CABA: Continuous Authentication Based on BioAura

Arsalan Mosenia, *Student Member, IEEE,* Susmita Sur-Kolay, *Senior Member, IEEE,* Anand Raghunathan, *Fellow, IEEE,* and Niraj K. Jha, *Fellow, IEEE*

This paper can be cited as: A. Mosenia, S. Sur-Kolay, A. Raghunathan and N. K. Jha, "CABA: Continuous authentication based on BioAura," in IEEE Trans. Computers (TC), vol. 66, no. 5, pp. 759-772, May 1 2017
The latest version of this manuscript is available on http://ieeexplore.ieee.org/document/7723816/

***Abstract*—Most computer systems authenticate users only once at the time of initial login, which can lead to security concerns. Continuous authentication has been explored as an approach for alleviating such concerns. Previous methods for continuous authentication primarily use biometrics, e.g., fingerprint and face recognition, or behaviometrics, e.g., key stroke patterns. We describe CABA, a novel continuous authentication system that is inspired by and leverages the emergence of sensors for pervasive and continuous health monitoring. CABA authenticates users based on their BioAura, an ensemble of biomedical signal streams that can be collected continuously and non-invasively using wearable medical devices. While each such signal may not be highly discriminative by itself, we demonstrate that a collection of such signals, along with robust machine learning, can provide high accuracy levels. We demonstrate the feasibility of CABA through analysis of traces from the MIMIC-II dataset. We propose various applications of CABA, and describe how it can be extended to user identification and adaptive access control authorization. Finally, we discuss possible attacks on the proposed scheme and suggest corresponding countermeasures.**

***Index Terms*—Authentication, behaviometrics, biometrics, biostreams, biomedical signals, continuous authentication, machine learning, security, wearable medical devices.**

I. Introduction

Authentication refers to the process of verifying a user based on certain credentials, before granting access to a secure system, resource, or area [1]. Traditionally, authentication is only performed when the user initially interacts with the system [2]. In these scenarios, the user faces a knowledge-based authentication challenge, e.g., a password inquiry, and the user is authenticated only if he offers the correct answer, e.g., the password.

Although one-time authentication has been the dominant authentication mechanism for decades [3], several issues spanning user inconvenience to security flaws have been investigated by researchers [4], [5]. For example, the user has to focus on several authentication steps when he tries to unlock a smartphone based on a password/pattern-based authentication method. This may lead to safety risks, e.g., distraction when the user is driving. A serious security flaw of one-time authentication is its inability to detect intruders after initial authentication has been performed. For example, an unauthorized user can access private resources of the authorized user if the latter leaves his authenticated device unattended, or forgets to log out [6].

The above concerns have led to the investigation of continuous authentication mechanisms. Such mechanisms monitor the user's interactions with the device even after initial login to ensure that the initially-authenticated user is still the one using the device. Initial efforts in this direction were based on simple security policies that lock the user's device after a period of inactivity, and ask the user to re-enter the password. However, such schemes may be annoying to users while they still expose a window of vulnerability, leaving much room for improvement [7]. Thus, a rapidly-growing body of literature on the usage of biometrics, i.e., strongly-reliable biological traits such as facial features, and behaviometrics, i.e., measurable behavior such as frequency of keystrokes, for continuous authentication has emerged in the last decade [6], [8].

Recently, wearable medical sensors (WMSs), which measure biomedical signals, e.g., heart rate, blood pressure, and body temperature, have drawn a lot of attention from researchers and begun to be adopted in practice [9], [10]. A recent report by Business Insider [11] claims that 33 million wearable health monitoring devices were sold in 2015. It forecasts that this number will reach 148 million by 2019, and continue to grow rapidly thereafter. We suggest that, since such biomedical signals will be collected anyway for health monitoring purposes, they can also be used to aid authentication. The use of continuously-collected biomedical data for user verification and identification seems promising for three reasons. First, if the biomedical signals are collected by WMSs for medical purposes, using them for authentication does not require any extra device that is not already on the body. Second, this information is collected transparently to the user, i.e., with minimal user involvement. Third, unlike traditional biometrics/behaviometrics, e.g., face features and keystroke patterns, information that may frequently become unavailable, the stream of biomedical signals collected by WMSs is always available when the person is wearing WMSs.

In this paper, we present CABA, a transparent continuous authentication system based on an ensemble of biomedical signal streams (Biostreams in short) that we call *BioAura* [1]. A

Acknowledgments: This work was supported by NSF under Grant no. CNS-1219570 and Project X Fund from Princeton University.

Arsalan Mosenia is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA (e-mail: arsalan@princeton.edu).

Susmita Sur-Kolay is with the Advanced Computing and Microelectronics Unit, Indian Statistical Institute, Kolkata 700108, India (e-mail: ssk@isical.ac.in).

Anand Raghunathan is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA (e-mail: raghunathan@purdue.edu).

Niraj K. Jha is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA (e-mail: jha@princeton.edu).

[1] Aura is traditionally defined as the energy field around a person. Analogously, we use the term BioAura to define the biological field around a person, manifested as a set of Biostreams.

Biostream is a sequence of biomedical signal samples that are continuously gathered by a WMS for medical diagnosis and therapeutic purposes. The most important difference between a Biostream and a biometric trait is that a Biostream alone does not have enough discriminatory power to distinguish individuals. Thus, an authentication decision based on a single Biostream, e.g., body temperature or blood pressure, is unlikely to be sufficiently discriminative. However, when multiple Biostreams are combined into a BioAura, it leads to a powerful continuous authentication scheme.

Our key contributions can be summarized as follows:

1) We suggest a list of design requirements for any continuous authentication system.
2) In order to analyze the discriminatory power of BioAura, we propose a continuous authentication system based on BioAura (called CABA) and investigate it from both accuracy and scalability perspectives.
3) We suggest an adaptive authorization scheme and describe how it can be used to alleviate user inconveniences associated with the use of continuous authentication systems that might falsely reject the user.
4) We describe various possible attacks against the proposed continuous authentication system along with several countermeasures to prevent such attacks.

The rest of the paper is organized as follows. Section II describes the requirements that should be targeted in the design of continuous authentication systems and discusses how CABA addresses such requirements. Section III describes BioAura and the Biostreams that form the proposed BioAura. Section IV discusses the scope of CABA applications. Section V describes the CABA prototype and our experimental setup. Section VI investigates CABA from both accuracy and scalability perspectives. Section VII describes how CABA can support identification, i.e., the process of recognizing a user without knowing his user ID. Section VIII presents an adaptive authorization scheme that can be used along with CABA to enhance user convenience. Section IX discusses possible attacks against the proposed authentication system and describes different countermeasures against each attack. Section X discusses related work and compares CABA with previously-proposed continuous authentication systems. Section XI briefly describes possible privacy concerns surrounding the use of biomedical signals, how CABA can be used as a stand-alone one-time authentication system, and the effects of temporal conditions on authentication results. Finally, Section XII concludes the paper.

II. Desirable Authentication Requirements

In this section, we first describe the desirable requirements that every continuous authentication system must satisfy. Then, we discuss how CABA addresses such requirements.

A. Design-octagon

Even though several continuous authentication systems have been proposed in the past, they have not been evaluated against a comprehensive list of design requirements. A few studies, e.g., [12]–[14], consider a small set of requirements, e.g., cost and accuracy. However, there is no standard list of design requirements that a continuous authentication system must satisfy. We suggest such a list below. We call it the *Design-octagon* since it comprises eight design requirements (Fig. 1):

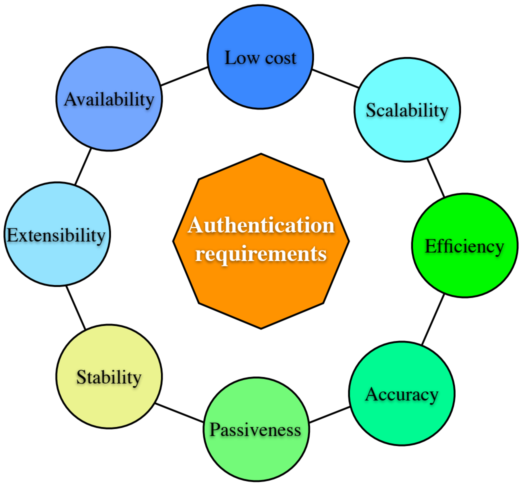


Fig. 1. Design-octagon: Desiderata for a continuous authentication system.

Passiveness: A user-friendly system must not require frequent user involvement [15]. For example, if the authentication system asks the user to re-enter his credentials often, it may be quite annoying to the user [13].

Availability: The system should provide a reliable authentication system at all time instances [13]. Lack of continuous availability is a significant drawback of several previously-proposed continuous authentication systems – they may often fail due to a lack of sufficient information [16]. For instance, a keyboard-based system may unintentionally reject a legitimate user when he is watching a movie and not using the keyboard.

High accuracy: Undoubtedly, the most important requirement of every authentication system is high accuracy. The system should be able to confidently and accurately distinguish legitimate users from impostors, and reject impostors' requests.

Scalability: The system should be able to handle a growing amount of work when the number of users increases [3]. In particular, its time and space complexity should increase modestly with an increase in the number of users [17].

Efficiency: A short response time, i.e., the time required to capture a test sample, process it, and provide a decision, is very desirable for two reasons. First, it is desirable for the system to quickly authenticate a legitimate user and reject an impostor to ensure user convenience [3]. Second, security may also suffer if there is an appreciable delay. For example, if authorization takes five minutes, an impostor may be able to control the system and access restricted resources in that five-minute timeframe, while the system is still processing.

