Cognitive Security: Securing the Burgeoning Landscape of Mobile Networks

Yao Zheng, Assad Moini, Wenjing Lou, Y. Thomas Hou, and Yuichi Kawamoto

Abstract

The rapid proliferation of personal wearable as well as embedded devices point to the emergence of networks of unprecedented size and complexity in the near future. Unfortunately, traditional network security solutions fall short of addressing the unique security requirements of the emerging environment given their general emphasis on administratively managed, preconfigured security context and strong physical security mechanisms. To cope with the security challenges of this emerging environment, novel cognitive-inspired security architectures have been proposed that emphasize dynamic, autonomous trust management. Cognitive security systems take advantage of sensing and computing capabilities of smart devices to analyze raw sensor data and apply machine learning techniques to make security decisions. In this article, we present a canonical representation of cognitive security architectures and examine the practicality of using these architectures to address the security challenges of rapidly growing networks of mobile/embedded autonomous devices including the ability to identify threats simply based on symptoms, without necessarily understanding attack methods. Using authentication as the main focus, we introduce our canonical representation and define various categories of contextual information commonly used by cognitive security architectures to handle authentication requirements, and highlight key advantages and disadvantages of each category. We then examine three grand challenges facing the cognitive security research including the tension between automation and security, the unintended consequences of using machine learning techniques as a basis for making security decisions, and the revocation problem in the context of cognitive security. We conclude by offering some insight into solution approaches to these challenges.

With the number of wearable, mobile, and embedded devices on the rise, we are increasingly faced with the limitations of traditional network security control and management solutions. The vast majority of these solutions heavily rely on pre-established security contexts that are manually configured using out-of-band channels. Not only are these security contexts subject to theft and/or forgery, but they are also inadequate to cope with the evolving and ad hoc nature of trust relationships in dynamic networks of autonomous mobile/embedded devices. In fact, reestablishing and maintaining trust in these environments is a critical element of maintaining security control chains, since users and devices must be networked on each encounter. Another drawback of the traditional methods is their reliance on users' ability to retain and recall appropriate secrets. Unfortunately, human memory is limited and unreliable when it comes to storing and retrieving cryptic security contexts, especially as the number of such secrets increases. As a result, faulty memory and human errors are becoming the bane of network security.

To overcome these limitations, a new class of cognitive-inspired security methods have been proposed ranging from intelligent credential generation to autonomous dynamic trust establishment and management. Instead of relying on preconfigured secrets, these new methods seek to authenticate users or devices by recognizing patterns of behavior or correlations in information collected by such devices. Such patterns or correlations can be explicit, such as a device's geolocations or a user's keystroke dynamics, or implicit, such as location fingerprints from ambient radio signals or a user's social preferences. Such patterns and/or correlations can serve as a secondary means for verifying users' identity, hence augmenting traditional authentication methods to achieve greater assurance through use of nonintrusive and innocuous means. We refer to this family of methods as *cognitive security*, as they typically involve an intelligent reasoning process informed by machine learning (ML) techniques. We refer to the corresponding patterns or knowledge as *cognitive features*.

There has been limited progress in advancing the basic ideas underpinning cognitive security. Riva *et al.* [1] designed a decision-tree-based cognitive system to determine when it is necessary to revalidate users' identities based on their behaviors. Zheng *et al.* [2] designed a location tag-based cognitive security system to verify the geographical proximity of devices using ambient radio signals. However, their prototypes are generally considered as pure academic practices, and are not adopted as mainstream authentication measures. While critics of cognitive security often question the viability of such mech-

Yao Zheng, Assad Moini, Wenjing Lou, and Y. Thomas Hou are with Virginia Tech.

Yuichi Kawamoto is with Tohoku University.

