

Analysis of Attacks on Continuous Authentication Methods and Ways of Defending Against Them

Iurii Matiushin, Vladimir Korkhov(✉), and Ilya Busko

St Petersburg State University, St Petersburg 199034, Russia
{y.matyushin,v.korkhov,i.busko}@spbu.ru

Abstract. User authentication remains one of the most pressing issues in modern cybersecurity, with traditional methods increasingly failing to meet the modern challenges. In particular, there is now a plethora of attacks traditional authentication algorithms are vulnerable to, and a successful attack can be seriously damaging for a system as a whole. For this reason, Continuous Authentication (CA) emerged as a new approach to securely authenticate users in a distributed system, with diverse methods using such technologies as machine learning, neural networks, behavioural biometrics, etc. However, several new attack vectors have been suggested that might compromise even CA-protected systems. In this article, we provide an analysis of possible attacks on various CA algorithms. Furthermore, we discuss the attacks' limitations, various factors that might improve authentication systems' resilience, as well as proposed defense mechanisms that could provide protection against these attacks. We conclude that most common CA methods today, while a step-up compared to traditional authentication methods, are still susceptible to being subverted, and implementing defense mechanisms and adopting security best practices should be an important consideration while designing a user authentication system.

Keywords: Continuous Authentication · Adversarial Attacks · Machine Learning · Cybersecurity

1 Introduction

1.1 User Authentication in Distributed Systems

User authentication, along with user identification and authorization, is one of the main components of the access granting process of any information system. In particular, to authenticate a user means to confirm a user's identity – that is to say, to make sure that the user is really who they claim to be.

There are various ways to confirm a user's identity, and over the decades, a great number of various authentication methods has been developed – from simply relying on a user knowing some secret piece of information (e. g. a password) to basing the authentication mechanism on a distinctive part of human anatomy or physiology (this

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2024
O. Gervasi et al. (Eds.): ICCSA 2024 Workshops, LNCS 14821, pp. 369–385, 2024.
https://doi.org/10.1007/978-3-031-65308-7_26

includes diverse biometric methods such as fingerprint scanning, face recognition, gait-based user recognition, and much more). The basic goal, however, remains the same no matter the principle used – to be able to tell, with high levels of accuracy, whether the person seeking to gain access to the system is a legitimate user or an impostor attempting to pass themselves off as a legitimate user.

User authentication is one of the most essential parts of an information system's security architecture. Its importance is obvious – if a system cannot tell a legitimate user apart from an impostor, it means that a malicious actor can compromise the system, causing serious damage. Just as importantly, a legitimate user can be prevented from accessing the information system.

1.2 Common Attacks on Authentication Methods

Just as authentication methods are numerous and diverse, various authentication attacks have also been developed [1].

An authentication attack can be understood as a sequence of actions aimed at subverting an authentication algorithm. Whereas user authentication aims to only allow legitimate users access to the protected system, the goal of an attack is to gain access in spite of not being a legitimate user. This means either obtaining the credentials used to confirm a user's identity or bypassing the credential part altogether and directly gaining access through some other means.

Below, some of the most common authentication attacks are listed, along with a brief description for each.

Brute force attack is the simplest form of attack on a password-based authentication system, being an attempt to simply guess a password via trying different possible symbol combinations. Other similar approaches include dictionary attacks (using a list of common passwords) and password spraying (trying several common passwords across a large number of users, thus bypassing a limit on authentication attempts for a single user).

Phishing refers to use of various social engineering techniques that aim to trick people into compromising their own security systems, whether by revealing authentication information to an attacker or by unknowingly installing malicious software. Various types include broad phishing, where an attacker targets a large group of users to try to gain access to one or a few accounts (similar to password spraying), and spear phishing, where few targets are carefully chosen and analyzed before launching the attack.

Credential stuffing makes use of the fact that a typical user has various accounts on multiple resources – dozens, or even hundreds separate pieces of authentication information – as well as of the fact that many users tend to recycle their passwords or otherwise use the same authentication information on different websites [2]. It means that if an attacker gains access to a certain user's credentials (e. g. from a website breach, which are common [3]), they can try to use the same credentials for the user's other accounts.

Man-in-the-middle attacks (MitM) include an attacker intercepting a network connection between a user and an information system the user is trying to authenticate in, and then getting access to the user's authentication credentials, or obtaining the session token and using it to log in under the user's guise.

Hybrid attacks combine two or more attack approaches. A typical example is a combination of dictionary and brute-force attacks, where the attacker assumes that the user might make their password a little different from a standard one that would appear in a dictionary. So the attacker uses a list of typical password, like one would in a regular dictionary attack, but also applies the brute-force approach to every password in the list, changing them slightly until a match is found [4].

Other attack types also exist, such as smudge attacks, thermal attacks, replay attacks, etc. [5, 6].