Low cost: Cost is an important factor in authentication systems used in low-security environments, e.g., in personal computers [13], [18]. In such environments, the cost of adding or modifying the authentication system should ideally be negligible. Thus, systems that do not need extra peripherals, such as retina scanners, would be generally preferred. However, for highly-secure environments, e.g., military bases, expensive authentication systems could be deployed [14].

Stability: Any trait that is recorded for processing for authentication purposes must ideally have only slight changes or maintain its pattern over a certain time period [18], [19].

Extensibility: The authentication system should be able to function on a wide variety of devices regardless of underlying hardware. Ideally, the system should not require dedicated hardware. One of the advantages of password-based authentication is that it can be easily extended to protect a large number of systems, devices, and resources with minimal system modification [3].

B. Addressing desirable requirements

In this subsection, we describe how CABA ensures all of the requirements discussed above.

Passiveness: In CABA, passiveness is addressed through the use of WMSs. These are small and compact sensors that are specifically designed to take the passiveness requirement into account, since continuous health monitoring needs to minimize user involvement. Thus, passiveness is not only a very desirable requirement for continuous authentication, but also a significant consideration in designing WMSs. Unfortunately, major biometrics-based systems, e.g., fingerprint-based, do not provide a high level of passiveness.

Availability: The use of WMSs as capture devices also ensures a continuous stream of information since this is also required for continuous health monitoring. However, neither biometrics nor behaviometrics guarantees availability. For example, keyboard/mouse-based continuous authentication systems fail when the user stops using the dedicated peripherals.

Accuracy: The accuracy of CABA is extensively investigated in Section VI in various experimental scenarios. Section X demonstrates that the accuracy of CABA, which is based on an ensemble of weakly discriminative Biostreams, is comparable to previously-proposed systems, which are based on strong biometrics.

Scalability: The scalability of CABA is investigated in Section VI based on two scalability metrics (time complexity and space complexity). Our analysis shows that an increase in the number of users can be easily handled in this system.

Efficiency: Authentication can be done in a few milliseconds. For each authentication attempt, the user can immediately provide the required data since the data are already collected using WMSs. The efficiency of the system is described in more detail in Section VI.

Low cost: As discussed later in Section III, the proposed BioAura consists of Biostreams that are collected for continuous health monitoring. If the user is already using a continuous health monitoring system, CABA can offer continuous authentication with minimal cost.

Stability: Our investigations of different Biostreams and their high authentication accuracy over different timeframes demonstrate that the collected Biostreams maintain their pattern over time. Therefore, they can be used as authentication traits.

Extensibility: By decoupling the collection of Biostreams from the authenticating device, CABA can be implemented in any general-purpose computing device with sufficient memory capacity and computation power. Unfortunately, neither biometrics- nor behaviometrics-based systems provide significant extensibility. For example, the nature of keyboard/mouse-based authentication schemes inherently limits their applications, i.e., they can only be used for implementing an authentication mechanism in a system that has a keyboard or a mouse.

III. BioAura

In this section, we first briefly describe how Biostreams can be collected using WMSs. Then, we discuss which Biostreams constitute the BioAura.

As mentioned earlier, BioAura is an ensemble of Biostreams, which are gathered by WMSs for medical diagnosis and continuous health monitoring. The most widely-used scheme for continuous health monitoring consists of two classes of components: (i) WMSs and (ii) a base station [20]. All WMSs transmit their data to the base station either for further processing or long-term storage. In recent years, smartphones have become the dominant base station since they are powerful and ubiquitous, and their energy resources are less limited relative to WMSs [20], [21]. Fig. 2 illustrates a simple continuous health monitoring system that consists of several small lightweight WMSs, which transmit their biomedical data to the smartphone over a Bluetooth communication link.

Smartphones can perform simple preprocessing to extract values of some important features from the data, and transmit those values. In CABA, the smartphone first executes a very simple feature extraction function that computes the average value of the samples in each Biostream over the last one-minute timeframe of data. Then, it only transmits a feature vector that contains these average values.

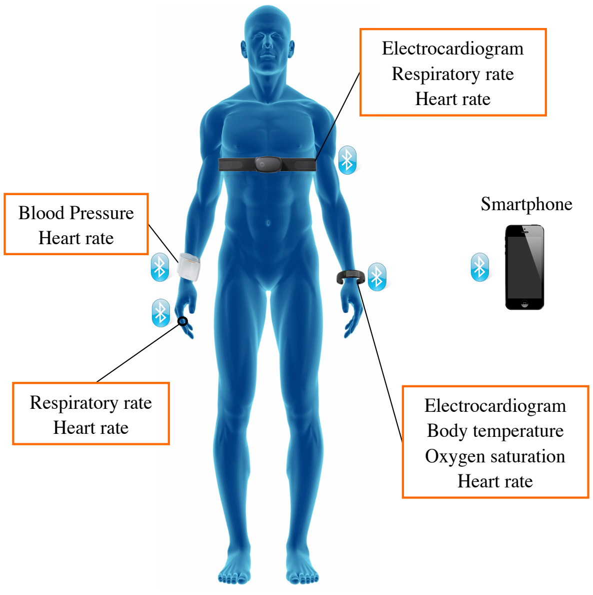


Fig. 2. A continuous health monitoring system consisting of several small lightweight WMSs that transmit biomedical data to the smartphone.

As mentioned earlier in Section I, with the expected widespread use of WMSs, CABA can be used to provide a continuous authentication system as a side benefit of continuous health monitoring systems. Our proposed BioAura consists of Biostreams that are essential for routine continuous health monitoring, and their collection needs minimum user involvement. Such Biostreams are expected to be included in long-term continuous health monitoring systems.

Table I shows the most widely-used Biostreams, their abbreviations/notations used in the medical literature, and their units [20], [22]. In this paper, we exclude the first three ones from the proposed BioAura, and include the other nine. Next, we discuss why the three Biostreams are excluded.

Electroencephalogram (EEG): EEG is excluded from BioAura because it cannot be conveniently captured. The current method for capturing EEG requires the user to wear a cap. Moreover, its capture devices cannot be miniaturized further because electrodes need to form a minimum diameter to be noise-robust [23].

Electrocardiogram (ECG): Even performing a low-complexity feature extraction on one minute of ECG signals requires at least $400\times$ more operations than performing a simple statistical feature extraction, e.g., averaging, on the respiratory rate values [24]. This would place a significant computational and energy demand on battery-powered devices such as smartphones and wearables. If we try to avoid the preprocessing, i.e., feature extraction, on the smartphone and just transmit the ECG signals to the authentication system, this would also entail significant energy consumption since ECG waveforms contain at least 200 samples/s [20], [25].

Blood glucose (BG): BG is excluded because currently the devices that measure BG are invasive, i.e., they require a sample of the user's blood.

Although we have currently used nine Biostreams to form the BioAura in the prototype implementation, CABA need not necessarily be limited to these nine. As other compact WMSs become available in the future, they could also be made part of the BioAura.

TABLE I
BIOSTREAMS, THEIR ABBREVIATIONS/NOTATIONS, AND UNITS

Biostream	Abbreviations/Notations	Unit
Electroencephalogram	EEG	μV
Electrocardiogram	ECG	μV
Blood glucose	BG	mg/dL
Arterial systolic blood pressure	ABPSYS	$mmHg$
Arterial diastolic blood pressure	ABPDIAS	$mmHg$
Arterial average blood pressure	ABPMEAN	$mmHg$
Heart rate	HR	$1/min$
Pulmonary systolic artery pressure	PAPSYS	$mmHg$
Pulmonary diastolic artery pressure	PAPDIAS	$mmHg$
Body temperature	T	$Celsius$
Oxygen saturation	SPO2	%
Respiratory rate	RESP	$1/min$

IV. Scope of Applications

In this section, we describe the possible applications of CABA. The concept of continuous authentication based on BioAura can be used to protect (i) personal computing devices and servers, (ii) software applications, and (iii) restricted physical spaces. Next, we conceptually describe how CABA can be used to protect each domain.

Computing devices, e.g., personal computers, laptops, tablets, and cell phones, or servers can employ two different approaches to utilize CABA: (i) they can use their own computing resources to implement a stand-alone version of CABA, or (ii) they can simply use decisions made by a version of the scheme implemented on a trusted server. We investigate both approaches.

Example 1: Suppose a tablet wants to authenticate its user. The tablet may be unable to dedicate its limited memory/energy resources to support the whole authentication process. In such a scenario, it can use decisions made by a trusted server running CABA. Fig. 3 illustrates this scenario. When the user tries to unlock the tablet, it informs the user's smartphone. The smartphone asks the trusted server to open a secure communication channel. The smartphone then sends the information required for specifying the device that needs to be unlocked, e.g., the tablet ID, along with the information that needs to be processed to authenticate the user, e.g., the user ID and a preprocessed frame of data points from his BioAura, to the trusted server. The server then authenticates the user and sends this decision to the tablet. After initial login, the trusted server demands fresh data points at certain intervals.

Example 2: Suppose the user wants to login to his personal computer. In this case, the computer has enough computational power and energy capacity to implement a stand-alone version of CABA. This case is similar to the one in Example 1, except that there is no need for a trusted server (Fig. 4).