0890-8044/16/$25.00 © 2016 IEEE

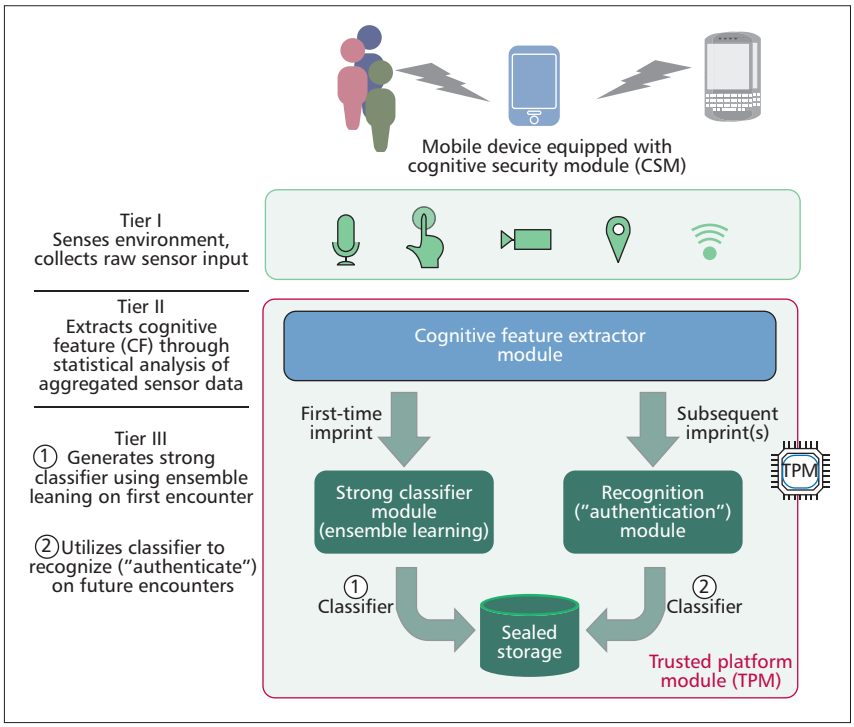

Figure 1. A three-tier general architecture for CMA.

anisms due to their low accuracy, we believe that the root of the problem runs much deeper.

In this article, to help understand the current state of cognitive security, we review the recent cognitive security systems and distill a common architecture. As a guideline, we present the desiderata for evaluating and selecting cognitive security features, and discuss the pros and cons of each one. We identify several critical factors that impede broad adoption of cognitive security architecture, including the tension between security automation and authentication, and potential vulnerability stemming from using ML-based algorithms within a cognitive authentication system.

A Canonical Cognitive Security Architecture

Cognitive security systems replace traditional challenge-and-response authentication methods with a sense-and-recognize decision process that is informed by near-real-time analysis of aggregated sensor data collected using various onboard sensor devices. A device equipped with a cognitive security module (CSM) authenticates its legitimate users or owners through a sense-and-recognize process, thus eliminating the need for negotiating a shared security context between the device and its users. Figure 1 depicts our canonical cognitive security architecture, organized into three functionally distinct tiers. The first tier consists of all the onboard sensors, and is responsible for monitoring and tracking all the interactions between the device and its environment including users or other nearby devices. Personal devices are increasingly integrating a larger array of onboard sensors and, as such, are an ideal platform for hosting a sense-and-recognize cognitive security system. Raw data collected by two or more sensors can be analyzed, correlated, and/or fused to facilitate a robust recognition decision process. A cognitive security system can utilize a touchscreen sensor to monitor its user's gestures, a gyro sensor to register users' gaits, or a microphone to record a user's voice. For device identification, a cognitive security system can utilize an available GPS sensor to establish presence of nearby devices, a camera can measure visual alignments, while an accelerometer can detect synchronized cross-device gestures. Raw sensor data collected by the first tier is aggregated at the second tier, and appropriate cognitive features are then extracted. Statistical analyses and MLtechniques are commonly employed in this tier to facilitate the recognition process. For example, voice data can be represented by and analyzed using a hidden Markov chain. An image can be segmented into a conditional random field. A series of accelerometer readings can be converted into velocities, accelerations, and moments. Selected cognitive features are then transferred to the third tier, which serves as the main processing engine for the cognitive security system. The third-tier functionality operates in two modes: learning and recognition. Learning occurs during the first long and feature-rich encounter between the device and a new (legitimate) subject; in this mode, the device learns to recognize its legitimate subject by fitting appropriately selected cognitive features using a strong classifier. To ensure the overall integrity of the cognitive security system, the classifier is enclosed in sealed, tamperproof, isolated storage protected by a trusted platform module (TPM). No one, including the device owner, can tamper with the classifier. Later, when the device interacts with a yet-to-be-recognized subject (a user or another device), it consults the classifier to categorize the subject's cognitive feature(s) to recognize its subject, thus ascertaining subject's legitimacy.