In numerous data breaches that occurred over the last several years, insufficiently secure authentication was a serious factor. According to 2023 Data Breach Investigations Report by Verizon [7], 74% of all breaches include such elements as use of stolen credentials and social engineering, and phishing in particular made up 12% of external attacks.

For this reason, traditional authentication schemes are widely considered to be "broken", i. e., not up to the task of protecting an information system against various possible types of attacks [8].

1.3 Continuous Authentication

The deficiencies of traditional user authentication, and the need to better protect the information systems' integrity, led to the development of new approaches to authentication. In particular, continuous authentication – or CA for short – is a new strategy that reimagines one of the most basic parts of securing a system: the notion of a "security perimeter".

Succinctly put, a major weakness of any traditional authentication system is the fact that the authentication process is performed only once – at the beginning of a user's interaction with the system. If the user, whether legitimate or not, "passes" the authentication stage, they get access to the system or a part of the system for the full duration of the system. No further actions to ensure the user's legitimacy are taken, and if an attacker hijacks the session – as it is possible to do via some of the attacks described above – the system is guaranteed to be compromised.

CA algorithms, on the other hand, continue to verify the user's identity throughout the session. This means that instead of relying on a single authentication result, which is unlikely to be perfectly accurate, the security mechanism can use an aggregate of multiple authentication results to determine whether the user is legitimate. Furthermore, even in case of session hijacking, the system remains protected, as the hijacker still has to undergo repeated authentication attempts.

In a previous work [9], we have analyzed the most common CA methods in use today. They are as follows.

Keystroke-based user recognition is based on the principle that each person's way of typing text via a keyboard is distinct and can be used to tell different people apart, in a similar manner to how one might identify a person's handwriting. Multiple factors can be used to establish a user's "signature"; how long it takes to press a key or release it, the length of pauses between key presses, analyzing ways of entering common words and key combinations, etc. Machine learning is a useful tool here, just as it is with other approaches to CA.

Face identification is a common approach in both traditional and continuous authentication, founded on the fact that each person's face has a number of "landmarks" – in other words, distinct characteristics – the combination of which can be used to confirm the person in question's identity, by comparing the digital representation of the features to a database entry corresponding to the relevant user. In CA, this process happens throughout the session, requiring constant or frequent usage of a video camera; on the other hand, each individual recognition does not need to be perfectly accurate – only after some number of attempts ending in mismatch does the authentication mechanism "raise a red flag".

Mouse movement recognition is a more recent approach, based on the unique way each user moves their mouse. Pressing and releasing mouse buttons is also a factor. Because a mouse and a keyboard are often used in tandem, there has also been some research on combining mouse movement-based and keystroke-based user recognition algorithms.

Touch gesture-based recognition is used as a CA method comparable with keystroke- and mouse movement-based recognition for devices that are equipped with a touchscreen. Here, the way a user "swipes" the screen is considered, based on such characteristics as a touch gesture's position, trajectory, finger pressure, etc. With mobile devices being increasingly widespread, this approach is also gaining popularity.

Behaviour-based recognition can be broadly defined as analyzing the actions a user usually performs throughout a session, and using them to build a model of a given user's regular behaviour. The actions considered can vary widely – from physical actions like walking to the patterns of user's interaction with the information system (e. g., sending and receiving data, using certain applications, etc.). If a user's actions during any given session differ from the regular patterns in a major way, the security mechanism will call for additional checks on the user's identity. This means that behaviour-based algorithms are usually employed in tandem with other means of user authentication.

A brief summary of the methods is shown in Table 1 below.

Table 1. Continuous authentication methods.

Method	Based on
Keystroke-based	How the user presses the keys, enters certain combinations, etc
Face recognition	Continuous observation of the user's face via a camera
Mouse movement	How the user moves the mouse, clicks the mouse buttons, etc
Touch gesture	How the user interacts with a touchscreen device, presses, swipes, performs multi-touch gestures, etc
Behaviour-based recognition	Whether the user's actions while interacting with the system resemble that user's regular action patterns

The list above is by no means exclusive, and new CA methods are constantly being developed and refined. For instance, in another article [10] we explore CA algorithms

that can be used in an IoT environment, including methods based on gait analysis, context cues, and other factors.

CA is effective in protecting against most attacks we described earlier [11]. However, some works have emerged recently exploring the possibility of compromising even CA-protected systems. In the remainder of this article, we investigate known adversarial attacks that could conceivably "defeat" CA, their estimated efficiency, as well as possible ways to ensure better protection against such attacks.

1.4 Article Structure

The article is organized as follows.

In Part 1, we briefly introduce the concept of user authentication, list common attacks on authentication systems, and discuss continuous authentication methods in use today.

In Part 2, we review the existing literature on the topic of attacks on continuous authentication.

In Part 3, we explore existing ways to attack various CA methods.

In Part 4, we go over the attacks' limitations, as well as possible ways to defend against the attacks.

In Part 5, we conclude the article, propose features and best practices for prospective CA systems, and outline future research directions.

