Intel® Technology Journal

JUNE 2014

Towards Secure and Usable Authentication

Publisher
Cory Cox

Managing Editor
Stuart Douglas

Content Architect
Bob Steigerwald and Anand Rajan

Program Manager
Stuart Douglas

Technical Editor
David Clark

Technical Illustrators
MPS Limited

Technical and Strategic Reviewers
Abhilasha Bhargav-Spanzel
Narjala Prakash Bhasker
Cory Cornelius
Lucas Davi
David Durham
Nathaniel J. Goss
Nathan Heldt-Sheller
Catherine Huang
Shengcai Liao
Jason Martin
Jennifer McKenna
Anand Rajan
Craig Schmugar
Hannah L. Scurfield
Ned Smith
Bob Steigerwald
Chieh-Yih Wan
Kevin C. Wells

Intel Technology Journal

ISBN 978-1-934053-67-6, ISSN 1535-864X

Intel Technology Journal
Volume 18, Issue 4

Publisher: Cory Cox
Managing Editor: Stuart Douglas

Library of Congress Cataloging in Publication Data:

Printed in China
10 9 8 7 6 5 4 3 2 1

First printing: July 2014

Notices and Disclaimers

ALL INFORMATION PROVIDED WITHIN OR OTHERWISE ASSOCIATED WITH THIS PUBLICATION INCLUDING, INTER ALIA, ALL SOFTWARE CODE, IS PROVIDED "AS IS", AND FOR EDUCATIONAL PURPOSES ONLY. INTEL RETAINS ALL OWNERSHIP INTEREST IN ANY INTELLECTUAL PROPERTY RIGHTS ASSOCIATED WITH THIS INFORMATION AND NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHT IS GRANTED BY THIS PUBLICATION OR AS A RESULT OF YOUR PURCHASE THEREOF. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY RELATING TO THIS INFORMATION INCLUDING, BY WAY OF EXAMPLE AND NOT LIMITATION, LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR THE INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT ANYWHERE IN THE WORLD.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

For more information go to http://www.intel.com/performance

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.
Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm

TOWARDS SECURE AND USABLE AUTHENTICATION

Articles

Foreword 7

Authenticate Once and Be Done: User-Centric Authentication through Rich Device Capabilities 8

Biometrics from the User Point of View: Deriving Design Principles from User Perceptions and Concerns about Biometric Systems 30

A Survey of Biometrics for Wearable Devices 46

Estimating the Utility of User-Passive Authentication for Smartphone Unlock 64

Heterogeneous Face Recognition: An Emerging Topic in Biometrics 80

On Designing SWIR to Visible Face Matching Algorithms 98

Adding Nontraditional Authentication to Android* 120

Security Analysis of Mobile Two-Factor Authentication Schemes 138

Trusted Execution Environment for Privacy Preserving Biometric Authentication 162

Protecting Sensor Data from Malware Attacks 178

The New Paradigm: When Authentication Becomes Invisible to the User 198

Foreword

Elisa Bertino
Computer Science Department, CERIAS, and Cyber Center, Purdue University

The problem of cyberspace securityis longstanding and is perhaps one reason why the Internet and other online environments haven't been used to their full potential. Among the many aspects of this problem, a major challenge is the reliable and convenient authentication of users, devices, and other parties in online environments. We have all heard about "digital identity theft," which generally refers to malicious parties stealing individuals' passwords or credentials for malicious purposes. Wide-scale adoption of mobile smart devices and new developments in the area of sensors and the Internet of Things is making the problem even more complex. Users now interact with a variety of parties, including other users, application, sensors, and while on the move and in different contexts. Protecting digital identity and at the same time securely authenticating users is a critical requirement.

Digital identity is a complex notion, and many definitions exist. We can define identity as the digital representation of information known about an individual or party. Such information, referred to as *identity attributes*, encompasses not only attributive information, such as social security number, dateofbirth, and country of origin, but also biometrics, such as iris or fingerprint features, and information about user activities, including Web searches and e-shopping transactions, and location and mobility patterns.Another definition is by the International Telecommunication Union that defines identity as "information about an entity that is sufficient to identify that entity in a particular context." This definition includes identifiers such as login names and pseudonyms. Yet another (complementary) definition of identity is that it is a claim a party makes about itself or some other party. The term "claim" refers to an assertion about the truth of something—typically, a truth that is disputed or in doubt. This definition points out that digital identities must be verified through an authentication process. Authentication has many forms, ranging from passwords to smartcards and biometric verification.