Two characteristics differentiate cognitive security systems from traditional security systems. First, the three-tier architecture does not implement an explicit challenge-and-response strategy. Rather, it relies on identity cues acquired through interactions with users. There is an advantage and a disadvantage for this design choice. The advantage is that, without a disruptive challenge-and-response protocol, the system can continuously recognize (authenticate) users and/or devices without interrupting ongoing user-device or device-device interactions. The disadvantage is that if the cognitive features do not support instantaneous authentication, there will be a delay between the suspicious activities and the authentication decision, which might be just long enough for adversaries to accomplish their goals. Second, compared to traditional security systems, cognitive security systems are usually multi-modal. Unlike traditional credentials, any cognitive feature alone cannot accurately determine users' identities [3]. Therefore, the behavior classification models for cognitive security systems usually require multiple cognitive features. To represent such models, information fusion and ensemble learning techniques are commonly employed to build a high-capacity model by combining low-fidelity cognitive features and weak classifiers.

Cognitive Features

Cognitive features are a key component of cognitive security architectures, and, as such, selecting the *right* features can go a long way in creating a robust and effective security architecture. Broadly speaking, cognitive features can be divided into three general categories: physiological, behavioral, and environmental. Physiological features represent intrinsic characteristics of users or devices such as heritage

	Physiological	Behavioral	Environmental
Human analogy	What do they look/sound like?	How do they behave?	With whom do they associate?
Examples	Heritage traits, manufacturing variations	Motor skills, inter-device communication patterns	Social relationship, ambiance signals
Distinguishability	High	Medium	Low
Consistency	High	Medium	Low
Invisibility	Low	Medium	High
Unforgeability	Low	Medium to high	High

Table 1. Three categories of cognitive features.

traits or manufacturing variations. Behavioral features characterize acquired skills or idiosyncratic aspects developed by users or devices, such as motor skills or inter-device communication patterns. Finally, environmental features represent contextual influences or conditions surrounding a user or device such as social relationships for humans or ambiance signals for devices.

Desiderata for Cognitive Features

Distinguishability: The most basic desideratum for qualifying as a cognitive feature is that it must allow us to distinguish between authorized and unauthorized users. Unfortunately, no single feature alone can provide sufficient evidence to categorically establish the identity of a specific user [3]. Among the three aforementioned types, physiological features tend to be used most mainly because they exploit intrinsic patterns of a user or a device, and are usually most useful to differentiate a particular user or device among others.

Consistency: Another factor that affects the design of cognitive security systems is the consistency of cognitive features employed by the system. Cognitive features of the same user or device can be mutually inconsistent across authentication sessions, making them less than reliable to use. This problem is common to most cognitive features often collected by mobile devices due to sporadic and erratic patterns of interactions between users and their devices.

Invisibility: Since cognitive security systems employ sense-and-recognize architecture, they are often vulnerable to shoulder surfing attacks, which often rely on similar sensing techniques. Physiological and behavioral features are often readily available to an adversary should he/she engage in surveilling the user. Environmental cognitive features, which involve identity cues hidden within natural or social environments, are more robust against shoulder surfing.

Unforgeability: The philosophy of cognitive security is based on the assumption that an adversary cannot impersonate a victim subject well enough to fool the authentication mechanism. Therefore, a natural question to ask is whether a particular cognitive feature can be forged by an attacker. Unforgeability clearly depends on not only the nature of the feature of choice but also the manner in which it is sensed, collected, processed, and ultimately used for decision making For example, a user's social circle of friends lists are easy to forge if they are used as answers to authentication questions. However, they can also be unforgeable in a vouch system where users' friends are contacted to vouch for the users' identities.

Cognitive Security Features

Table 1 presents three broad categories of cognitive features used in all cognitive security architectures. Each category is analogous to a class of mechanisms by which a human recognizes an object. For instance, physiological features correspond to the physiological characteristics on which humans rely to identify other entities or objects, including sound, pattern of speech, and so on. Physiological features excel at providing highly distinguishable and consistent identity cues. Mock *et al*. [4] combined an iris recognition algorithm with an eye tracking algorithm to provide a continuous authentication mechanism to identify desktop computer users. Kim *et al.* [5] designed a device authentication scheme by exploiting the unique power-up values in the embedded SRAM memory chip. Despite many strengths, physiological features suffer from certain weaknesses such as low, insufficient entropy. They are also highly exposed and visible, which makes them easy targets for duplication by an adversary.