2 Literature Review

In general, the topic of attacks on CA-protected systems remains little explored in the literature of the field [12]. However, several works have been published that detail possible ways to compromise the security of such systems.

In this section, we give a brief review of the current research on the topic of adversarial attacks on CA.

In **"Adversarial Attacks on Deep Learning Systems for User Identification based on Motion Sensors"** [13], C. Benegui and R. T. Ionescu analyze authentication systems based on motion sensors and deep learning methods. Several methods for generating adversarial examples that can aid the attackers with impersonating a legitimate user are studied. Experiments with various DNN architectures are carried out, and both architecture-specific and universal possible attacks are discovered. The authors show how adversarial input can be effective in subverting DNN-based mechanisms for confirming mobile user identity.

In **"Impact of injection attacks on sensor-based continuous authentication for. smartphones"** [14], L. Gonzalez-Manzano et al. focus on injection attacks that can possibly subvert user authentication based on behavioural biometrics. In particular, the authors explore a possibility of attacking a mobile phone authentication mechanism based on sensor data. Location, gyroscope, and accelerometer data are considered as authentication factors due to their widespread usage. The authors demonstrate how injection attacks can be effective against user behaviour-based CA even with lack of previous knowledge about the user. Estimated threat detection time in various attack

scenarios is also considered. Finally, the authors consider proposals about defending against injection-type attacks on smartphones.

In **"GANTouch: An Attack-Resilient Framework for Touch-based Continuous Authentication System"** [15], M. Agrawal et al. explore the topic of attacks on CA methods based on touch strokes, as well as ways to defend against such attacks. Several possible attack strategies are considered, as well as countermeasures against such attacks. The authors conduct experiments with both existing touch-based algorithms and their own novel CA framework, and demonstrate how to increase the approach's resilience against adversarial attacks. The question of gender fairness of touch-based methods is also touched upon.

In **"Continuous Authentication against Collusion Attacks"** [16], P. Lyu, W. Cai, and Y. Wang introduce a new type of attack model on CA called a collusion attack. The authors analyze the attack's feasibility and experimentally prove that their approach is able to successfully compromise certain gait-based CA methods. Furthermore, a new multi-cycle scheme to defend against collusion attacks is proposed.

In **"Template Inversion Attack against Face Recognition Systems using 3D Face Reconstruction"** [17], H. O. Shahreza and S. Marcel present a new attack that uses a geometry-aware generator network to generate 3D face images that can trick a face recognition-based CA system. They explore various attack scenarios, conduct experiments, and also discuss the limitations of the proposed attack, as well as certain ethical considerations.

In **"Malboard: A novel user keystroke impersonation attack and trusted detection framework based on side-channel analysis"** [18], N. Farhi, N. Nissim, and Y. Elovici develop a sophisticated attack on keystroke-based continuous authentication mechanisms. The authors demonstrate how their attack can evade specialized security tools and successfully attack a protected system in the majority of cases. They also present three detection modules that can detect keystroke injection attacks, including the novel attack they propose.

3 Adversarial Attacks on CA Methods

As we noted in the previous section, although the topic of adversarial attacks on information systems protected with continuous authentication mechanisms has been explored relatively little so far, several possible attack angles have emerged. Given the rising popularity of CA as a more secure alternative to traditional authentication methods, analyzing how CA methods themselves might be compromised, and how enhance their resilience against the suggested attacks, is an increasingly important problem.

In this section, various ways to attack continuous authentication-protected systems are explored and evaluated.

3.1 Injection Attack

One of the most promising directions in continuous authentication development is using CA to protect mobile users. It takes advantage of the fact that smartphones already collect various sensor data – including gyroscope data, accelerometer data, location information

and so on – that can be used to develop patterns that uniquely correspond to a certain user (for instance, one's gait, or a manner of walking) and can therefore be applied to the task of user identity verification.

However, the above assumes that the data provided by the phone's sensors are accurate and have not been changed by an outside actor. In reality, however, there are already several proposed methods to influence the sensors' output, from simply distorting it to establishing control over it. These methods can be both hardware-based (e. g. application of magnetic or acoustic pressure to a mobile device) and software-based (data-altering malware).

Two main attack scenarios have been proposed: one where the attacker has no previous knowledge about the victim – and thus does not know the sensor values typical for the user – and one where the attacker obtains knowledge about the victim beforehand, typically through using malware. In both cases, the attack itself includes the attacker somehow getting physical control of the victim's mobile device, and then influencing the device's sensors to inject falsified sensor values. The attack is considered successful if authentication is completed before the security system detects tampering, or the user realizes the device's disappearance.

Based on the attack scenario, two data injection strategies are available – Blind Rate (BR) injection and "Slogger" attack. Both essentially modify a series of sensor readings (formula) into new readings that should trick the authentication system. In BR, all features of the reading are modified by a fixed ratio, whereas the "Slogger" attack, which is only possible with previous knowledge, takes the user's maximum and minimum readings and injects the data randomly between them.