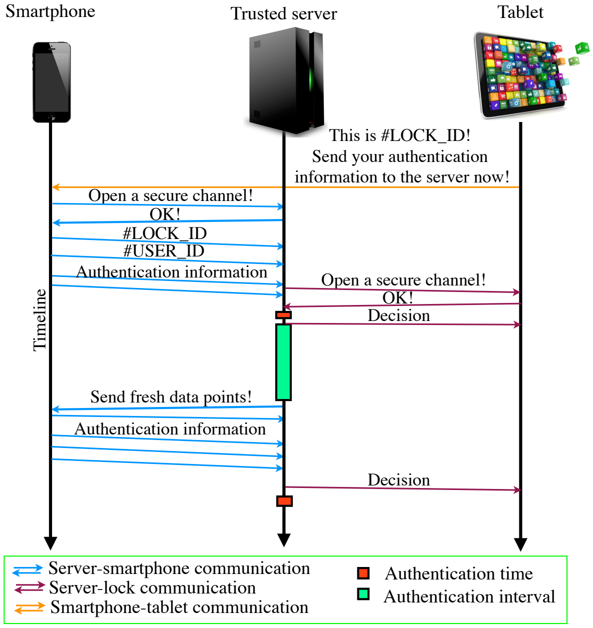


Fig. 3. The tablet wants to authenticate the user. The vertical arrows depict the timeline.

Similarly, CABA has the potential to provide continuous authentication for applications that need strong authentication, e.g., e-commerce applications. Its authentication decisions can be made on the same device that runs the application or on a

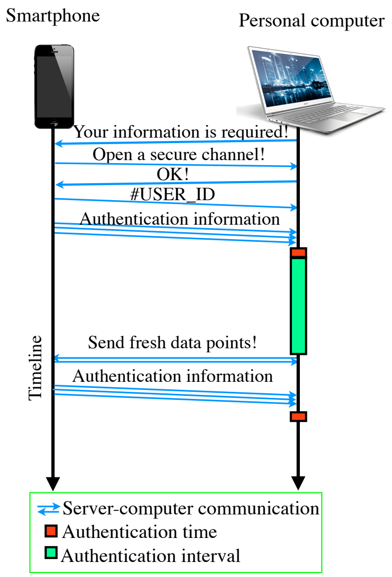

Smartphone
Personal computer
Your information is required!
Open a secure channel!
OK!
#USER_ID
Authentication information
Timeline
Send fresh data points!
Authentication information
⇄ Server-computer communication
🟥 Authentication time
🟩 Authentication interval

Fig. 4. The laptop wants to authenticate the user before allowing the user to utilize its resources or software applications.

powerful trusted server and then transmitted to the device that runs the application.

Example 3: Consider an online banking application that is installed on the user's smartphone. When the user opens the application to access his bank account, the smartphone opens a secure communication channel with the trusted server. Then, the smartphone sends the required information for specifying the application, e.g., the application ID, along with information needed for authenticating the user. The rest of the protocol is the same as before.

Finally, CABA can be used to control access to restricted physical spaces, e.g., buildings. Typically, the electronic device that controls the entrance, e.g., a smart lock, would not have enough computation power to use a stand-alone version of CABA. Hence, in such cases, the scheme can be implemented on a trusted server, and decisions then transmitted to the device. This case is similar to the one depicted in Fig. 3, with the tablet replaced by the lock.

V. Implementation and Experimental Setup

In this section, we first describe our implementation of CABA. We then discuss our experimental setup and different metrics that we used to investigate the proposed system.

A. Prototype implementation

Similar to other authentication systems, CABA has two operating phases: (i) enrollment phase in which CABA builds machine learning-based models for each user, given the training data, and (ii) user authentication phase in which the system continuously authenticates the user. The description of the two phases is presented next.

1) Enrollment phase: In the enrollment phase, the authentication system is given a training dataset. The system builds the model using a supervised learning approach, i.e., a machine learning approach in which the model is built based on labeled training data points.

Generally, the amount of information needed to build a model varies from one application to another. As we elaborate later in Section VI, we evaluated the number of training data points needed to investigate how much information should be sent to the authentication system to build a reliable and accurate model. Each data point in the training set is nine-dimensional and consists of the average values of successive measurements of a Biostream over a one-minute timeframe. The value of each dimension is represented using half-precision floating-point format that requires two bytes of storage. Therefore, if the smartphone needs to transmit data points extracted over a one-hour period, it only needs to send 1080 bytes of data to the authentication system over this period.

In order to maintain reliability, CABA should train a new model based on fresh biomedical data obtained at certain intervals. In other words, CABA should update the model regularly to ensure that the model maintains accuracy and can distinguish legitimate users from impostors. The frequency of model update, i.e., how frequently CABA should repeat the enrollment phase, depends on several factors, such as required accuracy and learning time. As we show later in Section VI, our experimental results indicate that when CABA re-trains the model every four hours, it achieves the best accuracy and the learning time is only a few minutes. Learning can be done transparently to the user. In other words, CABA can re-train the model while the user continues to be authenticated. For example, suppose the enrollment phase takes five minutes each time and is repeated every four hours, i.e., each model is used for four hours. CABA can start re-training to generate a new model after 3 hours and 55 minutes, and be ready with it after four hours have elapsed.

2) User authentication phase: In this phase, the system makes decisions using the already-trained model. In a continuous authentication scenario, the system should verify the user's identity at certain intervals. The frequency of authentication depends on several factors, such as the required level of security and the amount of information required for one authentication. In our prototype implementation, CABA re-authenticates the user every minute based on a given nine-dimensional data point Y that contains the average values of the chosen Biostreams over a specified time interval. When the user approaches the authentication system and requests authentication, the smartphone performs a simple computation on the already-gathered Biostreams and provides Y. Therefore, unlike most previously-proposed continuous authentication systems, e.g., keyboard/mouse-based systems, that require the user to wait while they collect authentication information, CABA obtains the required information almost *instantaneously* because the information has already been gathered and stored on the smartphone for the purpose of health monitoring.

Fig. 5 illustrates how CABA works when the user requests authentication. In a single verification attempt:

1) The smartphone preprocesses one minute of Biostreams collected from the user's BioAura. Then, it transmits the

preprocessed information (Y) along with user ID to the authentication system.

2) The Look-up stage sends Y to the appropriate classifier in the Jury stage based on the given user ID.
3) The dedicated classifier processes Y and outputs a binary decision (accept or reject).

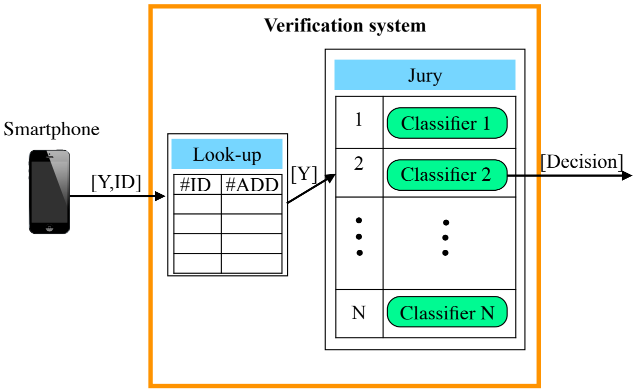


Fig. 5. User authentication phase: The user's smartphone provides Y and the user ID, and CABA outputs the decision.

Next, we provide a detailed description of Look-up and Jury stages.

Look-up stage: This stage forwards the nine-dimensional vector Y provided by the smartphone to the appropriate classifier based on the given user ID. In order to provide a fast search mechanism to find the appropriate classifier, this stage can be implemented using a hash table that associates user IDs with pointers to the classifiers.

Jury stage: The Jury stage consists of N binary classifiers, where N is the number of people who need to be authenticated. The i-th classifier is trained to only accept the data point Y that is extracted by the i-th user's smartphone from his BioAura. The training set of the i-th classifier consists of the i-th user's data points labeled as "accept" and others' data points labeled as "reject".

We have used two well-known binary classification methods: Support Vector Machine (SVM) [26] and Adaptive Boosting (AdaBoost) [27]. Next, we briefly describe each method.

- SVM: The basic concept in an SVM is to find a hyperplane that separates the n-dimensional data into two classes. However, since the data points in the dataset are not usually linearly separable, SVMs introduce the concept of kernel trick that projects the points into a higher-dimensional space, where they are linearly separable. When no prior knowledge about the dataset is available, SVMs usually demonstrate promising results and generalize well. A great number of previous research studies that perform authentication using machine learning methods only consider SVM with a linear kernel [28] or radial basis function (RBF) kernel [29]. In our paper, we decided to investigate both.
- AdaBoost: Although SVM has been commonly used in previously-proposed continuous authentication systems in a variety of scenarios, we decided to include AdaBoost as well. The idea behind AdaBoost is to build a highly accurate classifier by combining many weak classifiers that always perform a little bit better than random guessing on every distribution over the training set [27]. Since biomedical signals are individually slightly discriminative, they lead to weak classifiers, which can be collectively turned into a strong classifier using AdaBoost. Choosing appropriate types of weak classifiers is a significant consideration in AdaBoost. The most commonly used weak learning methods for implementing AdaBoost-based classifiers are decision stumps (also called one-node tree) and decision trees.

B. Experimental Setup and Metrics

Here, we first describe the parameters and dataset used in our experiments. Then, we discuss the accuracy and scalability metrics used to investigate the proposed authentication system.

1) Experimental parameters and dataset: Next, we discuss the parameters used in our experimental setup and describe the dataset.

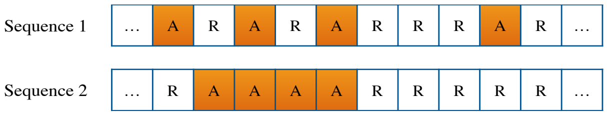
Parameters: We need the following five parameters in our experiments.