Behavioral features constitute the second category of cognitive features. Gafurov [6] provided a survey for mobile user authentication using gait patterns. Ming *et al*. [7] reviewed spontaneous device association based on user interactions. Cognitive features in this category do not offer significant advantages or disadvantages with respect to the aforementioned desiderata. They are not as distinguishable and consistent as physiological features. Tey *et al*. [8] show that a user's typing pattern varies significantly under different physical and psychological conditions, such as postures and moods. As a result, behavioral features may result in a higher false rejection rate (FRR), hence undermining their own usability and effectiveness. On the positive side, behavioral features are not as exposed as physiological cognitive features, which prevents adversaries from learning such features through observation. Certain behavioral features, such as motor skills, can also be highly unforgeable. Bike riding and video gaming are everyday examples that fit this description. They can easily be verified through observation or statistics but cannot easily be duplicated.

The last category is environmental cognitive features, which associate contextual environment factors with identities. For user authentication, a user's social environment is commonly exploited to validate his/her identity. Brainard *et al.* [9] designed a cryptographic vouching system that authenticates a user by verifying multiple vouching codes sent by the user's social friends. For device authentication, the ambient context collected from onboard sensors is used. The most recent work is from [10], where Miettinen *et al.* utilized the ambient signals to update the devices' pairing credentials. The advantages of environmental cognitive features are two-fold. First, the features are physically separated from users or devices, which prevents adversaries from extracting credentials through coercion or break-in. Second, the rapidly changing environment contains high entropy information, which is difficult to forge by adversaries. However, environmental cognitive features can be fairly inconsistent due to the unpredictable factors within the environment, which reduces their applicability in certain scenarios.

Grand Challenges

Combining multiple cognitive features from distinct domains can potentially enhance the usability and efficacy of a cognitive security system. That being said, cognitive

security systems face unique challenges stemming from their very architecture, which makes them very difficult to overcome. In this section, we introduce three key challenges faced by cognitive security systems and examine potential solution approaches. In particular, we consider the perils of applying security automation to authentication and discuss vulnerabilities introduced by using low-complexity ML algorithms. Finally, we define the revocation problem of cognitive security and show how it impacts selection of cognitive features.

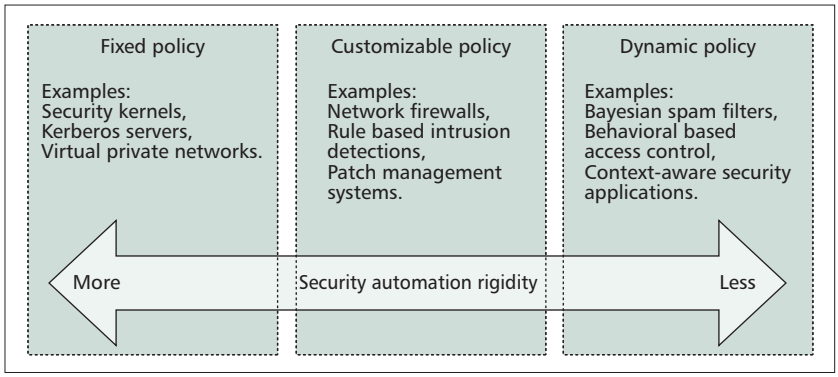

Figure 2. The spectrum of automation approaches

Security Automation for Authentication

Cognitive security is an artificial intelligence (AI) realization of security automation, which encompasses any system or technology that effectively removes the security decision and management from users. Edwards *et al.* [11] defined a range of strategies for security automation. Figure 2 presents the spectrum of automation strategies, with more rigid, less flexible automation strategies on the left and more flexible strategies on the right side of the spectrum. In this figure, the AI approach adopted by cognitive security is located on the far right end of the spectrum, where a dynamic security policy and a continuous-time adaptive system are employed for the automation. Security automation strategies of this category allow tailoring and personalization of the security environment for individual users, and as a result, they can more effectively reflect security requirements for a given situation.