Providing secure and privacy-preserving digital identity management and authentication requires addressing many challenges, including effective and continuous biometric-based authentication, flexible multifactor authentication, biometrics-based authentication on mobile devices, and security of authentication devices and sensors. I believe however that today we are in the position of being able to address many of these challenges. Progress has been made possible by important research advances in computer and network security, mobile device security, and biometric techniques. This *Intel Technical Journal Special Issue "Towards Secure and Usable Authentication"* provides an exciting view of recent advances that are making possible the secure and trusted use of identities in cyberspace.

A first group of articles in the special issue addresses different aspects of biometric-based authentication, including user perception of biometrics, face recognition and face matching between heterogeneous modalities, and continuous biometric authentication. Together these articles show that biometrics techniques are becoming very reliable. However, they clearly indicate that user perception is critical for the adoption of specific types of biometrics and that further challenges need to be addressed in order to support advanced uses of biometrics, such ascontinuous biometrics-based authentication. A second group of articles explores the use of rich capabilities offered today by mobile devices to authenticate users based on their context and for addressing even more challenging authentication requirements, such as continuous presence of a user in a given location. The articles show how these rich capabilities and environments enhance the usability and user convenience of authentication while at the same time supporting strongly assured authentication processes. Finally, a third group of articles focuses on security aspects of authentication and mobile devices. These articles cover attacks on multifactor authentication as well as protection techniques for mobile device sensors and trusted execution environments.

I believe that together these articles provide a comprehensive view of recent research efforts and technology advances in the area of digital identity management and discuss exciting research directions. I trust that from them you will get interesting insights and new research ideas. Enjoy the articles!!

AUTHENTICATE ONCE AND BE DONE: USER-CENTRIC AUTHENTICATION THROUGH RICH DEVICE CAPABILITIES

Contributors

Jason Martin
Intel Labs

Anand Rajan
Intel Labs

Bob Steigerwald
Intel Security Group

Today's authentication suffers from unsolved problems in usability and security. Adversaries have multiple attack vectors with which to steal user credentials, including phishing, malware, and attacks on service providers. Current security practices such as password-complexity policies and idle timeouts often compromise usability. Our vision is to provide the ideal balance of usability and security for authentication by leveraging the ever-increasing rich sensing capabilities of user devices as well as hardening the security capabilities of those devices. We establish the need for three key enhancements: strong multifactor user authentication, continuous user presence, and remote attestation. Through this approach, we aim to raise the security bar on usages that have traditionally favored convenience over authentication and improve the user experience of scenarios that focus on strong authentication today.

Introduction

"Authentication is the process of establishing confidence in user identities electronically presented to an information system."

Authentication is the process of establishing confidence in user identities electronically presented to an information system. For in-person transactions, people are often required to produce some form of physical identification such as a student ID, driver's license, or passport: *something you have*. While there have been recent advances in using biometrics or *something you are* for online transactions, the default method for many years has been a user ID and password, or *something you know*. Unfortunately, because of the growth of malware, attacks, and identity theft, passwords have become quite onerous in their complexity. When combined with web site policies for frequently changing a password or requesting that a user reenter a password after a timeout, the entire password model has become increasingly frustrating. The usability of the model is poor and leads people to come up with coping mechanisms that compromise security.

"…we describe a vision in which our increasingly capable and sensor-rich computing devices simplify factor collection and processing,…"

This article describes advances in approaches to multifactor authentication—a combination of two or more of *what you are*, *what you have*, and *what you know*. In this article, we describe a vision in which our increasingly capable and sensor-rich computing devices simplify factor collection and processing, dramatically improving the user experience while simultaneously raising the security assurance for relying parties. In the upcoming sections we review why there is a growing need for better authentication methods, the tradeoffs that exist between security and usability, our vision for highly secure multifactor authentication with continuous presence, and the efforts to ensure that this solution is consistent with evolving industry standards. In the process, we also make forward references to other articles in this edition of the *Intel Technology Journal* that provide greater depth on select topics.

