On the Design of Distributed Adaptive Authentication Systems

Patricia Arias-Cabarcos[*]
University Carlos III of Madrid
Leganés, Spain
ariasp@it.uc3m.es

Christian Krupitzer
University of Mannheim
68131 Mannheim, Germany
christian.krupitzer@uni-mannheim.de

ABSTRACT

Adaptive authentication allows a system to dynamically select the best mechanism for authenticating a user depending on contextual factors, such as location, proximity to devices, and other attributes. Current systems in the literature are built to demonstrate feasibility and basic usability improvements in specific scenarios, but none of them follows a methodological approach for system design, neglecting the huge body of research on adaptation. In this position paper, we posit the necessity to apply such a structured modelling procedure and show its potential benefits to achieve better and more usable designs. We discuss the modelling steps to be followed, identify key challenges to be addressed, and present an initial reference architecture for adaptive distributed authentication.

1. INTRODUCTION

Usable authentication research has been dominated by efforts to improve the ease and security of authentication mechanisms, and, particularly, by the exploration of alternatives to the omnipresent text passwords. We are witnessing the development of new less intrusive ways of authenticating users, such as gait recognition, keystroke dynamics, or even authentication by how the user sings, or thinks [15]. However, since there is no *one-size-fits-all* in security, no new mechanism is going to replace all the others and be accepted as the universal solution. In fact, some mechanisms are preferred under certain environmental conditions (e.g., voice recognition while driving), others are more secure to access sensitive applications (e.g., multifactor), and some of them can be only used in devices with the appropriate sensors and capabilities (e.g., brainwaves or heartbeat biometrics). Thus, a path to take the most of this heterogeneity and achieve better security and usability is the design of systems that are able to sense the environment and adapt the authentication mechanism to the surrounding conditions or context, as explored in [4, 16, 9, 12, 17, 3, 5].

We set out to review the significant literature on adaptive authentication (Section 2) and found it noticeable that current proposals have not been designed following the methodological principles that are well-known in the adaptive systems discipline [2, 6]. Instead, since the focus for authentication systems has been put on demonstrating feasibility rather than design, the so far proposed systems are difficult to extend or reuse (e.g., to include new authenticators[1] , adaptation strategies, contexts), which hinders faster advance on research. Furthermore, the lack of design analysis has led to poor formalizations of usability goals. To improve the situation, this paper arguments the importance of pursuing a methodological approach to design. We describe the modelling steps that should be followed and discuss how they can be applied to the authentication domain, characterizing the full problem space (Section 3). As an outcome of this procedure, we present our work-in-progress towards a reference architecture for adaptive authentication, identifying key research challenges (Section 4), as well as our main conclusions after this study (Section 5).

2. ADAPTIVE AUTHENTICATION SYSTEMS: A BRIEF REVIEW

Description. In the early 2000s there were already initial proposals on adaptive authentication tied to the appearance of the first ubiquitous computing systems, such as e.g., *Cerberus* for *GAIA* [1]. However, since efforts were soon switched to finding alternatives to passwords, it is not until the beginning of this last decade that research on adaptive authentication gained traction again. Among the most relevant recent works we find [4, 16, 9, 12, 17, 3, 5], whose main features are described in the following. The *CASA* framework [4] presents a probabilistic model to adapt user-to-smartphone authentication. Based on three location contexts (Home, Work, Other), the locking screen switches among PIN, password, or no authentication. The adaptation metrics are the probabilities that the user is correctly authenticated at a specific place using a specific method, which are difficult to estimate without wide-scale studies, and hard to extend to other contexts and authenticators. Similarly, *TreasurePhone* [16] adapts authentication based on location context, but this approach protects access to smartphone applications with different sensitivities, being more granular. There is another work closely-related to *TreasurePhone* [9], which adapts authentication to applications in a smart-space scenario. In this solution, adaptation of the authentication

[*]Supported by an Alexander von Humboldt Postdoctoral Fellowship at University of Mannheim

[1]We use authenticator and authentication mechanism interchangeably throughout the paper

Symposium on Usable Privacy and Security (SOUPS) 2017, July 12–14, 2017, Santa Clara, California.