The authors of the article [14] investigate both attack scenarios in various contexts. They note that in some circumstances, the CA system makes a mistake up to 13.2% of the time.

For mobile device sensor data, the Sherlock dataset is used. This dataset was collected from 47 users for a maximum of 36 months, recording location data every minute and gyroscope and accelerometer data every 15 s. Location records consist of 12 distinct features, whereas gyroscope and accelerometer records consist of 90 features.

Injection attacks against mobile devices remain little explored, despite the very real possibility of such attacks succeeding [19].

3.2 Collusion Attack

Another increasingly common CA method is gait-based user recognition. It is based on the principle that each person walks in a distinct manner, which can be used as a way to confirm a human user's identity. This approach can be considered one of the types of biometric behaviour-based user authentication.

In one article [20], three different methods of gait-based user recognition are described: using machine vision, using floor sensors, and using wearable sensors. In all cases, user gait data are collected, preprocessed to reduce noise levels, and then analyzed via machine learning or other approaches to discover a pattern corresponding to a given user and then use it for authentication.

Gait-based approach is especially convenient due to widespread use of mobile devices, which can collect gait data based on the output of various sensors (see also

the subsection on injection attacks above). That said, it has several vulnerabilities the authors of [16] explore in their article.

One of the vulnerabilities in question is that, as shown in subsection above, sensor data can be manipulated by an outside force; this means that gait-based recognition is also theoretically vulnerable to injection attacks.

The other problem is that gait – unlike certain other biometric traits like fingerprints or eye retina – is relatively easy to observe, making it possible to emulate a target user's manner of walking and thus increase the chances that the authentication system mistakes an attacker for a legitimate user. Lyu, Cai, and Wang explore this type of attack.

The basis of their attack is essentially mimicry. By observing the target user's gait, either through video surveillance or in person, and then imitating the gait's distinctive characteristics, such as walking speed and rhythm, the height of the foot from the ground, elbow movement, etc., a sufficiently well-trained attacker has been shown to successfully "fool" a gait-based system.

In general, attacks on gait-based authentication can be classified as zero-effort or passive (the attacker having similar biometric characteristics to the victim) and adversary or active (the attacker actively trying to mimic the victim's biometrics). Here, "collusion attack" refers to the combination of the two.

The authors have gathered gait data from 20 members of an honour guard unit over the course of three months. The subjects are physically similar to one another in terms of age, height, weight, and gender; they were also trained to walk in a specified manner as a part of their duties.

According to the authors, their attack reached the success rate of 39%.

3.3 Template Inversion Attack

Face recognition (or face identification, as described in Part 1) is also a popular method to use in CA systems, and one of the most common biometric authentication mechanisms in general.

Face recognition algorithms use either 2D face images or 3D face models; in general, 3D face recognition is more complicated and computationally demanding, but also offers a higher level of accuracy.

Several attacks against face recognition have been developed. These include print attacks (using static face images), replay attacks (using prerecorded face videos), 3D mask attacks (physical masks of the target's face) [21].

Recently, a novel method of subverting face recognition-based CA has been developed, referred to as template inversion attack. Briefly, this attack, also called GaFaR (Geometry-aware Face Reconstruction), uses GANs (Generative Adversarial Networks) to reconstruct a 3D face model from facial features extracted from 2D face templates [16].

The attack uses a generative neural radiance field (GNeRF) method: a GAN is used to embed the appearance of an object's (a target user's face) in a latent space, and then the GAN's latent code along with camera parameters are fed to a neural radiance field (NeRF) for the rendering process. To train the network, a semi-supervised learning approach is used, based on both real and synthesized images.

There are two main attack scenarios. In the "white-box" scenario, the attacker knows the face recognition model's internal functioning and parameters. On the other hand, in the "black-box" scenario, this information is unknown to the adversary.

During the attack, a generator network learns the mapping from facial templates, and a GAN-based mapping network reconstructs 3D faces. An attacker can inject the generated 3D face model in the authentication process and use it to try to gain access by impersonating a legitimate user.

The authors use several datasets during the experimental stage of their research: the FFHQ dataset (70,000 high quality face images), the LFW dataset (12,233 images of 5,749 people), and the MOBIO dataset (150 people, 12 sessions per person).

The article claims a success rate of up to 17.14%.

3.4 Keystroke Impersonation Attack

Keystroke recognition is one of the oldest CA methods. It can be compromised, but CA tools can generally identify machine generated keystrokes if they are not similar to a human user's input – for instance, many existing keystroke injection attacks are characterized by typing speed not characteristic of a human user.

A new attack called Malboard was proposed that injects malicious commands via keystrokes that imitate a human user's biometric characteristics. It is conducted via a USB device masquerading as a regular keyboard [18].