- Dataset length (L): This is the duration of Biostreams measurements, i.e., the number of hours of information we have for each person in our data set. In our experiments, we used 14 hours of data for each individual, i.e., $L = 14h$.
- Dataset dimension (n): This is the number of Biostreams that form the BioAura of an individual. In our setup, we have included nine Biostreams for each person, i.e., $n = 9$.
- Dataset size (N): This is the number of people in the dataset. For our experiments, we could only find 37 users ($N = 37$) in our dataset for whom the data: (i) include the nine targeted Biostreams, and (ii) are available over several hours with minimal missing values (we excluded a user for whom the data were not available for more than two consecutive hours).
- Training window size (TRW): This represents the duration of the signal measurements (expressed in hours) for each individual that we used for training the model in the enrollment phase. For example, if TRW is 1 hour, it means we have included 60 data points in our training set, where each point is a nine-dimensional vector consisting of the average values of successive measurements of the nine Biostreams over a one-minute timeframe. We vary TRW in our experiments to study its impact on the model's accuracy.
- Testing window size (TEW): This represents the duration of signal measurements (expressed in hours) for each individual for investigating the accuracy of the trained model in the user authentication phase. We vary the value of TEW in our experiments to investigate its impact on the performance of CABA.

Dataset: In order to investigate the accuracy of CABA, we used a freely available multi-parameter dataset, called MIMIC-II [30]. MIMIC-II was gathered in a controlled environment in which each user remains almost stationary during data collection. It has been extensively used in the medical and biomedical fields. It consists of several anonymized high-resolution vital sign trends, waveforms, and sampled biomedical signals for many individuals. We chose the 37 medical records in MIMIC-II that provide values for all of the required Biostreams for at least 14 hours. Biostreams were sampled using patient monitors (Component Monitoring System Intellivue MP-70 and Philips Healthcare) at the sampling rate of 125 Hz [30].

2) Accuracy metrics: Next, we describe five metrics that we used for analyzing the accuracy of the proposed authentication system. The first three are traditionally used for examining authentication systems. We define two more to investigate the accuracy in the context of continuous authentication.

- False acceptance rate (FAR): This is the ratio of falsely accepted unauthorized users to the total number of invalid requests made by impostors trying to access the system. In the context of continuous authentication, we use the notation $FAR_{t=TEW}$ to denote FAR under TEW. A lower FAR is preferred in cases in which security is very important [31].
- False rejection rate (FRR): This refers to the ratio of falsely rejected requests to the total number of valid requests made by legitimate users trying to access the system. We use the notation $FRR_{t=TEW}$ to denote FRR under TEW. A lower FRR is preferred for user convenience [31].
- Equal error rate (EER): This is the point where FAR equals FRR. Reporting only FRR or FAR does not provide the complete picture because there is a trade-off between the two since we can make one of them low by letting the other one become high. Therefore, we use EER (instead of FRR or FAR) for reporting CABA's accuracy. As before, we use the notation $EER_{t=TEW}$ to denote EER under TEW.
- False acceptance worst-case interval (FAW): The output of the authentication system in a time period T is a sequence of accept/reject decisions. As an example, Fig. 6 shows two possible output sequences over a ten-minute authentication timeframe when an impostor is trying to get authenticated. In both sequences, the number of falsely accepted requests is the same. However, in a continuous authentication scenario, the second sequence would be considered worse since the impostor can use the system over a four-minute timeframe without being detected, whereas in the first case the impostor can only use the system over a one-minute timeframe. We define FAW as the longest time interval (expressed in minutes for CABA) over which an impostor can be falsely accepted as a legitimate user. In the example of Fig. 6, FAW is one minute and four minutes for the first and second cases, respectively.
- False rejection worst-case interval (FRW): Analogously to FAW, we define FRW as the longest time interval (expressed in minutes) over which a legitimate user might be falsely rejected and marked as an impostor.



Sequence 1	...	A	R	A	R	A	R	R	R	A	R	...
Sequence 2	...	R	A	A	A	A	R	R	R	R	R	...

Fig. 6. Two possible output sequences over a ten-minute authentication timeframe. A (R) refers to an accept (reject) decision.

3) Scalability metrics: As mentioned in Section II, the time and space complexity of the authentication system should increase modestly with an increase in the number of users. In order to investigate the scalability of the proposed method, we express the time and space complexities of CABA using the well-known O notation, as a function of N (number of the people in the dataset).

VI. Authentication Results

In this section, we investigate CABA from both the accuracy and scalability perspectives.

A. Authentication accuracy

In order to investigate the accuracy of the authentication system, we implemented a prototype of CABA in MATLAB.

The accuracy of a model is generally investigated using a set of data points that is different from the set used in constructing the model. Thus, in order to train and test a model, the dataset can be divided into two parts: training and test sets. The classical K-fold cross-validation is not suitable for estimating the performance of a system that processes a time series, i.e., a sequence of data points consisting of successive measurements, because potential local dependencies across observations in a time series define a structure in the data that will be ignored by cross-validation [32]. Thus, in this work, instead of using traditional cross-validation, we designed several experimental scenarios for evaluating the accuracy of the authentication system. We describe these scenarios next.

1) *Baseline:* In the baseline scenario, we break the available dataset into two equal parts, i.e., $TEW = TRW = 7h$. We use the first half of the dataset (the first seven hours) of each individual to train the model and the second half to test it. We use all the Biostreams, i.e., $n = 9$, to train and test our system. We use two classification methods: SVM and AdaBoost. In the case of SVM, we use two kernels (linear and RBF). In the case of AdaBoost, we consider decision stumps (one-node tree) and decision trees with 5, 10, 15, and 20 nodes as weak classifiers. We run 40 iterations for all Adaboost-based classifiers since we determined experimentally that the training error becomes zero within these many iterations and testing error becomes minimum. The value of $EER_{t=7h}$ is reported in Table II for all classifiers. AdaBoost with a tree size of 15, i.e., with 15 nodes in the tree, has the minimum value of $EER_{t=7h}$. Increasing tree size usually improves the accuracy of Adaboost-based classifiers. However, using larger trees leads to more complex models, which are more susceptible to overfitting [33]. This can be seen when we move from a tree size of 15 to 20.

TABLE II
CLASSIFIERS AND THEIR $EER_{t=7h}$

Type of classifier	Specification	$EER_{t=7h}$ (%)
SVM	Kernel = Linear	3.0
	Kernel = RBF	2.6
AdaBoost	Tree size = 1	3.1
	Tree size = 5	2.9
	Tree size = 10	2.9
	Tree size = 15	2.4
	Tree size = 20	2.5

Table III summarizes FAW and FRW for all classification schemes. Consider RBF SVM as an example. Its FAW is 4 minutes, which suggests that, in the worst case, an impostor can deceive the authentication system for a 4-minute timeframe. Its FRW is 3 minutes, which suggests that, in the worst case, a legitimate user is falsely rejected for a stretch of 3 minutes.

TABLE III
CLASSIFIERS AND THEIR FAW AND FRW

Type of classifier	Specification	$FAW\ (min)$	$FRW\ (min)$
SVM	Kernel = Linear	4	3
	Kernel = RBF	4	3
AdaBoost	Tree size = 1	5	3
	Tree size = 5	4	3
	Tree size = 10	4	3
	Tree size = 15	4	3
	Tree size = 20	4	4

2) *Biased FAR_t/FRR_t:* Even though it is easier to compare authentication methods based on their EER_t, we may want to minimize FAR_t in highly-secure environments in order to ensure that an impostor is not authorized or minimize FRR_t to enhance user convenience. A low FAR_t indicates a high security level and a low FRR_t ensures user convenience. In this experimental scenario, we use the same parameters that are used in the baseline. However, false acceptance and false rejection are penalized differently. We consider two cases: (i) try to make FAR_t close to zero ($FAR_{t=7h} < 0.1\%$) and measure FRR_t, and (ii) try to make FRR_t close to zero ($FRR_{t=7h} < 0.1\%$) and measure FAR_t. Tables IV and V summarize the results for these two cases. Based on Table IV, CABA can be seen to ensure that impostors are not accepted, but at the cost of an increase in FRR. Based on Table V, CABA can be seen to not negatively impact user convenience, i.e., not falsely reject the user, while rejecting impostors in more than 90% of the cases.

3) *Variable window size:* As mentioned earlier, we set the training and testing window sizes to $7h$ in the baseline. Here, we change the size of the training and testing windows such that $TRW = 2, 3, \cdots, 12h$ and $TEW + TRW = 14h$. Fig. 7 shows the average EER_t for different classifiers with respect to TRW. For all classifiers, as we increase TRW from $2h$ to $6h$, EER_t decreases drastically. Then it remains almost constant until TRW reaches $11h$. Above this TRW, EER starts increasing for two possible reasons. First, the model may become overfitted. Second, the number of test points may be inadequate.

TABLE IV
CLASSIFIERS AND THEIR FRR ($FAR \approx 0$)

Type of classifier	Specification	$FRR(\%)$
SVM	Kernel = Linear	10.2
	Kernel = RBF	9.6
AdaBoost	Tree size = 1	10.0
	Tree size = 5	9.7
	Tree size = 10	8.7
	Tree size = 15	8.4
	Tree size = 20	8.9

TABLE V
CLASSIFIERS AND THEIR FAR ($FRR \approx 0$)

Type of classifier	Specification	$FAR(\%)$
SVM	Kernel = Linear	8.9
	Kernel = RBF	7.6
AdaBoost	Tree size = 1	10.7
	Tree size = 5	9.2
	Tree size = 10	7.8
	Tree size = 15	7.6
	Tree size = 20	8.2

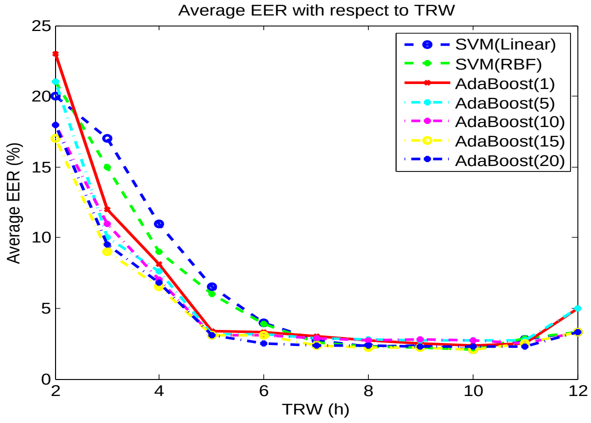


Fig. 7. Average EER_t for different classifiers with respect to TRW.