However, the problem for cognitive security resides in the way security automation handles failures. Due to the existence of outliers in cognitive features, there will be false positive and false negative cases when a cognitive security system fails to determine users' or devices' identities, thus resulting in automation failure. Traditional security automation approaches handle such failures by incorporating an exception-handle-feedback loop. For instance, when a Bayesian spam filter fails to determine whether a message is spam or not, it leaves it to the user to make the decision. Based on the user's decision, the filter then retrains the spam classifier so that it can handle similar cases in the future. For cognitive security systems, however, establishing such a feedback loop is difficult. When a cognitive security system fails to identify users or devices, there is no higher authority that can handle the exception. Clearly, the system cannot rely on the very users or devices it is expected to authenticate to establish their own identities. Nor can it ignore authentication failures and let a potentially malicious user/device get by the system without any further checking. As a result, a legitimate user or device might be denied access without any further explanation.

One possible solution is to implement a secondary, more traditional authentication mechanism as a fallback measure. When the primary cognitive security system fails, it can send the unlock code to the legitimate user through an email address previously used on this device. This way, the system can hand the failure to the user and reestablish the loop. However, some users might consider this as a disruption in workflow and distraction of attention.

Another possibility is to seek "assistance" from peer trusted devices instead, that is, to delegate authentication services to a trusted device within the immediate network of devices. This is a practical approach as users are expected to carry multiple smart devices with distinct sensing capabilities. They are often equipped with peer-to-peer communication protocols such as WiFi and Bluetooth. Therefore, if the cognitive security system of one mobile device fails, it can rely on other devices in close vicinity that have successfully authenticated the users or devices. This solution essentially expands the ensemble classifier from intra-device cognitive features to inter-device cognitive features to minimize exception handling.

Finally, a more graceful way to acquire users' feedback is to contextualize cognitive security decisions. Instead of simply rejecting unidentified users or devices without further explanation, a cognitive security system may present more detailed explanation of a failed classification result, thus helping users to decide on the course of corrective actions. However, as we show later, due to the low computational complexity of machine learning algorithms, a detailed output may enable adversaries to rapidly reverse engineer and break the system.

Impact of Machine Learning on Security Functions

Here we examine the advantages and disadvantages of using ML in cognitive security systems. As stated previously, cognitive security systems rely on ubiquitous sensing of their environments to recognize users and/or other devices they encounter. As a practical matter, ML techniques can be used to discover otherwise hidden patterns and uncover correlations in sensor data collected from disparate sources. Using ML techniques can assist a cognitive security system to discern users' identities by fusing evidence from multiple sources. For example, a password can be associated with typing patterns, while face data can be linked to vocal sequences. Using a reliable learning algorithm such as SVM or Boosting, the system can then combine these separate sources of evidence to make a better judgment concerning a user's legitimacy.

A disadvantage of using ML in the context of security decision making is that ML techniques are not suited to protecting secrecy. The reason is that the computational complexity of ML algorithms is generally much lower than that of cryptographic functions. As an example, consider a cognitive security system that relies on a learning algorithm to decide users' or devices' identities using a classifier f: X {0, 1} chosen by the learning algorithm. Such classifiers are often reversible and not collision-resistant. For instance, the classifier of a linear SVM is the composition of sigmoid and a linear function

$$f(x) = h(g(x)) = \frac{1}{1+e^{-g(x)}} = \frac{1}{1+e^{-w^T x}},$$

where $x \in R^m$ is the cognitive feature vector and $w \in R^m$ is the weight vector. If adversaries are allowed to query f with m with distinct cognitive feature vectors, they can reconstruct the classifier's decision boundary and fabricate a bogus cog-

nitive feature vector to fool the system [12]. Additionally, if the adversaries gain control over a portion of the training data, they can break the integrity of the classifier by injecting fraudulent data during the training process [12]. In both cases, by exploiting the low-complexity ML algorithms, the adversaries can either slip past the security system as false negatives or block the access of legitimate users via false positives.

There are a few techniques to protect the confidentiality of the data or the classifiers in ML systems. Graepel *et al*. [13] devised a fully homomorphic Encryption (FHE) based training procedure that operates on encrypted data. Bost *et al.* [14] constructed a suite of FHE-based classification protocols that allow users to query a classifier without learning its parameters. However, these types of techniques usually rely on preshared security context, which contradicts the sole purpose of cognitive security systems.