mechanism is based on the reliability that the user is positioned in front of the application, which is communicated by different user devices pre-registered with the system as accredited tokens. The three approaches described so far [4, 16, 9] do not incorporate authentication mechanisms other than traditional password/PIN, and even require the user to carry additional tokens (e.g., accredited devices, NFC tags). Nevertheless, there are other recent solutions [12, 17, 3] that introduce more usable authenticators and whose adaptation techniques are not only based on changes on the security level inferred by the context, but also include contextual factors related to usability. One example of such an approach is *Progressive authentication* [12], which continuously adapts access to smartphone applications based on multimodal biometric signals and PIN-based authentication outcomes fed to a classifier. The trust score given by the classifier is mapped to three sensitivity levels under which applications are categorized, and signals can be switched off/on depending on the required sensitivity level. Another example of usable adaptation is [17], which adjusts the smartphone lock mechanism between voice recognition, face scan, and fingerprint, depending on which one is more usable for the current context based on sensed activity, light level, noise level, etc. The main gap in [17] is the lack of security-based adaptation, though they mention that this functionality could be added by reasoning on the FAR (False Acceptance Rate) and FRR (False Reject Rate) values of the authenticators. In this sense, [3] does characterize the security strength of 15 different authenticators and defines an algorithm that selects a multifactor authentication mechanism optimized for the security level required by an application and for the environmental usability conditions. However, the selection is limited to multifactor authenticators and the proposal is centered on the selection algorithm but does not define how this logic can be included and executed within an adaptation architecture. In this sense, none of the above works describe an architecture that decouples all the adaptation functionalities, capturing both usability and security features, and facilitating seamless integration of different authenticators, contexts, and adaptation algorithms. *CORMORANT* [5] aims at a similar goal, but still does not capture all the dimensions of an adaptive system.

Position. When reflecting on the design of current adaptive authentication systems, we found out that no work so far has applied a methodological approach, reason for which they lack completeness and extensibility. In the following, we show that by applying a systematic modelling approach grounded in the adaptive systems discipline [2, 6], it is possible to fully characterize the adaptive authentication problem space and define a reference architecture for distributed adaptive authentication that facilitates easy integration of diverse authenticators. All in all, this will foster research advance on adaptive authentication as it will permit collaboration of different specialized communities, which can focus on sub-problems that can be contributed as components towards the higher goal of adaptive authentication, e.g., definition of usability and authentication strength metrics, design of new implicit authenticators, definition of contexts, or adaptation algorithms. Instances of the reference system would be directly comparable and could be evaluated under the eyes of both usable security and adaptive systems experts.

3. SYSTEM MODEL

An adaptive system is composed by a set of managed resources and its adaptation logic, which *monitors* the environment and managed resources (M), *analyzes* the data for changes (A), *plans* adaptation (P), and controls the *execution* of the adaptation (E), based on a shared *knowledge* repository (K). These activities are known as MAPE-K cycle [6], and they are implemented within a feedback loop in the adaptation logic. To model what should be done in the MAPE logic, there are a set of dimensions to consider, derived from the answer to five basic questions: *Why*, *When*, *What*, *Where* and *How* to adapt? [8]. By transferring these concepts to adaptive authentication, we confront the design of systems were the managed resources are the devices with authenticators. Thus, to depict the design possibilities of the adaptation logic that controls these resources, in the following, we answer the five adaptation questions by naming the adaptation dimensions from the taxonomy presented in [8] that are relevant for adaptive authentication. Figure 1 shows an overview of the resulting adaptation taxonomy for authentication.

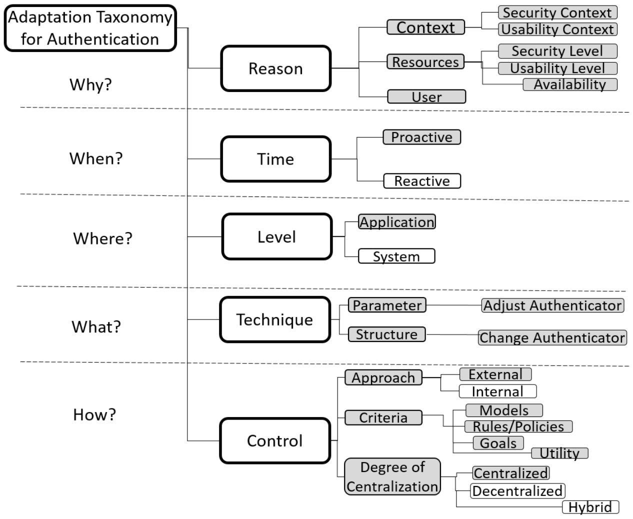

Figure 1: Taxonomy of modelling dimensions and guiding questions for adaptive systems' design based on [8], tailored to the authentication domain. Grey boxes highlight desirable features in an adaptive authentication system.