4) *Moving training window:* In this scenario, the training window moves behind the testing window (Fig. 8). We consider $TEW = TRW = 1, 2, \cdots, 7h$. Our experimental results demonstrate that this verification scheme provides the best result for $TEW = TRW = 4h$, for which the average EER_t is 1.9% and the classification method is AdaBoost with a tree size of 15 nodes. This suggests that we can achieve the best accuracy for $TRW = 4h$, under the assumption that the trained model is valid for the next four hours.

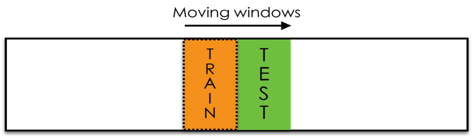


Fig. 8. Moving training window.

5) *Reducing the number of Biostreams:* We also investigate the impact of dropping a Biostream. Traditionally, feature reduction is used to remove redundant or irrelevant features from the data set before commencing on the training process in order to decrease unnecessary computational cost. However, in our scenario, the main purpose of feature reduction is to investigate how each feature affects accuracy. If CABA can provide an acceptable accuracy with fewer features, fewer WMSs would be required. We dropped one feature at a time and computed $EER_{t=7h}$ of the system. All other configurations are kept the same as in the baseline. Fig. 9 illustrates how $EER_{t=7h}$ changes for each of the seven classifiers used in our experiments (two SVM classifiers with different kernel types and five AdaBoost classifiers with different tree sizes) when we drop different Biostreams. The green bar shows the baseline scenario in which no feature is dropped. We can see that dropping the respiratory rate (temperature) has maximum (minimum) negative impact on authentication accuracy. Thus, the most and least important Biostreams are respiratory rate and body temperature, respectively.

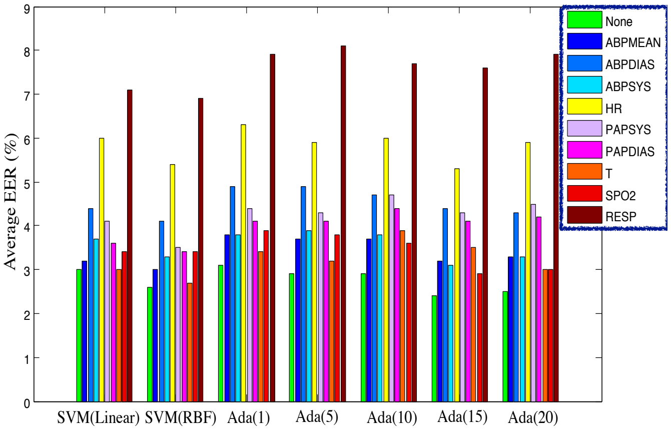


Fig. 9. $EER_{t=7h}$ for different classifiers when Biostreams are dropped one at a time. The green bar depicts the baseline scenario in which no feature is dropped. The abbreviations/notations provided in Table I are used to label other bars.

B. CABA scalability

We discuss below the worst-case time and space complexities of CABA.

1) Time complexity: As discussed earlier, CABA can be implemented in such a manner that the time required by the enrollment phase is hidden from the user's perspective. Hence, we focus on the time complexity of the user authentication process. We found that the required time for processing an authentication request for $N = 37$ was on the order of a few milliseconds for all classification methods, when CABA was implemented on a MacBook Pro (2.3 GHz Intel Core i7 processor with 8 GB memory). This suggests that CABA can re-authenticate the user very quickly.

When a person requests authentication by providing his ID and feature vector Y, the Look-up stage forwards Y to one and only one classifier in the Jury stage based on the given user ID. Then, the classifier's decision is the final decision of the authentication system. Hence, in order to analyze the time complexity of a single user authentication process, we need to consider the time complexity of the Look-up stage, and one classifier in the Jury stage, as follows:

- Look-up stage: If the Look-up stage is implemented using a hash table that associates user IDs with pointers to classifiers, then its search operation (finding the location of the classifier associated with the user ID) can be performed in $O(1)$ time.
- One classifier in the Jury stage: The time complexity of the classifier varies from one classification algorithm to another. The time complexities of AdaBoost classifiers and the SVM classifier with a linear kernel do not depend on N, i.e., they have time complexity of $O(1)$. The time complexity of SVM with an RBF kernel is $O(n_{SV})$, where n_{SV} is the number of support vectors. Theoretically, n_{SV} grows linearly with a linear increase in N. Thus, the SVM classifier with an RBF kernel has a time complexity of $O(N)$.

Hence, the overall time complexity of user authentication is $O(1)$ for AdaBoost classifiers and the SVM classifier with a linear kernel, and $O(N)$ for the SVM classifier with an RBF kernel.

2) Space complexity: We first investigate how much memory is required for our prototype implementation of CABA. Then, we discuss how the amount of memory required to store the two stages (Look-up and Jury) increases with N.

The amount of memory required for storing the Look-up stage in our prototype, where $N = 37$, was less than 1 kB. The amount of memory required for storing a single classifier in the Jury stage varies from tens of bytes (for SVM with a linear kernel) to a few kB (for AdaBoost with a tree size of 20). Therefore, the total amount of memory allocated to the authentication system is less than 1 MB.

We investigate the space complexity as a function of N.

- Look-up stage: If the Look-up stage is implemented using a hash table, its space complexity is $O(N)$.
- Jury stage: The space complexity of a single classifier in the Jury stage depends on the type of classifier. The space complexities of AdaBoost classifiers and the SVM classifier with a linear kernel do not depend on N, i.e., they have space complexity of $O(1)$. However, the space complexity of the SVM with an RBF kernel is $O(N)$. Since the number of classifiers in the Jury stage increases linearly with N, its space complexity is $O(N)$ (when an AdaBoost classifier or SVM classifier with a linear kernel is used) or $O(N^2)$ (when the SVM classifier with the RBF kernel is employed).

VII. Using BioAura for Identification

The majority of continuous authentication systems only support continuous verification in which the user provides a user ID and the system checks if the user is the person he purports to be. In this section, we describe how CABA can be slightly modified to also identify the user from a database of users by processing feature vector Y provided by the smartphone. An identification scenario consists of four steps. The first three steps are similar to the ones discussed in Section V for continuous authentication. In the fourth step, CABA processes the decisions of all classifiers in the Jury stage to indicate that the user is not in the database, or conclude that he is, in which case it returns his user ID. This step can be implemented in different ways. In our implementation, CABA processes all outputs of the Jury stage and outputs the user ID if there is only one classifier whose output is an accept decision. Otherwise, it indicates no match. Our experimental results demonstrate that this scheme provides the best result, for which the identification rate is 96.1%, with the AdaBoost classification method with a tree size of 15 nodes. Identification rate is a commonly used metric for this scenario [31]. It is defined as the percentage of attempts correctly identified to the total number of attempts made.

VIII. Real-time Adaptive Authorization

In this section, we first define the concept of authorization. Then, we propose a real-time adaptive authorization (RAA) scheme, which uses the decisions from CABA to provide an extremely flexible access control model. The RAA concept is not limited to CABA. It provides an adjustable access control model for any authorization system that authorizes the user based on decisions of a continuous authentication system.

Authorization is defined as the process of establishing if the user, who has been already authenticated, should be allowed access to a resource, system, or area [34].

Traditional authorization schemes grant a specific access level to the authenticated user based on his user ID. However, the fact that continuous authentication systems have a non-zero FRR implies that such a simple scheme may unintentionally block a legitimate access when the authentication system fails to recognize a valid user for a short period of time. Consider a scenario in which a continuous authentication system is used to protect a personal laptop from unauthorized users. The authentication system first authenticates the user. Then, the authorization scheme specifies the user's access level based on the user ID. However, the laptop may log out the user when the authentication scheme falsely rejects him. RAA schemes can be used to alleviate user inconvenience caused by false reject decisions. They continuously adjust the user's access level based on the last decision of the authentication system. Next, we propose an RAA scheme that can be used with a continuous authentication system.

A trust level-based RAA adaptively changes the user's access level based on a parameter called trust level (TRL). TRL is a recently-suggested parameter that represents how much we trust a user based on previous decisions of the continuous authentication system [35]. TRL has a value between 0 and 100, where a higher number indicates a higher level of trust. The initial value of TRL is 100 when the user is authenticated and authorized for the first time. The value of TRL is continuously updated using a trust update procedure after each user authentication. A simple trust update procedure may be to just increase (decrease) the TRL by a constant step after each accept (reject) decision. *Trust update procedure* shows the pseudo-code for such an approach. We need to set two parameters: W_{Accept} and W_{Reject}. The values of W_{Accept} and W_{Reject} should be chosen such that the TRL value becomes 0 as soon as we detect the presence of an impostor and becomes 100 when we confidently verify that the user is legitimate. Consider AdaBoost classification with a tree size of 15 nodes that yields $FRW = 3$. This indicates that the authentication system may falsely reject three consecutive requests of a legitimate user in the worst case. Therefore, if the RAA scheme gets at least four consecutive reject decisions from the authentication system, it becomes confident that the user is an impostor ($TRL = 0$). Hence, we can set W_{Reject} for this classifier as follows: $W_{Reject} = \frac{-100}{FRW+1} = \frac{-100}{4} = -25$. $FAW = 4$ for the above-mentioned classification method, which indicates that in the worst case, an impostor may be falsely accepted as a legitimate user in four successive trials. Therefore, if the authentication system outputs five consecutive accept decisions, TRL should become 100. Thus, we can set W_{Accept} as follows: $W_{Accept} = \frac{+100}{FAW+1} = \frac{+100}{5} = +20$.