The Growing Need for Better Authentication

Anyone who spends time online (beyond merely web surfing) is keenly aware of the methods web sites use to authenticate. Complex passwords, wavy text to interpret, pictures to confirm, a pin sent to a cell phone, secret questions, and so on are all designed to prevent other people or computers from impersonating you. In this section we examine the growing issue of identity theft and the many attack vectors that adversaries are using to get past site security and exploit your identity. We also explore the variety of ways users interact with services and how that negatively impacts the user experience of authentication.

Identity Theft Is a Growing Issue

Identity theft occurs when someone pretends to be someone else by assuming that person's identity, usually to obtain credit card, bank account, or other personal information for the purpose of committing fraud. A typical path to identity theft is to first obtain someone's login ID and password through a phishing scam. Unfortunately many people who fall prey to one of these scams exacerbate the problem when they use the same login information for multiple sites, making it far easier for the attacker to obtain credit card data, bank account numbers, and more.[1][2] In fact attempts to steal data from individuals is widespread and growing more sophisticated.[3] One obvious way to combat attackers is to use unique, strong passwords for every site. Another approach is to take advantage of multifactor authentication when available, which provides increased security for legitimate users and protection from hackers or malware, collectively referred to as adversaries, attempting to impersonate an individual.

"…attempts to steal data from individuals is widespread and growing more sophisticated."

Adversaries Use Multiple Attack Vectors

Adversaries are using multiple attack vectors to steal user credentials, including phishing, malware, and attacks on service providers. These attacks are compounded by password reuse across multiple service providers, a common tactic used by users to mitigate the user experience issues with passwords. Additionally password infrastructures require a fallback mechanism to handle lost or forgotten passwords, which is most often implemented as a set of security questions the user must answer. These questions are often guessable or attainable public data, enabling adversaries to easily bypass the passwords and gain control over a user's account by resetting the password. Similarly it is possible to gain control over a user's account by social engineering of the service provider's technical support staff, convincing them that the attacker is the account owner and resetting access through administrative interfaces.[4] Lastly, because users attempt to decrease friction on their mobile devices by removing passwords and enabling "remember me" options, physical theft of mobile devices becomes an increasing concern for account security.

"Adversaries are using multiple attack vectors to steal user credentials, including phishing, malware, and attacks on service providers."

Authentication with Passwords Is Increasingly Difficult to Perform

Users have an increasingly wide variety of devices and mechanisms they use to interact with their valuable services, ranging from classic computing devices such as desktop and laptop computers, to smartphones and tablets with

"…we should be able to take advantage of increasingly rich device capabilities to authenticate users in a continuous manner…"

primarily touch interfaces, to set-top boxes with primarily remote control interfaces, to automobiles with very limited interfaces or interfaces that are further constrained under certain circumstances (such as driving). All of these impact whether a classic model of asking the user for a password is truly feasible for day-to-day use. We believe that we should be able to take advantage of increasingly rich device capabilities to authenticate users in a continuous manner thereby eliminating the traditional password in most cases.

Current Authentication Approaches

The challenges in authentication can be broken down into three primary questions:

1. Who is present at the time of initial access request?
2. Is that person still present at a later point during a transaction?
3. How can a remote service know about the local user?

Passwords combined with login timeouts are the key mechanisms used today to protect multiple types of access:

- Device access (example: OS login/lock)
- Data access (examples: disk encryption, app-specific data encryption)
- Service provider account access
- Transaction intent (examples: banking transaction confirmation, device security policy changes), which is typically a cognitive interrupt for an already authenticated user

Each of these access types has its own set of usability and security challenges, and combined they frustrate the user constantly. We want our solution to enable the security required for the most common uses along with the user experience required for frequent access throughout the user's day, and only occasionally requiring additional authentication.

Passwords—What You Know

The user ID and password has long been the default standard for computer security. Passwords were first used at MIT in 1961 for access to the Compatible Time-Sharing System (CTSS) to give each user their own private set of files. "Putting a password on for each individual user as a lock seemed like a very straightforward solution."[5] As the Internet took off in the 1990s, passwords still worked fairly well because there was not much personal data that needed protecting. Fast-forward to today, and active Internet users can encounter 25 password-protected sites a day that contain a myriad of personal information. "The reality is, we have a system that not only is insecure but it's totally unusable."[6]

"The primary challenge of passwords today is usability."