3.1 Why to adapt?

In an adaptive system, the reason for adaptation is a change on one or more system elements: technical resources, environment, and/or user [6]. First, in the specific case of authentication, the technical resources are the devices with different authenticators available for the user, which may be active or inactive. For example, if the system detects that there is a user-owned fixed computer in the proximity of her smartphone, authentication to the latter can be based on periodic face scans realized by the computer camera, or through typing dynamics, instead of activating a password screen or other locking mechanism in the phone. Second, with regards to the environment, there is a wide range of factors that may impact authentication selection. On the one hand, there are factors related to security like the location context (which changes the attack surface), authentication mechanisms' strength, or application sensitivity levels. On the other hand, there are environmental conditions that affect usability, like noise levels for voice recognition, user activity, or battery level. Third, changes on user authentication preferences can be considered for better adaptation, e.g., personal interaction preferences or those related to user disabilities. Furthermore, when there are various users registered to the authentication system, adaptation is also triggered when a change on the user is detected. Architecturally, when designing for this dimension, we decide what should be monitored, i.e., the security and usability values of the authenticators, as well as all the factors influencing them. These data determine the input to the selection algorithms, which should be designed at this point to optimize both security and usability. In this sense, the continuous or discrete nature of the different authenticators has design implications. Therefore, when the pool of managed resources includes continuous authenticators [10], algorithms can be defined to maintain the user authenticated during a session and only re-authenticate when changes make necessary to raise the level of authentication. On the contrary, if we only manage one-time authenticators (e.g., password, fingerprint), algorithms cannot maintain a continuously authenticated session. Here it is interesting to compare security/usability trade-offs considering the time to detect a malicious user and performance costs in the continuous case, versus the lower consumption of one-time authenticators but their inability to detect a change on the user. Finally, the adaptation reason must be in line with user-defined or organization-defined authentication requirements. For example, a company implementing an adaptive authentication system may require all the authentications happening in common rooms to be hardened with respect to authentication events in personal offices.

3.2 When to adapt?

Reasoning for adaptation can be triggered reactively or proactively. Reactive reasoning triggers an adaptation after an event, whereas proactive reasoning prepares adaptation or adapts the system if the adaptation logic anticipates events that would trigger adaptation [6]. In the scope of authentication, reactiveness means that the selected authentication mechanism is chosen when the user tries to access an application or device. In the proactive case, we have, e.g., systems that automatically change the authentication mechanism for device lock/unlock when the location of the user is considered more secure. In this latter situation, though there is no need to adapt until the user wants to access the device, the selected mechanism is anticipated and ready for the time of authentication, which makes the process faster and more seamless. Architecturally, this distinction between reactive and proactive adaptation has an impact on how to design algorithms for analysing the monitored data. Reactive approaches just need to continuously monitor user access events, at which point additional context information is acquired and analysed on-the-fly. In turn, proactive approaches need to continuously monitor and analyse more data combined with prediction for anticipated preselection. Common examples of these data are: location, battery level, activity, or environmental conditions like noise or light, aligned with the usability and security goals established in the *Why?* modelling step. Regarding usability, the time of adaptation is a central question. Proactive adaptation is preferable because it avoids interruptions in the user's workflow. However, time and battery consumption may be an issue due to the intensive monitoring and analysis and the associated frequent changes of authenticators. Ideally, the system could predict the intention of the user to authenticate and change the mechanism only in that case, but the complexity of prediction algorithms of that kind would presumably increase. Thus, important questions that need to be explored when designing and testing for usability regarding the time dimension are: Would a proactive approach be perceived as more or less usable? Would the time and battery consumption be unacceptable for user adoption? Would such automatic inferences lead to the system appearing more trustworthy or the contrary? Would a reactive approach imply unacceptable delays for users in the authentication process? How would these approaches affect the design of consistent authentication "ceremonies"? Could we use hybrid approaches for better trade-offs between seamlessness-performance?