We can set different threshold values for different applications. We set the threshold value to 100 for accessing email and financial accounts to ensure that the user can access such accounts only when the system is confident that the user is legitimate. However, for less sensitive applications, e.g., simple web surfing, a lower level of trust might be sufficient. Using CABA in conjunction with RAA can enhance user convenience, while ensuring high security for critical applications.

Trust update procedure

```
Given: The latest $decision$ of the authentication system and current TRL value

$TRL \leftarrow TRL + F_{Update}$, where
    $F_{Update}(decision) = \begin{cases} W_{Accept}, & \text{if } decision = Accept, \\ W_{Reject}, & \text{otherwise.} \end{cases}$
If $(TRL > 100)$
    $TRL \leftarrow 100$
end
If $(TRL < 0)$
    $TRL \leftarrow 0$
end
Return $TRL$

Output: $TRL$
```

IX. Potential Threats and Countermeasures

Next, we describe possible attacks/threats against CABA that can be exploited by attackers to bypass CABA. For each attack, we also suggest possible countermeasures.

1. Eavesdropping: This is defined as the act of covertly listening to confidential conversations of others [36], which, in our context, can be done by intercepting the communication between two devices using an appropriate equipment, e.g., HackRF [37]. Eavesdropping can occur when unencrypted information is transmitted over an untrusted channel.

Countermeasures: The most effective and well-known defense against eavesdropping is encryption. For example, the transmitted message can be encrypted using Advanced Encryption Standard [38]. However, implementing a strong encryption in WMSs may not be possible in the current state of the technology since they have limited energy and memory capacity. Fortunately, eavesdropping does not pose a direct threat to the authentication system. In other words, it is possible to design the authentication system assuming that eavesdropping does occur on the communication between the WMSs and the smartphone. In this case, CABA would require that the data be sent from a smartphone that is previously registered in the system to ensure that the attacker is not able to capture the biomedical information and send the captured information to CABA using another smartphone. The smartphone can send its unique ID over a secure communication link to CABA before transmitting the biomedical information.

2. Phishing: This is an attack that attempts to fool the user into submitting his confidential or private information, e.g., username, password, email address, and phone number, to an untrusted server or device [39]. For example, the attacker might attempt to fool the user's smartphone by sending a counterfeit request that asks the smartphone to send its authentication-related information to the attacker's server.

Countermeasures: The most effective way to address phishing attacks is to use a digital certificate, i.e., an electronic document that allows a device to exchange information securely using the public key infrastructure [40]. The certificate carries information about the key and its owner. In CABA, the server's digital certificate can be examined by the smartphone to ensure that the server is trusted.

3. Replay attack: In a replay attack, an attacker records the data, packets, and user's credentials, which are transmitted between two devices, e.g., a WMS and the smartphone, and exploits them for a malicious purpose. In a replay attack against the authentication system, the attacker attempts to impersonate a legitimate user in order to bypass the authentication procedure and gain full access to the protected device, application, or area. Unlike the attacks based on eavesdropping, in a replay attack, the attacker does not need to interpret the packets. In fact, he can even record encrypted packets and retransmit them in order to bypass the system.

Countermeasures: An encrypted timestamp, i.e., a sequence of encrypted information identifying when the transmission occurred, can be utilized to enable the authentication system to check that the packets were not previously recorded. Moreover, the packet should include a field that contains the encrypted information, e.g., a hashed device ID, which can be used in the authentication system to uniquely specify the sender of the packets and check if the sender is known and trusted.

4. Poisoning attack: In a poisoning attack, the attacker changes the final learning model by adding precisely-selected invalid data points to the training dataset [41]. In CABA, the attacker might threaten the integrity of the machine learning algorithm by using an untrusted WMS that aims to add malicious data points to the training set.

Countermeasures: We describe two types of countermeasures against poisoning attacks.

1. Outlier detection: One of the common goals of defenses against poisoning attacks is to reduce the effect of invalid data points on the final result. In a machine learning method, such invalid data points are considered outliers in the training dataset. Several countermeasures against poisoning attacks have been discussed in [42].

2. Digitally-signed biomedical information: A digital signature can be used to check the authenticity of the information. It is a mathematical method for demonstrating the authenticity of a transmitted message. Thus, it provides the means to the recipient to check if the message is created by a legitimate sender. The WMSs and the smartphone can digitally sign the biomedical information before transmitting it.

X. Comparison between CABA and previously-proposed systems

In this section, we first describe why previously-proposed authentication systems based on biomedical signals (EEG and ECG) may not be well-suited to continuous authentication. Then, we compare CABA to three promising biometrics-/behaviometrics-based continuous authentication systems.

The use of EEG [44] and ECG [45], [46] signals, as biomedical traits with high discriminatory power for user authentication, has received widespread attention in recent years. Although such authentication systems show promising results, they do not provide a convenient method for long-term continuous user authentication for two reasons. First, they commonly need long measurement times and impose a heavy computational load on the system [47]. Second, due to the size/position requirements of the electrodes that enable EEG/ECG acquisition [23], [45], these systems can mainly be used for one-time user authentication (or short-term continuous authentication) systems. For example, the user needs to wear a large cap to collect the data for EEG-based authentication [44], which is not convenient for long-term continuous authentication.

As mentioned in Section I, several biometrics-/behaviometrics-based continuous authentication systems have been proposed. Among them, facial recognition systems [6], [13], [48], [49], which use facial features (as biometrics), and keyboard-/mouse-based authentication systems [1], [8], [16], [50]–[52], which rely on keystroke/mouse dynamics (as behaviometrics), are the most promising.

Facial recognition systems make use of low-cost cameras that are commonly built into most laptops. They are accurate when the user looks straight at the webcam. However, their performance is significantly affected by illumination, pose, expression or changes in the image acquisition method [13]. Moreover, the user's facial images is unavailable when the user turns his head or does not look at the camera. Such systems are also not useful for tablets and smartphones since the user typically does not face a built-in camera in these cases.

Previous keyboard-/mouse-based authentication systems report promising results and provide user authentication in a convenient manner. However, they have four drawbacks that

TABLE VI
COMPARISON OF DIFFERENT CONTINUOUS AUTHENTICATION SYSTEMS

System	Passiveness	Availability	Accuracy	Scalability	Efficiency	Low cost	Stability	Extensibility	EER
Keyboard-based [31]	+	-	+	+	+	+	+	-	0.5% to 17.6%
Mouse-based [8]	+	-	+	+	+	+	+	-	2.5% to 26.8%
Facial recognition [43]	+	-	+	+	+	+	+	-	2.4% to 20.0%
CABA	+	+	+	+	+	+	+	+	1.9%

limit their applicability: (i) their performance is easily impacted by environmental variables, such as changes in software environments, input devices, task, or interaction modes [8], [31], (ii) they can only be employed when system has a keyboard/mouse, (iii) the data often become unavailable, e.g., when the user is watching a movie on his computer, and (iv) keyboard-based systems need active involvement of the user for long sessions, e.g., several minutes [16], to guarantee acceptable accuracy, and mouse dynamics based systems have still not reached an acceptable accuracy levels [8].

Unlike most continuous authentication systems that support personal computers and laptops, CABA can be used to protect personal computers, servers, software applications, and restricted physical spaces. Moreover, WMSs ensure a continuous data stream. This enables the user to freely move and change his posture while being authenticated. In addition, unlike previous systems, CABA can be implemented on any general-purpose computing unit with sufficient memory capacity and computation power. Table VI compares CABA to continuous keyboard-based, mouse-based, and facial recognition systems.

XI. DISCUSSION

Here, we address three items not yet explained in detail. First, we discuss an important privacy concern surrounding the use of biomedical signals. Second, we describe how CABA can also be used as a stand-alone one-time authentication system. Third, we discuss the impact of temporal conditions on authentication results.

A. Health information leakage

An important privacy concern associated with the use of biomedical signals is the possibility of health information leakage. For example, an adversary might extract disease-specific information from such signals, e.g., certain heart rate ranges may be correlated with cardiovascular disease [53]. Exposure of a serious illness or a condition that carries social stigma would naturally raise serious privacy concerns [54]. However, since CABA does not rely on high-precision measurements (it only processes the average values of Biostreams over specific time frames), the amount of health-related information potentially leaked by CABA is less than leaked by EEG/ECG-based approaches that rely on high-quality EEG/ECG signals. Similar concerns have been discussed in previous research efforts for both biometrics and behaviometrics, and usually addressed by suggesting legislation [55].

B. One-time authentication based on BioAura

CABA can also be used as a stand-alone one-time authentication system. We discuss several such scenarios next.

- Battery-powered devices: Incurring overheads of continuous authentication on a battery-powered device may drain its battery quickly, and lead to user inconvenience.
- Low-security environment: Continuous authentication may not be required in a low-security environment, e.g., a common room in an apartment.
- Intentionally-shared resources: A user might want to intentionally authorize a group of users to access some specific locations or resources. For example, consider a user who uses a smart lock, which grants access to him when he approaches the door of his house. He may want to open the door for his guests and leave the house.