Another solution to circumvent this problem is to use unforgeable cognitive features. This way, even if adversaries know the requirements of legitimate cognitive features, they cannot fabricate one that fits such requirements. For example, using ambient noise and luminosity to find co-present devices [2, 10] is an example that exploits the unforgeable environmental features for cognitive security. However, in these schemes, the ML algorithms are only responsible for extracting the cognitive features, whereas other mechanisms are used for the recognition/authentication process.

Appropriate use of ML techniques in a cognitive security system directly affects the system's security objective. Current cognitive security research mainly focuses on discovering the right features, that is, the ones that can uniquely identify individual users or devices. However, there is little research evaluating the system's vulnerabilities when adversaries can exploit the adaptive nature of cognitive security systems. As a result, the security analyses on most cognitive security systems are usually based on unrealistic attack models and lack rigorous proofs.

Revocation

In traditional security systems, compromised credentials must be revoked and reissued by a legitimate authority. The same principle applies to cognitive security systems; however, revoking certain features may be problematic or infeasible as is the case with biometric-based authentication systems. Nearly all the features exploited by cognitive security systems tend to be intrinsic to devices or humans and as such cannot easily be replaced or reissued. This poses a serious problem if an attacker successfully compromises the classifier used in the system. In traditional biometric authentication systems, to safeguard user biometrics, collected samples are stored as seeded hashed digests so that if they are compromised, users can easily be reenrolled and a distinct biometric template can be generated using a different seed. Unfortunately, applying such a technique to cognitive security features may not be feasible as hashing may render cognitive features indistinguishable.

One possible approach to dealing with revocation involves using modifiable cognitive security features such as acquired skills. For example, Bojinov *et al*. [15] discovered that, using serial interception sequence learning (SISL), users often subconsciously memorize specific typing sequences, which allows them to achieve typing speeds exceeding those of professional typists. This effect can be further boosted if users' original typing habits are used to select the training sequences. An adversary cannot quickly duplicate such a skill even if he/she knows the target typing speed required to authenticate. Better yet, users can be trained with new sequences when the old ones are compromised, which makes the method more amenable to revocation. Unfortunately, this method is suited to user authentication only, and finding a revocation-friendly cognitive security system for authenticating devices remains an open challenge.

Conclusion

With the rapid proliferation of mobile and networkable devices, there is a pressing need for an intelligent, adaptable security solution. Cognitive security systems replace the traditional challenge-and-response authentication model with a sense-recognize model that allows these devices to continuously identify (i.e., recognize) their legitimate users without relying on pre-established security context (e.g., shared secrets or cryptographic means). The new security model exploits the built-in sensing capabilities of each device to continuously collect information about its environment, user, and other nearby devices, then analyzes the sensed data using machine learning techniques to first learn, then recognize the identity of its subject during future encounters. In this article, we have presented the basic building blocks of a cognitive security architecture and categorized contextual features commonly used by cognitive security systems. We believe that selecting appropriate cognitive features is only the first step in developing a robust and effective cognitive security solution and an important topic for cognitive security research. To create a truly intelligent, adaptable, context-aware cognitive security solution, great challenges remain ahead, including devising an elegant security automation feedback mechanism, overcoming the limitations and improving the capacity of machine learning, and addressing system maintenance issues such as revocation. Historically, solving many network security and AI problems have proved to be challenging. We believe that combining ideas and solution techniques from two already established fields will offer new insights and opportunities for synthesizing novel solution approaches that could not have been possible otherwise and at the same time will open up new research vistas.

Acknowledgments

This work is supported in part by the U.S. National Science Foundation under grants CNS-1405747, CNS-1446478, and CNS-1443889.