3.3 Where to adapt?

A system might be adapted on several levels, e.g., application, operation system, or communication (middleware) [6]. The adaptation logic must be aware of the relevant levels for a specific system. In the authentication domain, applicable levels are system and application. System-level adaptation implies that the selected authenticator gives access to the whole system, while application-level adaptation means that a different authenticator is selected for each application. Architecture solutions covering the application level are desirable because they provide room for more granular security in the adaptation strategies. That is, when adapting for a whole system, the strength of the selected authenticator must fit the highest level of sensitivity of all the applications in the system, and so its usability might be worse than that of lower strength authenticators required for most applications. The counterpart of application-level adaptation is the need for configuration of adequate policies mapping sensitivity to authentication levels. In this regard, an important usability question is: How can policies be configured in a user-friendly way?

3.4 What to adapt?

It is not sufficient to only identify the levels where the adaptation should take place. Additionally, the specific adaptation actions that should be carried out on those levels need to be also identified, i.e., the adaptation technique. Techniques can be either parametric, which modify system behavior by adjusting system parameters; or structural, which subsume changes in the structure of the technical system, i.e., an exchange of components, a new composition of components, or the removal/addition of components [6]. In the authentication domain, both techniques are applicable. On the one hand, parametric adaptation can be used to adjust an inner element of a specific authenticator, such e.g., the number of features in face recognition, or the use of feature-level, score level or fusion-level algorithms in multi-modal authentication [13]. Examples of this kind of parametric adaptation are the work in [11], an approach that improves accuracy of smartphone gait-based authentication by changing the parameter "user template" depending on device placement; or the multimodal authenticator implemented in Progressive

Authentication [12], which can switch on/off the different signal parameters, - such as face, voice, or placement. - fed to a classifier. On the other hand, structural adaptation can be used to activate or deactivate an authentication mechanism. A flexible adaptive authentication architecture should provide components for achieving both techniques. Furthermore, parameters should be mapped to usability and security values, so they can be converted in actionable elements to be orchestrated by the adaptation logic.

3.5 How to adapt?

The last modelling dimension refers to the adaptation control, for which the literature on adaptive systems describes three different aspects, namely approach, criteria, and degree of centralization [6]. The logic can thus follow an internal approach, which intertwines the adaptation logic with the system resources; or an external approach, which splits the system into adaptation logic and managed resources, increasing maintainability through modularization. With regards to criteria, the logic can be based on models, goals, rules/policies, utility functions, or combinations of different criteria. The adaptation possibilities must be analyzed with the help of the criteria and the best one must be chosen. Another aspect of the adaptation logic is the degree of decentralization, hence, the distribution of the MAPE-K components which can lead to centralized, decentralized, or hybrid adaptation logics. When architecting for the adaptive authentication domain, external approaches are preferred for flexibility. All the different criteria could be applicable and their performance should be evaluated to find out which alternative offers the best solution. Hence, the approach must offer flexibility regarding analysis and planning, i.e., the used algorithms need to be exchangeable. Finally, centralization is desirable for controlling the swarm of user devices in a cohesive way, allowing cooperation towards authentication with low communication overheads compared to decentralized approaches. An additional usability aspect to consider here is who is the owner/administrator of the system. If an adaptive authentication system protects user devices and applications, then all the management tasks fall into the hands of users. If the system is run by a company and protects access to its services, the organization would oversee the management but this comes with privacy issues related to the collection of user information for implicit authentication: How should the user be notified? Would the user be eager to adopt such a system despite the collection of personal data? How complex would be the registration procedures?