Generally, a continuous authentication system that has high accuracy and a short response time may be able to provide stand-alone one-time authentication or complement a traditional authentication system (whose decision is only considered at the time of initial login). As discussed in Section VI, CABA provides an accurate decision within a few milliseconds and, hence, is also useful for one-time authentication.

C. The impact of temporal conditions

The negative impact of temporal conditions, e.g., emotional/physical conditions and changes in posture, gesture or facial expressions, on widely-used biometrics-/behaviometrics-based systems have been discussed earlier [56]. Similarly, some biomedical signals may change significantly due to a change in physical activity. This may negatively impact authentication accuracy. For example, when the user suddenly starts running, his blood pressure, heart rate, and respiration rate increase. Therefore, if the authentication system has only been trained using data collected when the user was at rest, it might fail to authenticate the user after he finishes running. A solution would be to design a state-aware system that takes different emotional states and physical activities into account. Algorithms exist for recognizing emotional states [57] and the type [58] and intensity [59] of physical activities using WMSs. Such algorithms can be used in conjunction with CABA.

XII. CONCLUSION

In this paper, we proposed CABA, a novel user-transparent system for continuous authentication based on information that is already gathered by WMSs for diagnostic and therapeutic purposes. We described a prototype implementation of CABA and comprehensively investigated its accuracy and scalability.

We also described how CABA can be used to support user identification. We then presented an RAA scheme that uses the decisions from CABA to enable flexible access control. We compared CABA to previously-proposed continuous authentication systems (biometrics- and behaviometrics-based), and highlighted its advantages. We discussed several attacks against the proposed authentication system along with their countermeasures. Finally, we briefly described an privacy concerns surrounding the use of biomedical signals, how CABA can also be used for one-time authentication, and impact of temporal conditions on authentication.

References

[1] P. S. Teh, A. B. J. Teoh, and S. Yue, "A survey of keystroke dynamics biometrics," *The Scientific World Journal*, vol. 2013, 2013.

[2] T. Sim, S. Zhang, R. Janakiraman, and S. Kumar, "Continuous verification using multimodal biometrics," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 687–700, 2007.

[3] J. Bonneau, C. Herley, P. C. Van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in *Proc. IEEE Symp. Security and Privacy*, 2012, pp. 553–567.

[4] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith, "Smudge attacks on smartphone touch screens," in *Proc. USENIX Wkshp. Offensive Technologies*, vol. 10, 2010, pp. 1–7.

[5] C. Ma, D. Wang, and S. Zhao, "Security flaws in two improved remote user authentication schemes using smart cards," *Int. J. Communication Systems*, vol. 27, no. 10, pp. 2215–2227, 2014.

[6] K. Niinuma, U. Park, and A. K. Jain, "Soft biometric traits for continuous user authentication," *IEEE Trans. Information Forensics and Security*, vol. 5, no. 4, pp. 771–780, 2010.

[7] S. Egelman, S. Jain, R. S. Portnoff, K. Liao, S. Consolvo, and D. Wagner, "Are you ready to lock?" in *Proc. ACM Conf. Computer and Communications Security*, 2014, pp. 750–761.

[8] C. Shen, Z. Cai, and X. Guan, "Continuous authentication for mouse dynamics: A pattern-growth approach," in *Proc. IEEE Int. Conf. Dependable Systems and Networks*, 2012, pp. 1–12.

[9] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 40, no. 1, pp. 1–12, 2010.

[10] R. Gravina, P. Alinia, H. Ghasemzadeh, and G. Fortino, "Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges," *Information Fusion*, vol. 35, pp. 68–80, 2017.

[11] "Growth trends, consumer attitudes, and why smartwatches will dominate," http://www.businessinsider.com/the-wearable-computing-market-report-2014-10, accessed: 08-1-2015.

[12] M. Mihajlov and B. Jerman-Blažič, "On designing usable and secure recognition-based graphical authentication mechanisms," *Interacting with Computers*, vol. 23, no. 6, pp. 582–593, 2011.

[13] K. Niinuma and A. K. Jain, "Continuous user authentication using temporal information," in *Proc. SPIE Defense, Security, and Sensing*, 2010, p. 76670L.

[14] S. Liu and M. Silverman, "A practical guide to biometric security technology," *IEEE IT Professional*, vol. 3, no. 1, pp. 27–32, 2001.

[15] W. Shi, J. Yang, Y. Jiang, F. Yang, and Y. Xiong, "Senguard: Passive user identification on smartphones using multiple sensors," in *Proc. IEEE Int. Conf. Wireless and Mobile Computing, Networking and Communications*, 2011, pp. 141–148.

[16] J. Roth, X. Liu, and D. Metaxas, "On continuous user authentication via typing behavior," *IEEE Trans. Image Processing*, vol. 23, no. 10, pp. 4611–4624, 2014.

[17] N. Bartlow and B. Cukic, "Evaluating the reliability of credential hardening through keystroke dynamics," in *Proc. Int. Symp. Software Reliability Engineering*, vol. 6, 2006, pp. 117–126.

[18] E. Alsolami, "An examination of keystroke dynamics for continuous user authentication," Ph.D. dissertation, Queensland University of Technology, 2012.

[19] M. Schuckers, "Some statistical aspects of biometric identification device performance," *Stats Magazine*, vol. 3, 2001.

[20] A. M. Nia, M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, "Energy-efficient long-term continuous personal health monitoring," *IEEE Trans. Multi-Scale Computing Systems*, vol. 1, no. 2, pp. 85–98, 2015.

[21] A. Bourouis, M. Feham, and A. Bouchachia, "Ubiquitous mobile health monitoring system for elderly (UMHMSE)," *arXiv preprint arXiv:1107.3695*, 2011.

[22] U. Varshney, "Pervasive healthcare and wireless health monitoring," *Mobile Networks and Applications*, vol. 12, no. 2-3, pp. 113–127, 2007.

[23] M. Teplan, "Fundamentals of EEG measurement," *Measurement Science Review*, vol. 2, no. 2, pp. 1–11, 2002.

[24] E. B. Mazomenos, D. Biswas, A. Acharyya, T. Chen, K. Maharatna, J. Rosengarten, J. Morgan, and N. Curzen, "A low-complexity ECG feature extraction algorithm for mobile healthcare applications," *IEEE J. Biomedical and Health Informatics*, vol. 17, no. 2, pp. 459–469, 2013.

[25] P. R. Rijnbeek, J. A. Kors, and M. Witsenburg, "Minimum bandwidth requirements for recording of pediatric electrocardiograms," *Circulation*, vol. 104, no. 25, pp. 3087–3090, 2001.

[26] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[27] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *J. Japanese Society For Artificial Intelligence*, vol. 14, pp. 771–780, 1999.

[28] H. Gascon, S. Uellenbeck, C. Wolf, and K. Rieck, "Continuous authentication on mobile devices by analysis of typing motion behavior," in *Proc. Sicherheit*, 2014, pp. 1–12.

[29] Z. Sitová, J. Šeděnka, Q. Yang, G. Peng, G. Zhou, P. Gasti, and K. S. Balagani, "HMOG: New behavioral biometric features for continuous authentication of smartphone users," *IEEE Trans. Information Forensics and Security*, vol. 11, no. 5, pp. 877–892, 2016.

[30] M. Saeed, M. Villarroel, A. T. Reisner, G. Clifford, L.-W. Lehman, G. Moody, T. Heldt, T. H. Kyaw, B. Moody, and R. G. Mark, "Multiparameter intelligent monitoring in intensive care II (MIMIC-II): A public-access intensive care unit database," *Critical Care Medicine*, vol. 39, no. 5, p. 952, 2011.

[31] S. P. Banerjee and D. L. Woodard, "Biometric authentication and identification using keystroke dynamics: A survey," *J. Pattern Recognition Research*, vol. 7, no. 1, pp. 116–139, 2012.

[32] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010.

[33] D. Mease and A. Wyner, "Evidence contrary to the statistical view of boosting," *J. Machine Learning Research*, vol. 9, pp. 131–156, 2008.

[34] S. P. Miller, B. C. Neuman, J. I. Schiller, and J. H. Saltzer, "Kerberos authentication and authorization system," in *Proc. Project Athena Technical Plan*, 1987.

[35] I. Deutschmann, P. Nordstrom, and L. Nilsson, "Continuous authentication using behavioral biometrics," *IEEE IT Professional*, vol. 15, no. 4, pp. 12–15, 2013.

[36] A. Mosenia and N. K. Jha, "A comprehensive study of security of Internet of Things," *IEEE Trans. Emerging Topics in Computing*, 2016.

[37] "HackRF One," https://greatscottgadgets.com/hackrf/, accessed: 10-1-2015.

[38] J. Daemen and V. Rijmen, *The Design of Rijndael: AES-The Advanced Encryption Standard*. Springer Science and Business Media, 2013.

[39] C. Abad, "The economy of phishing: A survey of the operations of the phishing market," *First Monday*, vol. 10, no. 9, 2005.

[40] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in *Proc. ACM Conf. Human Factors In Computing Systems*, 2006, pp. 581–590.

[41] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," *arXiv preprint arXiv:1206.6389*, 2012.

[42] M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, "Systematic poisoning attacks on and defenses for machine learning in healthcare," *IEEE J. Biomedical and Health Informatics*, vol. 19, no. 6, pp. 1893–1905, 2015.

