users with cognitive difficulties who cannot be expected to memorize long numeric sequences. For more flexibility, it is also possible to create a combined system which both displays the OTP and modulates it using the piezo-gyro channel, allowing the user to choose the method which is more appropriate for the current situation.

Three important remaining areas for future work are miniaturizing the device, improving its performance in noisy environments, and reducing the authentication time. Our implementation currently uses an off-the-shelf Arduino wired to a piezo-electric speaker, transistors, and batteries. Using an existing hardware design let us quickly and reliably develop a working proof of concept. This resulting system is large and consumes extra current, due to unnecessary circuitry in the Arduino board. Replacing the Arduino with a custom PCB based on the same microcontroller CPU will make the system smaller and extend its battery life, while allowing code reuse. To improve the decoding performance of the system outside the quiet lab setting, we can apply the machine learning methods suggested by Tharayil et al. [11], or the signal processing techniques described by Gao et al. [5]. Both of these works made effective use of the piezo-gyro channel while the device was being actively manipulated, even if the user was walking or running. To reduce the authentication time, the system can use a more powerful transducer, increasing the signal-to-noise ratio of the piezo-gyro signal and allowing a higher data rate. This may, however, have an effect on the battery life of the authenticator token.

The proof of concept we developed helped us show the viability of the method and explore several implementation aspects. Functional testing has shown that the solution works in a controlled lab setting across multiple devices. The user study showed that this solution has great potential as an alternative to existing solutions. Participants were open to this method, and a sizeable portion even showed preference over existing solutions. We hope to develop our design further, and believe it will contribute to wider acceptance of two-factor authentication by additional audiences. In particular, we are excited about the ability to increase the security afforded to users with impaired mobility, vision, and cognition.

ACKNOWLEDGMENTS

The authors would like to thank Omer Shwartz and Natan Vinshtok-Melnik for sharing their knowledge and proficiency in the hardware implementation stages.

References

[1] Mozhgan Azimpourkivi, Umut Topkara, and Bogdan Carbunar. Camera based two factor authentication through mobile and wearable devices. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol., 1(3):35:1–35:37, 2017.

[2] Joseph Bonneau, Cormac Herley, Paul C. van Oorschot, and Frank Stajano. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In IEEE Symposium on Security and Privacy, pages 553–567. IEEE Computer Society, 2012.

[3] Benyamin Farshteindiker, Nir Hasidim, Asaf Grosz, and Yossi Oren. How to phone home with someone else's phone: Information exfiltration using intentional sound noise on gyroscopic sensors. In WOOT. USENIX Association, 2016.

[4] Dinei A. F. Florêncio and Cormac Herley. A large-scale study of web password habits. In WWW, pages 657–666. ACM, 2007.

[5] Ming Gao, Feng Lin, Weiye Xu, Muertikepu Nuermaimaiti, Jinsong Han, Wenyao Xu, and Kui Ren. Deaf-aid: mobile IoT communication exploiting stealthy speaker-to-gyroscope channel, pages 53:1–53:13. ACM, 2020.

[6] Nikolaos Karapanos, Claudio Marforio, Claudio Soriente, and Srdjan Capkun. Sound-proof: Usable two-factor authentication based on ambient sound. In USENIX Security Symposium, pages 483–498. USENIX Association, 2015.

[7] T. Moore, T. Kosloff, J. Keller, G. Manes, and S. Shenoi. Signaling system 7 (ss7) network security. In The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002., volume 3, pages III–III, 2002.

[8] Magnus Nyström. The securid(r) SASL mechanism. RFC, 2808:1–11, 2000.

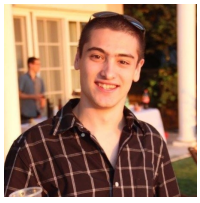
[9] Stefan Schneegass, Youssef Oualil, and Andreas Bulling. Skullconduct: Biometric user identification on eyewear computers using bone conduction through the skull. In CHI, pages 1379–1384. ACM, 2016.

[10] Yunmok Son, Hocheol Shin, Dongkwan Kim, Young-Seok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. Rocking drones with intentional sound noise on gyroscopic sensors. In USENIX Security Symposium, pages 881–896. USENIX Association, 2015.

[11] Kevin Sam Tharayil, Benyamin Farshteindiker, Shaked Eyal, Nir Hasidim, Roy Hershkovitz, Shani Houri, Ilia Yoffe, Michal Oren, and Yossi Oren. Sensor defense in-software (SDI): practical software based detection of spoofing attacks on position sensors. Eng. Appl. Artif. Intell., 95:103904, 2020.

[12] J. Zhang, X. Tan, X. Wang, A. Yan, and Z. Qin. T2fa: Transparent two-factor authentication. IEEE Access, 6:32677–32686, 2018.

[13] Bing Zhou, Jay Lohokare, Ruipeng Gao, and Fan Ye. Echoprint: Two-factor authentication using acoustics and vision on smartphones. In MobiCom, pages 321–336. ACM, 2018.



DAN ARAD is an M.Sc. student in the Department of Software and Information Systems Engineering in Ben-Gurion University of the Negev, Israel. He received his B.Sc. degree in Computer Science and Physics from The Hebrew University of Jerusalem, Israel in 2015. His research interests include embedded systems, reverse engineering, cyber security and FPGAs.

YOSSI OREN (SM' 17) received his M.Sc. degree in Computer Science from the Weizmann Institute of Science, Israel, and his Ph.D. degree in Electrical Engineering from Tel Aviv University, Israel, in 2008 and 2013 respectively. He is a Senior Lecturer (Assistant Professor) with the Department of Software and Information Systems Engineering in Ben-Gurion University, Israel. His research interests include implementation security (power analysis and other hardware attacks and countermeasures; low-resource cryptographic constructions for lightweight computers) and cryptography in the real world (consumer and voter privacy in the digital era; web application security).