The USB protocol has certain vulnerabilities that make it easier for an outside adversary to launch attacks. In particular, these are the fact that the host typically allows the USB device to send data that bypass any security mechanisms (e. g. malicious human interaction device (HID) codes can be sent straight to the operating system without any inspection by anti-virus software), as well as the fact that a USB device declares its own identity and characteristics, allowing a malicious USB device to impersonate other similar devices by sending the host false information. These facts make using malware-infected or even specifically built USB an extremely viable attack vector.

In the Malboard attack, a keyboard is physically modified to include a Teensy microcontroller, a USB Host Mini, and ESP8266 Wi-Fi component. Teensy is capable of modifying its own VID (vendor ID) and PID (product ID), making it possible to impersonate a regular keyboard and bypass most anti-keylogger protections (which largely rely on VIDs and PIDs). USB Host Mini makes it possible for Teensy to communicate with a keyboard as a host. Finally, ESP8266 allows Malboard to communicate with a distant attacker and to send the legitimate user's keystroke pattern data.

Once the user connect the keyboard to the computer, Teensy intercepts the user's keystrokes in the manner of a man-in-the-middle attack and uses them to build the user's profile. Once the user has not been active in a while, Malboard can start injecting malicious commands using the profile built. It can also communicate with the remote attacker to request instructions and commands.

The attack types include RSI (remote server injection) and PAI (physical access injection); in the former case, Malboard injects commands by itself as described above, although it can send requests to an outside server, whereas in the latter case, the attacker connects to Malboard and manually types commands, while Malboard modifies the attacker's typing pattern to resemble that of the user.

The typing data were gathered by the authors from 30 human subjects aged 20–38, all used to working with computers (students and university graduates). Each subject performed three distinct tasks, namely username and password input, credit card number input, and free-text input.

According to the authors, depending on the attack type and the detection mechanism employed by the user, Malboard evades notice in 83.3–100% of cases.

3.5 Attacks on Touch-Based CA

There are multiple types of attacks on touch-based CA systems (TCAS). Some of the most common ones – and easiest to implement – include zero-effort, population, and random-vector attacks. All of these attacks include creating adversarial examples in the form of vectors – sets of characteristics that are used by the CA system. The intent is to make the authentication mechanism mistake the generated vector for one related to the actual human user.

In a zero-effort attack, the attack vectors are chosen randomly among those generated by each and every impostor (a user other than the genuine user). This attack is the easiest one to replicate, though obviously the least efficient, since it makes no effort to make the attack vectors resemble the real ones. It is therefore often used to evaluate a TCAS's effectiveness as a baseline of sorts.

In a population-based attack, the attack vectors are generated by taking multiple users' vectors into account, e. g. by using the means of each characteristic. The result is referred to as a "master key" (cf. Creating an "average" male and female face based on a large dataset). A more sophisticated version creates multiple "master keys" by using clustering. In either case, access to a large number of samples is required.

In a random-vector attack, the attack vectors are generated randomly by utilizing prior knowledge, e. g. vector length and minimum/maximum values of the user's vector's characteristics. This method was found to be effective against behaviour-based biometric authentication methods [22].

The authors of [15] have carried out all three types of attacks against TCAS. Four datasets were used: BBMAS-Touch (117 users, 173 swipes per user), Serwadda (190 users, 273 swipes per user), HMOG (100 users, 2275 swipes per user), and UMDAA-02Touch (48 users, 4063 swipes per user).

The article finds that the results vary based on both attack type and authentication system type. The overall results are 7–12% for the zero-effort attack, 14–27.5% for the population attack, and 12.5–21.5% for the random-vector attack.

4 Ways of Defending Against the Possible Attacks

4.1 Defense Against Injection Attacks

The authors of the article [14] stress that mobile device sensors should not be thought of as trusted sources by default, since it is possible for an outside attacker to influence them both by using malware and by applying certain types of pressure on the device itself. The possibility of such an attack, according to the authors, should be considered by the design of the system from the beginning.

In their work, the authors introduce a threshold that determines how many failed authentication attempts – caused by sensor data differing from the original – it takes to classify the source as an attacker and to block the mobile device. The goal is that the authentication mechanism should detect an intruder both accurately (few false positives) and swiftly (so as not to give the attacker sufficient time to cause damage to the system).

According to the article's findings, how well a mobile device can defend against injection attacks largely depends on which sensors are used for user authentication, as well as which methods are employed to differentiate between "right" and "wrong" input values (that is to say, the correct choice of a classification mechanism, such as DSM, KNN, HAT, etc.).

In terms of sensor usage, whereas accelerometer and gyroscope injection results do not differ considerably depending on whether the attacker has prior knowledge about the target – and, therefore, whether the "Slogger" attack can be used – location data injection is harder to be detected if the attacker has information about the target. Thus, location-based CA could be more vulnerable in this case.

Often, the KNN (K-Nearest Neighbour) algorithm is used for the purposes of continuous authentication. In KNN, an instance is classified according to k of its closest members (by distance), where k is a changeable parameter. The authors find that in all cases, the smaller the k parameter used, the better the system is at detecting an intruder. Furthermore, they demonstrate that KNN shows better results in general compared to other algorithms, whether using accelerometer data, gyroscope data, or location data.