[43] D. V. Jadhav, P. K. Ajmera, and N. S. Nehe, "Real time human face location and recognition system using single training image per person," in *Proc. Annual IEEE India Conference*, 2011, pp. 1–5.

[44] J. Sohankar, K. Sadeghi, A. Banerjee, and S. K. Gupta, "E-bias: A pervasive EEG-based identification and authentication system," in *Proc. 11th ACM Symp. QoS and Security for Wireless and Mobile Networks*, 2015, pp. 165–172.

[45] Z. Zhao, L. Yang, D. Chen, and Y. Luo, "A human ECG identification system based on ensemble empirical mode decomposition," *Sensors*, vol. 13, no. 5, pp. 6832–6864, 2013.

[46] R. D. Labati, R. Sassi, and F. Scotti, "ECG biometric recognition: Permanence analysis of QRS signals for 24 hours continuous authentication," in *Proc. IEEE Int. Wkshp. Information Forensics and Security*, 2013, pp. 31–36.

[47] I. Nakanishi, S. Baba, and C. Miyamoto, "EEG based biometric authentication using new spectral features," in *Proc. IEEE Int. Symp. Intelligent Signal Processing and Communication Systems*, 2009, pp. 651–654.

[48] D. Crouse, H. Han, D. Chandra, B. Barbello, and A. K. Jain, “Continuous authentication of mobile user: Fusion of face image and inertial measurement unit data,” in *Proc. IEEE Int. Conf. Biometrics*, 2015, pp. 135–142.

[49] P.-W. Tsai, M. K. Khan, J.-S. Pan, and B.-Y. Liao, “Interactive artificial bee colony supported passive continuous authentication system,” *IEEE Systems Journal*, vol. 8, no. 2, pp. 395–405, 2014.

[50] S. Mondal and P. Bours, “Context independent continuous authentication using behavioural biometrics,” in *Proc. IEEE Int. Conf. Identity, Security and Behavior Analysis*, 2015, pp. 1–8.

[51] I. Traore, I. Woungang, Y. Nakkabi, M. S. Obaidat, A. A. E. Ahmed, and B. Khalilian, “Dynamic sample size detection in learning command line sequence for continuous authentication,” *IEEE Trans. Systems, Man, and Cybernetics*, vol. 42, no. 5, pp. 1343–1356, 2012.

[52] P. Bours and S. Mondal, “Performance evaluation of continuous authentication systems,” *IET Biometrics*, vol. 4, no. 4, pp. 220–226, 2015.

[53] K. Fox, J. S. Borer, A. J. Camm, N. Danchin, R. Ferrari, J. L. L. Sendon, P. G. Steg, J.-C. Tardif, L. Tavazzi, and M. Tendera, “Resting heart rate in cardiovascular disease,” *J. American College of Cardiology*, vol. 50, no. 9, pp. 823–830, 2007.

[54] A. Nia, S. Sur-Kolay, A. Raghunathan, and N. Jha, “Physiological information leakage: A new frontier in health information security,” *IEEE Trans. Emerging Topics in Computing*.

[55] D. R. Carpenter, A. J. McLeod Jr, and J. G. Clark, “Using biometric authentication to improve fire ground accountability: An assessment of firefighter privacy concerns,” *in Proc. Americas Conference on Information Systems*, p. 11, 2008.

[56] K. Delac and M. Grgic, “A survey of biometric recognition methods,” in *Proc. IEEE Int. Symp. Electronics in Marine*, 2004, pp. 184–193.

[57] C. Peter, E. Ebert, and H. Beikirch, “A wearable multi-sensor system for mobile acquisition of emotion-related physiological data,” in *Proc. Int. Conf. Affective Computing and Intelligent Interaction*, 2005, pp. 691–698.

[58] T. Denning, A. Andrew, R. Chaudhri, C. Hartung, J. Lester, G. Borriello, and G. Duncan, “BALANCE: Towards a usable pervasive wellness application with accurate activity inference,” in *Proc. Wkshp. ACM Mobile Computing Systems and Applications*, 2009, p. 5.

[59] P. Alinia, R. Saeedi, R. Fallahzadeh, A. Rokni, and H. Ghasemzadeh, “A reliable and reconfigurable signal processing framework for estimation of metabolic equivalent of task in wearable sensors,” *IEEE J. Selected Topics in Signal Processing*, vol. 10, no. 5, pp. 842–853, 2016.

Arsalan Mosenia received his B.S. degree in Computer Engineering from Sharif University of Technology, Tehran, Iran, in 2012, and M.A. degree in Electrical Engineering from Princeton, NJ, in 2014. He is currently pursuing a Ph.D. degree in Electrical Engineering at Princeton University, NJ. His research interests include Internet of Things, information security, mobile computing, distributed computing, and machine learning.

Susmita Sur-Kolay (SM’05) received the B.Tech. degree in electronics and electrical communication engineering from Indian Institute of Technology, Kharagpur, India, and the Ph.D. degree in Computer Science and Engineering from Jadavpur University, Kolkata, India. She was in the Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA, from 1980 to 1984. She was a Post-Doctoral Fellow at the University of Nebraska-Lincoln, Nebraska-Lincoln, NE, USA, in 1992, a Reader in Jadavpur University from 1993 to 1999, a Visiting Faculty Member with Intel Corporation, Santa Clara, CA, USA, in 2002, and a Visiting Researcher at Princeton University in 2012. She is a Professor in the Advanced Computing and Microelectronics Unit, Indian Statistical Institute, Kolkata. She has co-edited two books, authored a book chapter in the Handbook of Algorithms for VLSI Physical Design Automation, and co-authored about 100 technical articles. Her current research interests include electronic design automation, hardware security, quantum computing, and graph algorithms. She was a Distinguished Visitor of the IEEE Computer Society, India. She has served as an Associate Editor of the IEEE Transactions on Very Large Scale Integration Systems, and is currently an Associate Editor of ACM Transactions on Embedded Computing Systems.

Anand Raghunathan is a Professor and Chair of VLSI in the School of Electrical and Computer Engineering at Purdue University, where he leads the Integrated Systems Laboratory. He has co-authored a book (“High-level Power Analysis and Optimization”), eight book chapters, 21 U.S patents, and over 200 refereed journal and conference papers. His publications have been recognized with eight best paper awards and four best paper nominations. He received the Patent of the Year Award (recognizing the invention with the highest impact), and two Technology Commercialization Awards from NEC. He was chosen by MIT’s Technology Review to be among TR35 (top 35 innovators under 35 years, across various disciplines of science and technology) in 2006, for his work on “making mobile secure”. He has chaired the ACM/IEEE International Symposium on Low Power Electronics and Design, the ACM/IEEE International Conference on Compilers, Architecture, and Synthesis for Embedded Systems, the IEEE VLSI Test Symposium, and the IEEE International Conference on VLSI Design. He has served as Associate Editor of the IEEE Transactions on CAD, IEEE Transactions on VLSI Systems, ACM Transactions on Design Automation of Electronic Systems, IEEE Transactions on Mobile Computing, ACM Transactions on Embedded Computing Systems, IEEE Design & Test of Computers, and the Journal of Low Power Electronics. He was a recipient of the IEEE Meritorious Service Award (2001) and Outstanding Service Award (2004). He is a Fellow of the IEEE, and Golden Core Member of the IEEE Computer Society. He received the B. Tech. degree in Electrical and Electronics Engineering from the Indian Institute of Technology, Madras, and the M.A. and Ph.D. degrees in Electrical Engineering from Princeton University.

Niraj K. Jha (S’85-M’85-SM’93-F’98) received his B.Tech. degree in Electronics and Electrical Communication Engineering from Indian Institute of Technology, Kharagpur, India in 1981, M.S. degree in Electrical Engineering from S.U.N.Y. at Stony Brook, NY in 1982, and Ph.D. degree in Electrical Engineering from University of Illinois at Urbana-Champaign, IL in 1985. He is a Professor of Electrical Engineering at Princeton University. He is a Fellow of IEEE and ACM. He has served as the Editor-in-Chief of IEEE Transactions on VLSI Systems and an Associate Editor of IEEE Transactions on Circuits and Systems I and II, IEEE Transactions on VLSI Systems, IEEE Transactions on Computer-Aided Design, Journal of Electronic Testing: Theory and Applications, and Journal of Nanotechnology. He is currently serving as an Associate Editor of IEEE Transactions on Computers, IEEE Transactions on Multi-Scale Computing Systems, and Journal of Low Power Electronics. He has also served as the Program Chairman of the 1992 Workshop on Fault-Tolerant Parallel and Distributed Systems, the 2004 International Conference on Embedded and Ubiquitous Computing, and the 2010 International Conference on VLSI Design. He has served as the Director of the Center for Embedded System-on-a-chip Design funded by New Jersey Commission on Science and Technology. He is the recipient of the AT&T Foundation Award and NEC Preceptorship Award for research excellence, NCR Award for teaching excellence, and Princeton University Graduate Mentoring Award. He received the Distinguished Alumnus Award from I.I.T., Kharagpur in 2014. He has co-authored or co-edited five books that include two textbooks: Testing of Digital Systems (Cambridge University Press, 2003) and Switching and Finite Automata Theory, 3rd edition (Cambridge University Press, 2009). He has also co-authored 15 book chapters and more than 430 technical papers. He has coauthored 14 papers that have won various awards, and another six that have received best paper award nominations. He has received 16 U.S. patents. He has served on the program committees of more than 150 conferences and workshops. His research interests include FinFETs, monolithic 3D IC design, low power hardware/software design, computer-aided design of integrated circuits and systems, secure computing, and embedded computing. He has given several keynote speeches in the area of nanoelectronic design and test.