Challenges

The primary challenge of passwords today is usability. There are far too many passwords to remember and increasingly complex policies are outpacing human memory capacity.[7] An ideal password system for security purposes would

require long and complex passwords, different passwords for every service, frequently changing passwords, and entry of the password for every transaction (especially high-value transactions). In contrast, the ideal password system for user experience would require the user to remember at most one thing, be easy to remember and never change, be entered infrequently, and be easy to enter from any user interface (and thus limiting the alphabet of available characters to use). These user experience desires are what leads to weak security as the aspects of passwords that makes them easy to remember and enter are the same aspects that make them guessable to an attacker. Similarly the security demands of complexity and low guess ability lead to user frustration and difficulties even creating passwords that are acceptable to the system, much less memorable.[23]

Coping Mechanisms

Coping mechanisms that people use to remember passwords can compromise security and degrade usability at the same time. Often users will write down their passwords on paper or cards, leaving them pasted to their monitor or kept with their computer, or resident in their wallets. These techniques expose the user to account compromise by individuals with physical access to the device and can complicate account protection when a device is stolen. They also put the user at risk of losing access should they misplace the paper they wrote the password on.

"Coping mechanisms that people use to remember passwords can compromise security and degrade usability at the same time."

Another coping technique is to keep a list of passwords in a file on their computer, similar to a manual password manager. This technique leaves the user greatly exposed should they get malware on their computer or accidentally expose the file. It is also not very convenient from a usability perspective to copy and paste individual passwords whenever login is needed.

Perhaps the best-known coping mechanism is to choose an easy-to-remember password, and studies have shown this is extremely common. The company SplashData conducts an annual study based on the millions of passwords that are stolen and posted online each year. In 2013, the #1 password on the list was "123456", followed closely by "password" in the #2 spot.[9]

"In 2013, the #1 password on the list was "123456", followed closely by "password" in the #2 spot."

To help people manage their increasingly complex passwords, many entities have developed password managers or vaults. Password managers attempt to address the user experience challenges of traditional passwords by maintaining a database of user's passwords and automatically plugging them in for each service provider. The database is usually protected by a single master password or secure token.

Examples include LastPass, KeePass, 1Password, RoboForm, SafeKey, Password Box, KeyLemon, KeyChain, and MyIDKey.

While they can be helpful, password managers have traditionally had integration challenges because they typically do not address all of the environments where passwords are required. The password managers usually handle passwords in the domain in which they reside such as browsers, but not for plug-ins, apps or OS infrastructure. In addition, the password manager

"Password managers... do not address all of the environments where passwords are required."

itself can become the target of security attacks, such as the suspected LastPass compromise.[8] Elcomsoft performed an evaluation of common password managers uncovering several implementation flaws.[24] Fundamentally a password manager changes the authentication mechanism from "something you know" (the password) to "something you have" (control of the password manager).

Tokens—What You Have

Tokens are physical devices that the authorized user must present to the system in order to be authenticated to that system. Examples include smartcards, USB dongles, NFC tags, automobile key fobs, and even software instantiations on secondary devices such as an app on the user's smartphone that must be present to access their account on a primary device. Tokens are a useful physical world analogy, since most users are familiar with tokens in the form of physical keys used for door locks and other traditional physical locks. Tokens with no second factors only represent that the individual accessing the system bears the token, and do not actually identify the individual accessing the system.

"Tokens are a useful physical world analogy,…"

Challenges

The aspect that tokens do not identify the bearer, only that the token is present, leaves standalone tokens vulnerable to simple theft or loss. From a user experience perspective the tokens are frequently subject to loss or to simply being forgotten in the wrong location, leading to denial of access to legitimate users. In addition many of these tokens have difficulties with the infrastructure required in newer operating systems or service providers in order to allow access, and they add nontrivial costs, relegating them to be niche point solutions rather than a general authentication mechanism that could be used to replace passwords.