4. A REFERENCE ARCHITECTURE

Here we outline the architecture (Figure 2) for an adaptive authentication system. The Adaptation Logic (AL) is implemented in a centralized fashion, which might be available as Cloud service for devices. The AL integrates a MAPE-K loop according to the presented system model: The Monitor component includes three modules to register context, user, and authenticator related changes. The monitoring functionality is distributed, with sensors implemented in the Managed Resources and orchestrated by the AL. Furthermore, an Authenticator Registry database is dynamically updated in the Knowledge component with the available authentication mechanisms and their metadata (usability features, security features, and communication endpoints for adaptation). All these data are fed to the Analyzer module in the Analysis component, which searches for adaptation events and notifies the Plan component. The Plan component can be seen as the adaptation brain, whose logic is defined in the Authenticator Selector module, which implements the selection logic to determine which of the available authenticators is optimal for the sensed conditions. Different algorithms can be implemented as plugins built on Rules/Policies, Models, Utilities, or Goals.

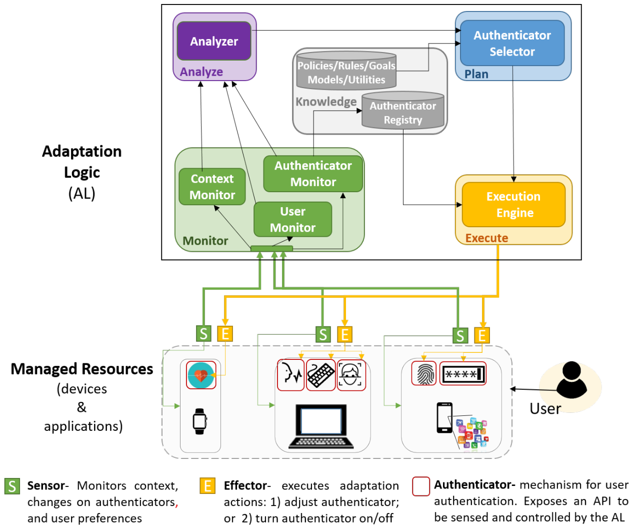
Adaptation Logic (AL)
Analyzer
Analyze
Policies/Rules/Goals Models/Utilities
Knowledge
Authenticator Registry
Authenticator Selector
Plan
Context Monitor
Authenticator Monitor
User Monitor
Monitor
Execution Engine
Execute
S
E
Managed Resources (devices & applications)
****
User
S Sensor- Monitors context, changes on authenticators, and user preferences
E Effector- executes adaptation actions: 1) adjust authenticator; or 2) turn authenticator on/off
Authenticator- mechanism for user authentication. Exposes an API to be sensed and controlled by the AL

Figure 2: Reference architecture for adaptive authentication.

Decisions made in the Plan component are moved to the Execute component to be translated into actions. The execution of adaptations is distributed: the Execution Engine communicates with the Effectors located in the Managed Resources, indicating which parameters should be changed for adaptation, or activating/deactivating a complete authenticator. The Effectors perform these modifications in the associated Managed Resource. Starting from this high-level architecture, we aim at defining standard interfaces and input formats for the different modules. With such a generic definition, elimination, addition, or exchange of elements would be straightforward. To achieve this purpose, we have elicited as key research challenges:

[RC-1] *Authenticator Abstraction.* It is required to define an abstraction that provides standardized means to discover or detect the presence of an authenticator, read its features and access its functionalities: activate/deactivate or adjust behavior through parameters. Effectors and Sensors will be built on platform-specific APIs implementing this abstraction. This definition requires investigating which features are relevant for analyzing adaptation events and for authenticator selection. In this sense, minimum required features are usability and security strength values for different contexts, which leads to RC-2.

[RC-2] *Authenticator Metrics.* It is required to investigate which metrics are adequate to describe strength and usability for authenticators of different types (continuous/one-time, biometric/token/knowledge-based, probabilistic/deterministic, etc.), and how to characterize them to be accessible in a standardized way. Existing strength metrics are, e.g., the NIST Levels of Assurance, or FAR/FRR rates for biometrics. For usability, there are ongoing proposals to standardize metrics like e.g., the SUS metric [14]. Furthermore, the relation of metrics to context should be modelled, i.e., the description of functions or rules that reflect the increase/decrease of usability and security values with respect to contextual factors. This leads to RC-3.

[RC-3] *Context Modelling.* It is required to investigate which contextual factors impact the usability of an authenticator and which other contextual factors impact security. Furthermore, we need to define standardized means for context representation, activation, and deactivation, to make context available to programmers of authentication mechanisms.