References

[1] O. Riva *et al.*, "Progressive Authentication: Deciding When to Authenticate on Mobile Phones," *Proc. USENIX Security Symp.*, 2012, pp. 301–16.
[2] Y. Zheng *et al.*, "Sharp: Private Proximity Test and Secure Handshake with Cheat-Proof Location Tags," *Proc. Computer Security ESORICS 2012*, Springer, 2012, pp. 361–78.
[3] R. Greenstadt and J. Beal, "Cognitive Security for Personal Devices," *Proc. 1st ACM Wksp. AISec*, Alexandria, VA, 2008.
[4] K. Mock *et al.*, "Real-Time Continuous Iris Recognition for Authentication Using an Eye Tracker," *Proc. 2012 ACM Conf. Computer and Commun. Security*, Raleigh, NC, 2012.
[5] J. Kim, J. Lee, and J. A. Abraham, "System Accuracy Estimation of SRAM-Based Device Authentication," *Proc. 16th Asia and South Pacific Design Automation Conf.*, Yokohama, Japan, 2011.
[6] D. Gafurov, "A Survey of Biometric Gait Recognition: Approaches, Security and Challenges," *Proc. Annual Norwegian Computer Science Conf.*, 2007, pp. 19–21.
[7] M. K. Chong, R. Mayrhofer, and H. Gellersen, "A Survey of User Interaction for Spontaneous Device Association," *ACM Computer Surveys*, vol. 47, 2014, pp. 1–40.
[8] C. M. Tey, P. Gupta, and D. Gao, "I Can be You: Questioning the Use of Keystroke Dynamics as Biometrics," 2013.
[9] J. Brainard *et al.*, "Fourth-Factor Authentication: Somebody You Know," *Proc. 13th ACM Conf. Computer and Commun. Security*, Alexandria, VA, 2006.
[10] M. Miettinen *et al.*, "Context-Based Zero-Interaction Pairing and Key Evolution for Advanced Personal Devices," *Proc. 2014 ACM Conf. Computer and Commun. Security*, Scottsdale, AZ, 2014.

[11] W. K. Edwards, E. S. Poole, and J. Stoll, "Security Automation Considered Harmful?" *Proc. 2007 Wkskp. New Security Paradigms*, 2008, pp. 33–42.
[12] M. Barreno *et al.*, "The Security of Machine Learning," *Machine Learning*, vol. 81, 2010, pp. 121–48.
[13] T. Graepel, K. Lauter, and M. Naehrig, "ML Confidential: Machine Learning on Encrypted Data," *Proc. Information Security and Cryptology ICISC 2012*, Springer, 2013, pp. 1–21.
[14] R. Bost *et al.*, "Machine Learning Classification over Encrypted Data," *Proc. Network and Distributed System Security Symp.*, 2015.
[15] H. Bojinov et al., "Neuroscience meets Cryptography: Crypto Primitives Secure Against Rubber Hose Attacks," *Commun. ACM*, vol. 57, 2014, pp. 110–18.

Biographies

YAHO ZHENG (zhengyao@vt.edu) is a Ph.D. student at Virginia Tech. He received his B.S. degree in microelectronics from Fudan University in 2007 and his M.S. degree in electrical engineering from Worcester Polytechnic Institute in 2011. Between 2007 and 2009, he was a researcher at Siemens AG, Beijing, China. His research interests include information security and privacy, machine learning, and cryptography.

ASSAD MOINI (amoini@vt.edu) is a Ph.D. student at Virginia Tech and the chief technology officer at Five9Group. He has 30 years of industry experience in all areas of systems and software engineering, network security, and applied R&D in defense, aerospace, and the commercial market. He has served as Co-PI and directed many research and development efforts.

WENJING LOU [F] (wjlou@vt.edu) has been a professor of computer science at Virginia Tech since 2011. From 2003 to 2011, she was a professor in the Department of Electrical and Computer Engineering at Worcester Polytechnic Institute. She received her Ph.D. in electrical and computer engineering from the University of Florida. Her research interests lie in cybersecurity and wireless networks.

Y. THOMAS HOU [F] (thou@vt.edu) is the Bradley Distinguished Professor of Electrical and Computer Engineering at Virginia Tech, Blacksburg. He received his B.E. degree from the City College of New York in 1991, his M.S. degree from Columbia University in 1993, and Ph.D. degree from NYU Polytechnic School of Engineering in 1998, all in electrical engineering. From 1997 to 2002, he was a researcher at Fujitsu Laboratories of America, Sunnyvale, California.

YUICHI KAWAMOTO (youpsan@it.ecei.tohoku.ac.jp) received his B.E. degree from the School of Information Engineering, and M.S. degree from the Graduate School of Information Science (GSIS) at Tohoku University, Japan, in 2011 and 2013, respectively. Currently, he is pursuing a Ph.D. degree in the GSIS at Tohoku University. He received the Best Paper Award at IEEE GLOBECOM '13 and at the International Wireless Communications and Networking Conference 2014. He is recipient of an award from the Japan Society for the Promotion of Science in 2015.