Coping Mechanisms

Due to the security issues surrounding token theft, the tokens are often combined with a second factor such as a PIN or biometric that must be used to "unlock" the token prior to it being used. These solutions mitigate the security threats to some degree but at the expense of the ease of use provided by a standalone security token. The user experience issues with tokens, such as forgetting them or losing them frequently leads users to leave the tokens permanently attached to the device they are intended to secure, leading to significant decrease in the security offered by the device.

"Out-of-band two-factor-systems... allow the user to submit a second authentication credential made available through one of their other devices."

Out-of-Band Two-Factor Systems

Out-of-band two-factor systems (or commonly just two-factor authentication, 2FA) focus on a mechanism to allow the user to submit a second authentication credential made available through one of their other devices. Common mechanisms include SMS-based one-time passcodes or mobile application one-time password generators. The security of these mechanisms is dependent on the attacker not having access to the secondary devices that the secret challenge is delivered to or generated by.

Challenges

The security of 2FA systems is completely dependent upon the security of the receiving or generating device and the synchronization of that device with the authentication service that is challenging the user. In the event the secret shared between the second factor and the verifier is compromised, the second factor can be fully emulated by an attacker such as what happened to the RSA SecurID.[25] More recently a new more convenient 2FA system for users has emerged that sends a security code to the user's registered phone via SMS. This may be generally good enough for average users but attacks have been demonstrated against the SMS infrastructure by social engineering the phone provider to change the user's phone number to an attacker-controlled SIM card, allowing them to receive the secret code.[31] Implementation differences make it more difficult to characterize the security model of mobile app-based one-time password (OTP) or security code systems. From a user experience perspective these systems are relatively disruptive to use as they must challenge the user and the user must either wait for a code to be delivered to their device or generate a code in a separate application than the one they were using during the challenge. As with token systems, 2FA is susceptible to device loss or theft, inconveniencing the user and requiring a recovery mechanism, as well as potentially providing the second factor to an attacker. For more information on 2FA security, see the article in this edition titled "Security Analysis of Mobile Two-Factor Authentication Schemes."

"…a new more convenient 2FA system for users has emerged that sends a security code to the user's registered phone via SMS."

Coping Mechanisms

The main coping mechanism for the user experience difficulties with 2FA is to utilize it primarily for initial login and authorization of trusted devices and/or applications. Many of the 2FA instances on twofactorauth.org implement this strategy, where a browser, app, or device is authorized for further access with only single-factor authentication (or none), with 2FA being used only for initial login.[35] This strategy balances usability and security, though it leaves the 2FA implementation vulnerable to cookie or other secondary factor theft from the trusted devices.[36]

The security issues around 2FA are being worked out by the industry, with telecommunications companies working to lessen the threat of SMS reissuance through social engineering and app best practices being deployed for development of OTP applications on target devices. In addition, hardware-bound 2FA solutions such as Intel IPT can provide a much more robust implementation of 2FA.[34]

Biometrics—What You Are

Biometrics is the measurement and statistical analysis of biological data.[33] Authentication based on biometrics identifies or verifies an individual based on distinct physiological and/or behavioral characteristics. For example, physiological authentication includes distinctive facial features, hand geometry, and fingerprints. Behavioral biometrics include how an individual walks (gait), how they type a phrase on a keyboard, how they operate a mouse, and others.

"Authentication based on biometrics identifies or verifies an individual based on distinct physiological and/or behavioral characteristics."

"…a biometric authentication system depends on a confidence or matching measurement…"

Challenges

Biometrics faces a number of challenges, which can be separated into user experience and security impact areas. Unlike a password or security token system, biometric authentication is dependent on probabilistic recognition to determine whether the biometric data presented belongs to an authentic user. The system can be either a verification (that the presented biometric data belongs to a specific user) or identification (of the presented biometric data by searching through a database of valid users). Thus a biometric authentication system depends on a confidence or matching measurement (for example: the presented biometric data belongs to user X with 97 percent confidence) rather than an objective true or false value as in password or security token systems. The confidence/matching measurements may falsely authenticate an invalid user or falsely reject a valid user. The rate of these errors must be minimized in order to have a reliable biometric system. Assuming that the biometric is universal, distinctive, and permanent, there are a number of other user experience issues. The collectability and acceptability of the biometric relate to how difficult the biometric is to use and whether users will view the biometric as a positive technology. Social acceptance issues such as the fear of surveillance or religious or vanity issues with parts of the body can impact the acceptance of a biometric.