4.2 Defense Against Collusion Attacks

The authors of the article [16] propose a defense model that is aimed at resisting collusion attacks, as well as using fewer resources by not introducing additional sensors for CA purposes. It is based on a standard model of gait authentication.

The defense model consists of several steps.

The first step is data collection. A mobile device carried by a user records their motion data while the user is walking. Only accelerometer data are used.

The next step is data preprocessing. First, the motion data are extracted from the raw data using the sliding window; then, noise is removed from the data using a filter; finally, the motion data are converted into a sequence of cycles of a certain length. Abnormal cycles (such as atypical motion caused by the user stumbling) are removed.

The third step is to transform the data from the device's coordinate system by using the device's body inclination angles.

The final step is to compare the resulting cycles in order to measure their similarity to those of a genuine user. KNN classification algorithm is used for this purpose, and five cycles closest to the "center" (the cycle with the smallest sum of distances from other cycles) are submitted for comparison.

The proposed defense system uses confidence scores to determine if the user should be authenticated. A threshold is set, and the confidence score is calculated based on whether the resulting distance (similarity between the submitted cycles and the genuine user's template) is above the threshold. In order for the authentication to be successful, the confidence score must exceed 50%.

4.3 Defense Against Template Inversion Attacks

The authors of the article [17] introducing a new template inversion attack note that several factors can limit their attack's success.

One of these factors is bias in the generated faces concerning such characteristics as race and age. In particular, the authors note that their adversarial examples generated during a white-box attack showed bias for dark skin and old age. One possible reason for the bias's existence is the datasets used for training the models being themselves inherently biased.

Another factor is that, surprisingly, face recognition models with higher accuracy are more vulnerable to the proposed attack. On the contrary, models with worse recognition performance are associated with more failed attack attempts.

Although defense strategies are not discussed in detail, the article encourages the development of more secure face recognition systems, as well as using biometric template protection schemes proposed in the literature.

4.4 Defense Against Keystroke Impersonation Attacks

As the authors of the Malboard attack [18] noted, most keystroke impersonation attacks can be detected by existing security systems that analyze keystroke characteristics and dynamics and detect non-human patterns. However, more sophisticated attacks such as Malboard are capable of imitating all of a regular keyboard's features, so additional defense strategies are required.

Three main strategies include manual inspection (checking once before connecting a device), continuous inspection (regular inspections throughout the device's use), and combined inspection (combining both previous strategies). Since Malboard relies on hardware installed in a keyboard, detecting the malicious device with the naked eye is possible.

In order to detect the Malboard attack, a detection framework can make use of the following factors: power consumption, vocal keystroke sound delay, and typo deletion. Power consumption refers to the fact that additional hardware requires additional power to function, vocal keystroke sound delay is the result of Teensy acting as the middleman inside the keyboard, and typo deletion makes use of the fact that Malboard is not capable of handling keyboard corrections like a human user would.

There are limitations both to the Malboard attack (multiple person using the keyboard, multi-factor authentication) and to some detection methods (noise preventing keyboard sound analysis, typo deletion detection decreasing usability, etc.).

4.5 Defense Against Touch-Based Attacks

While studies such as [23] have suggested that raw data level distance-based biometric authentication methods are more attack resilient, they also have higher error rates compared to machine learning-based ones. Thus, the authors of [15] chose a machine learning-based method for their authentication mechanism.

The article introduces a Generative Adversarial Networks assisted TCAS framework (G-TCAS) and compares it against traditional, or "vanilla", TCAS methods (referred to as V-TCAS).

Compared to V-TCAS, which work by extracting features from a genuine user's touch gestures and then train the authentication models based on these features, G-TCAS adds a pair of GANs into the process. The first, Genuine-GAN, generates swipes similar to those of a genuine user. The second, Impostor-GAN, generates swipes similar to impostor swipes (real swipes belonging to the impostors). Both GANs consist of a discriminator and a generator, trained simultaneously. The goal of including an Impostor-GAN is to help the authentication framework separate the genuine data from the injected one more effectively.

According to the authors, GAN-assisted touch-based continuous authentication mechanisms are more resilient to all three types of attacks discussed in the previous part (zero-effort, population, and random-vector attacks).

The authors also note that a population-based attack is the most damaging type of the three, and encourage future researchers to test their systems under all three attack types so as to avoid misleading performance results (e. g. an authentication mechanism might be highly resilient to a zero-effort attack, but less so to a population-based or a random-vector attack).

4.6 Attack and Defense Summary

A short summary of proposed attacks on various continuous authentication-protected systems and possible ways to defend against them is presented in Table 2.

Table 2. Attacks on CA-protected systems.