5. CONCLUSIONS

We have analyzed the modelling dimensions for adaptive authentication systems and presented a reference architecture that we believe could be the basis for faster collaborative research. We plan to address the identified research challenges by completing the architecture design and implementing a proof-of-concept prototype. For the implementation, the FESAS [7] framework developed in the adaptive systems community, is a suitable candidate for implementing the adaptation logic as it offers a set of reusable process elements and system components that will make practical realization easier. Additionally, FESAS focuses on simplifying the reusability and exchange of algorithms in the adaptation logic which enables tailoring of adaptive authentication to an application's requirements, e.g., by supporting different adaptation metrics (cf. Section 3.5). The prototype will demonstrate the designed abstractions, including a basic set of diverse authenticators, selection algorithms and plugins. Based on it, we aim at incrementally testing and comparing different adaptive authentication configurations.

6. REFERENCES

[1] J. Al-Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas. Cerberus: a context-aware security scheme for smart spaces. In *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*, pages 489–496. IEEE, 2003.

[2] B. H. Cheng, R. De Lemos, H. Giese, P. Inverardi, J. Magee, J. Andersson, B. Becker, N. Bencomo, Y. Brun, B. Cukic, et al. Software engineering for self-adaptive systems: A research roadmap. In *Software engineering for self-adaptive systems*, pages 1–26. Springer, 2009.

[3] D. Dasgupta, A. Roy, and A. Nag. Toward the design of adaptive selection strategies for multi-factor authentication. *Computers & Security*, 63:85–116, 2016.

[4] E. Hayashi, S. Das, S. Amini, J. Hong, and I. Oakley. Casa: context-aware scalable authentication. In *Proceedings of the Ninth Symposium on Usable Privacy and Security*, page 3. ACM, 2013.

[5] D. Hintze, R. D. Findling, M. Muaaz, E. Koch, and R. Mayrhofer. Cormorant: towards continuous risk-aware multi-modal cross-device authentication. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pages 169–172. ACM, 2015.

[6] J. O. Kephart and D. M. Chess. The vision of autonomic computing. *Computer*, 36(1):41–50, 2003.

[7] C. Krupitzer, F. M. Roth, C. Becker, M. Weckesser, M. Lochau, and A. Schürr. Fesas ide: An integrated development environment for autonomic computing. In *Autonomic Computing (ICAC), 2016 IEEE International Conference on*, pages 15–24. IEEE, 2016.

[8] C. Krupitzer, F. M. Roth, S. VanSyckel, G. Schiele, and C. Becker. A survey on engineering approaches for self-adaptive systems. *Pervasive and Mobile Computing*, 17:184–206, 2015.

[9] G. Lenzini, M. S. Bargh, and B. Hulsebosch. Trust-enhanced security in location-based adaptive authentication. *Electronic Notes in Theoretical Computer Science*, 197(2):105–119, 2008.

[10] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, 2016.

[11] A. Primo, V. V. Phoha, R. Kumar, and A. Serwadda. Context-aware active authentication using smartphone accelerometer measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 98–105, 2014.

[12] O. Riva, C. Qin, K. Strauss, and D. Lymberopoulos. Progressive authentication: Deciding when to authenticate on mobile phones. In *USENIX Security Symposium*, pages 301–316, 2012.

[13] A. Ross and A. K. Jain. Multimodal biometrics: An overview. In *Signal Processing Conference, 2004 12th European*, pages 1221–1224. IEEE, 2004.

[14] S. Ruoti and K. Seamons. Standard metrics and scenarios for usable authentication. In *Symposium on Usable Privacy and Security (SOUPS)*, 2016.

[15] M. A. Sasse. "technology should be smarter than this!": A vision for overcoming the great authentication fatigue. In *Workshop on Secure Data Management*, pages 33–36. Springer, 2013.

[16] J. Seifert, A. De Luca, B. Conradi, and H. Hussmann. Treasurephone: Context-sensitive user data protection on mobile phones. In *International Conference on Pervasive Computing*, pages 130–137. Springer, 2010.

[17] A. Wójtowicz and K. Joachimiak. Model for adaptable context-based biometric authentication for mobile devices. *Personal and Ubiquitous Computing*, 20(2):195–207, 2016.