In addition biometrics are susceptible to attacks. Besides the false accept rate (which defines the rate at which a biometric will accept an imposter as a genuine user), biometrics are also susceptible to spoofing attacks, which can be measured by their spoof resilience and retainability of the physiological characteristic. For example, fingerprints are left on many surfaces that the user touches, leaving them susceptible to being captured (and hence have low retainability). However depending upon the liveness and anti-spoofing technology in a fingerprint scanner the technology may be able to distinguish between a legitimate user-presented fingerprint and an imposter spoofed fingerprint.

"If captured, unlike a password or a token, a given user cannot simply change or replace their biometric data."

Given the potential for an attacker to capture the raw biometric data from a user, such data is considered extremely sensitive. If captured, unlike a password or a token, a given user cannot simply change or replace their biometric data. For this reason biometric data and stored biometric templates must be protected.

Coping Mechanisms

As biometrics continues to be an emerging technology solution to authentication, biometric techniques are often deployed as an optional or secondary technology. So for users who are experiencing reliability difficulties with biometrics, the coping mechanism is to simply disable the biometrics.

From a security perspective there are technology solutions for some of these risks. For biometric template protection, the most common mechanism is to

encrypt the templates to a given system, rendering them unusable to other systems. Emerging techniques such as template salting or noninvertible transforms are promising for template safety in the event of a theft, but remain potentially detrimental to the quality of the biometric matching.[26] Liveness detection and anti-spoofing remain difficult areas for biometrics with many biometric systems still vulnerable to spoof attacks performed by skilled attackers.

"Federation is a mechanism that can reduce the burden on the user to remember individual account passwords,…"

Federation

Federation is a mechanism that can reduce the burden on the user to remember individual account passwords, instead requiring that they remember only a single password for the Identity Provider (IDP) and registering their individual service provider accounts with that IDP. Figure 1 illustrates the concept that federated identity is a collaboration between a user, an identity provider, and a relying party. A user, which might be an individual or non-person entity, works through the identity provider to obtain a credential that will satisfy the relying party to provide user access. Depending on the security level required by the relying party, the identity provider may have to collect multiple factors from the user before issuing a credential. Because there is a prearranged trust relationship between the identity provider and the relying party, the relying party will accept the credential generated and admit the user.

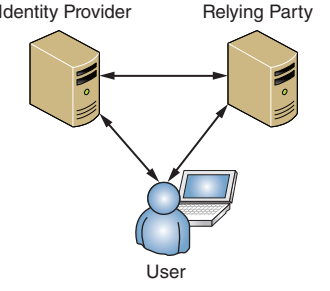


Figure 1: A user collaborates with an identity provider to obtain credentials to access a relying party service (Source: Intel Corporation, 2014)

Multiple standards have emerged that provide a foundation for federated identity solutions including SAML[37], OpenID[38], and Oauth[39]. See the section "Challenges and Opportunities Ahead" for more details on these standards.

Challenges

The federated ID approach has promise for widespread adoption but has suffered from business pressures such as competition between service providers over identity, low service provider adoption due to concerns over controlling their customer database, and security and privacy issues. Despite these challenges, the federation model has been gaining support more recently due to the success of Google Account[40] and Facebook Connect[41], with others such as Microsoft Live accounts also gaining traction. However, this also leads to service providers needing to support multiple IDP services and user confusion when they have to choose between multiple IDPs for a given service provider.

Coping Mechanisms

As federated identity solutions are intended to replace a reliance on passwords, users typically don't develop a coping mechanism because they are often given a choice for how to authenticate. Typically a user will enter a single master password and then choose between receiving a text, a call, or an email that contains a second factor for them to enter. Given this strong authentication, the user can use this credential to access multiple enabled sites without having to re-authenticate.

Vision: Strong Multifactor Authentication that Is User Centric and Continuous

"Multifactor authentication is the combination of different authentication factors into a single authentication decision."