Attack	Description	Experimental dataset(s)	Success rate, %	Defense strategies
Injection attack	Capturing a mobile device and injecting sensor data via malware or magnetic/acoustic pressure, either at random or based on previous knowledge of the user	Sherlock dataset (47 users, 36 months)	13.2	Choosing hard-to-spoof sensor types and ML techniques, not treating sensors as trusted sources
Collusion attack	Mimicking the user's gait based on observing their gait and obtaining various details of the gait	Original dataset (20 subjects, 3 months)	39	A multi-step defense mechanism analyzing and transforming gait data, using confidence scores to determine the user's authenticity

(*continued*)

Table 2. (*continued*)

Attack	Description	Experimental dataset(s)	Success rate, %	Defense strategies
Template inversion attack	Generating a 3D reconstruction of the user's face from 2D face images	FFHQ (70,000 images) LFW (12,233 images of 5,749 people) MOBIO (150 people, 12 sessions per person)	< 17.14	Exploiting bias in adversarial examples, employing biometric template protection schemes
Keystroke impersonation attack	Keystroke injection via a keyboard-impersonating USB device, which intercepts keystroke data and builds the victim's profile	Original dataset (30 subjects, 3 tasks per subject)	83.3–100	Detection based on power consumption, keystroke sound, and typo deletion
Attacks on touch-based CA	Zero-effort attack (use random existing vectors) Population attack (generate average vectors based on large datasets) Random-vector attack (generate random vectors based on previous knowledge)	BBMAS-Touch (117 users, 173 swipes per user) Serwadda (190 users, 273 swipes per user) HMOG (100 users, 2275 swipes per user) UMDAA-02Touch (48 users, 4063 swipes per user)	7–12 (Zero-effort) 14–27.5 (Population) 12.5–21.5 (Random-vector)	Using GAN-assisted touch-based authentication, testing authentication methods against various attack types

5 Conclusions and Future Work

5.1 Overall Conclusions

Continuous authentication is a novel approach to user authentication in information systems. Its main principle is to confirm the user's identity throughout the session, as opposed to just once at the beginning of the user's interaction with the system. Various CA algorithms have been developed, based on factors ranging from keystroke dynamics to user gait.

CA is, as a rule, significantly more secure compared to traditional authentication, and can protect a system against multiple common attack types. However, as we have seen, CA methods are far from being invulnerable from outside influence, and it is possible for attackers to subvert them and gain illegitimate access to a system.

In particular, we have explored injection attacks, collusion attacks, template inversion attacks, keystroke impersonation attacks, as well as attacks on touch-based CA including zero-effort, population, and random-vector attacks. All those attacks are aimed at different types of CA and have demonstrated varying levels of effectiveness, but it can be stated with confidence that the most common CA methods as of today are all potentially vulnerable to certain types of attacks.

Ways to protect against the aforementioned attacks have also been suggested, as well as the attacks' limitations. In many cases, choosing the best available algorithms and optimal input data drastically improves the CA methods' resilience, whereas in other cases, developing new and improved defensive schemes is key.

Machine learning is an essential tool for implementing efficient continuous authentication schemes. ML-based methods were used as a part of defense mechanisms against various types of attacks investigated in this article. Investigating the best ML algorithms for application in a user authentication context, as well the optimal ways to choose their parameters, is therefore an important research direction in the field of cybersecurity.

5.2 Proposals for Resilient CA Methods

Based on the research carried out in this article, we can propose several features and best practices that could potentially make future continuous authentication methods more secure in the face of the novel attacks.

Firstly, a continuous authentication system should use confidence scores and set thresholds for recognizing a given user as a possible threat. CA, like any other security system, always involves a trade-off between security and user convenience; modifiable thresholds allow the authentication mechanism to be more flexible, allowing a certain amount of non-standard responses from a legitimate user while also quickly and precisely identifying potential threats based on repeated authentication failures.

Secondly, several different data sources can be used for authentication. We have seen that some types of data are more vulnerable to certain attack compared to others (e. g., location data compared to accelerometer and gyroscope data). If possible and practical, a hybrid authentication system should be developed.

Thirdly, the authentication mechanism should be tested against various possible attacks, instead of using only the simplest ones to simulate. Furthermore, periodical checks are also essential, as seen with the keystroke impersonation attack. Various hard-to-fake characteristics can be used to make sure a CA system was not compomised.

Fourthly, the number of data sources treated as automatically trustworthy should be limited as much as possible. As shown above, information such as mobile device sensor data can be tampered with by outside forces, leading to a possible vulnerability in an authentication mechanism based on this information. Similarly, USB devices that can bypass security software and send false information to the system are also a potential source of danger. CA methods should follow the Zero-Trust paradigm – "never trust, always verify".

Finally, the novel attacks such as ones analyzed above should be kept in mind while designing an authentication system. CA is effective against "traditional" attack types, both simple and hybrid, but the methods and algorithms used to achieve continuous authentication can also have inherent weak points. As an example, gait-based authentication means that a user's gait can potentially be observed and copied, and an attack has already been developed that takes advantage of this fact.

5.3 Future Work Directions

Future work directions include further research on the topic of attacks on continuous authentication methods and the defense mechanisms capable of increasing the methods' resilience. In addition, we plan to develop a novel CA framework and test it against the new attacks discussed herein.

Acknowledgement. Supported by Saint Petersburg State University, project ID: 95438429.

References

1. Wang, X., et al.: Attacks and defenses in user authentication systems: a survey. J. Netw. Comput. Appl. **188** (2021)
2. Awad, M., et al.: Password security: password behavior analysis at a small university. In: 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), pp. 1–4. IEEE, Khaimah, United Arab Emirates (2016)
3. IBM Cost of a Data Breach Report 2023. https://www.ibm.com/reports/data-breach. Accessed 20 Mar 2024
4. Newlin Rajkumar, M., Dhurka, V., Kayathri, P.: Survey a secured privacy authentication with recovery. World Sci. News **37**, 265–278 (2016)
5. Identity Attacks that Exploit Your Broken Authentication. https://www.okta.com/resources/whitepaper/5-identity-attacks-that-exploit-your-broken-authentication. Accessed 25 Mar 2024
6. Mayrhofer, R., Sigg, S.: Adversary models for mobile device authentication. ACM Comput. Surv. **54**(9), 1–35 (2021)
7. Verizon 2023 Data Breach Investigations Report. https://www.verizon.com/business/en-nl/resources/reports/dbir/. Accessed 06 Mar 2024
8. Zhang, Y., et al.: Trustworthy interaction model: continuous authentication using time–frequency joint analysis of mouse biometrics, Behav. Inform. Technol. (2024)
9. Matiushin, I., Korkhov, V.: Continuous authentication methods for zero-trust cybersecurity architecture. In: Gervasi, O., et al. Computational Science and Its Applications – ICCSA 2023 Workshops. ICCSA 2023. Lecture Notes in Computer Science, vol. 14109. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-37120-2_22
10. Matiushin, I., Korkhov, V.: Continuous Authentication in Internet of Things Systems [Manuscript submitted for publication] (2023)
11. Hussain Al-Naji, F., Zagrouba, R.: A survey on continuous authentication methods in Internet of Things environment. Comput. Commun. **163**, 109–133 (2020)
12. Pourshahrokhi, N., et al.: Generative adversarial attacks on motion-based continuous authentication schemes. In: 2021 14th International Conference on Security of Information and Networks (SIN), Edinburgh, United Kingdom, pp. 1–6. IEEE (2021)
13. Benegui, C., Ionescu, R.T.: Adversarial attacks on deep learning systems for user identification based on motion sensors. In: Yang, H., Pasupa, K., Leung, A.CS., Kwok, J.T., Chan, J.H., King, I. (eds.) Neural Information Processing. ICONIP 2020. Communications in Computer and Information Science, vol. 1333. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-63823-8_85
14. Gonzalez-Manzano, L., et al.: Impact of injection attacks on sensor-based continuous authentication for smartphones. Comput. Commun. **163**, 150–161 (2020)

15. Agrawal, M., et al.: GANTouch: an attack-resilient framework for touch-based continuous authentication system. IEEE Trans. Biomet. Behav. Identity Sci. **4**(4), 533–543 (2022)
16. Lyu, P., Cai, W., Wang, Y.: Continuous authentication against collusion attacks. Sensors **22**(13), 4711 (2022)
17. Shahreza, H.O., Marcel, S.: Template inversion attack against face recognition systems using 3D face reconstruction. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 19662–19672. IEEE, Paris, France (2023)
18. Farhi, N., Nissim, N., Elovici, Y.: Malboard: a novel user keystroke impersonation attack and trusted detection framework based on side-channel analysis. Comput. Secur. **85**, 240–269 (2019)
19. Sikder, A.K., Aksu, H., Uluagac, A.S.: 6th sense: a context-aware sensor-based attack detector for smart devices. In: 26th USENIX Security Symposium, USENIX Security 17, pp. 397–414. USENIX, Vancouver, Canada (2017)
20. Gafurov, D.: A survey of biometric gait recognition: approaches, security and challenges. In: Proceedings of the Annual Norwegian Computer Science Conference, Olso, Norway (2007)
21. Facial presentation attacks detection, an overview. https://alicebiometrics.com/en/facial-presentation-attacks-detection-an-overview. Accessed 19 Mar 2024
22. Zhao, B.Z., Asghar, H.J., Kaafar, M.A.: On the resilience of biometric authentication systems against random inputs. In: Proceedings of the Network and Distributed Systems Security Symposium. The Internet Society, San Diego, USA (2020)
23. Pagnin, E., Dimitrakakis, C., Abidin, A., Mitrokotsa, A.: On the leakage of information in biometric authentication. In: Meier, W., Mukhopadhyay, D. (eds.) Progress in Cryptology -- INDOCRYPT 2014. INDOCRYPT 2014. Lecture Notes in Computer Science(), vol. 8885. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13039-2_16