Multifactor authentication is the combination of different authentication factors into a single authentication decision. Each of the factor types has class-level threats and user experience challenges:

- What you know—shareable, guessable, difficult to remember if complex enough to prevent guessing
- What you have—can be stolen, implementations can be broken into, lost, forgotten, damaged
- What you are (biometrics)—can be compelled involuntarily, many are spoofable, acquisition challenges, environmental influences, privacy concerns and perceptions, false rejects

Multifactor authentication attempts to address these challenges by combining the factors to mitigate the risks of individual factors.

"The combination of multiple factors increases authentication confidence."

Factor Fusion

The combination of multiple factors increases authentication confidence. For example, successful face authentication on a system with a registered device ID logged into a trusted network offers much higher confidence than any of the individual factors. Combining, or fusing multiple factors, requires thoughtful consideration of the weight or influence of individual factors and how the combination of weights yields a single authentication decision at any given moment.

Current Popular Approaches to Multifactor Authentication

The following sections describe several current approaches to multifactor authentication.

Device ID

Device ID technologies allow a service provider to confirm cryptographically that a given device is being used for the current user session. If a service provider uses this to restrict the security domain to a small set of user devices this lowers the dependency on the primary authentication factor (such as a password) for security. The adversary must typically provide a much stronger identity proof or they must find a way to perform malicious transactions from one of the user's devices via malware or user misdirection.

Two-Step Verification—Out-of-Band OTP

As indicated earlier, 2FA is a rapidly emerging technique for multifactor authentication using today's infrastructure. The fastest growing mechanism is to use a user's mobile device to deliver the second factor confirmation code via SMS or push notifications. For more information on 2FA, refer to the article entitled "Security Analysis of Mobile Two-Factor Authentication Schemes" in this issue of the *Intel Technical Journal*.

Biometrics

Also emerging is the addition of biometrics to enable multifactor authentication. To-date implementations for consumers and enterprise have typically favored face recognition, voice recognition, and fingerprints as the sensing technologies for those factors are widely available in existing devices.

Emerging Improvements to Multifactor Authentication

The following are emerging improvements for multifactor authentication.

Further Biometric Improvements or New Biometrics

Unsupervised facial recognition has traditionally had challenges with spoofability and false reject rates, but new advances in 3D and infrared facial recognition promise to address some of these shortcomings. With 3D facial recognition the security bar is raised higher for an adversary as two-dimensional biometric data that is readily available in photos is no longer sufficient to spoof the matching algorithm. Many of the algorithms also are improving upon rejection rates by allowing multiple angles and having many more features to match, allowing only a portion of the face to be matched while improving accuracy. For more details on leading edge face recognition approaches, see the articles in this issue entitled "Heterogeneous Face Recognition: An Emerging Topic in Biometrics" and "On Designing SWIR to Visible Face Matching Algorithms."

"Unsupervised facial recognition has traditionally had challenges with spoofability and false reject rates,…"

Another strong biometric technology is iris recognition, but the technology typically relies upon the use of infrared imaging and lighting. As these technologies are incorporated into consumer devices for depth camera usages it opens up the possibility of bringing this strong biometric to common use. The challenge will be to ensure the biometric is enabled with good liveness and anti-spoofing, because iris patterns may be relatively easy to acquire by an adversary.

Vein recognition technologies (such as palm vein, finger vein, retina, or face veins) are considered very strong biometrics from a security perspective. With focus on miniaturizing and lowering the power consumption of these devices, veins may become a competitor to fingerprint for consumer touch-based biometric usages. A strong advantage vein technologies have over fingerprint is that the biometric data is not left imprinted on the devices or everyday objects as fingerprints are, and thus are less available to the adversary.

"…veins may become a competitor to fingerprint for consumer touch-based biometric usages."

Behavioral biometrics such as typing and touch patterns are emerging as a good passive verification technique while the user performs actions on their device. Typing dynamics are able to passively confirm the user while they type normally, which can be used to bolster confidence that the user is still present without forcing them to provide a biometric or other authentication factor. Similarly, touch dynamics can be used to passively confirm the user while they touch or swipe the screen during normal use, again without causing an interruption or change in behavior on the part of the user.

Gait recognition using motion sensors in a modern mobile device is another promising area of emerging biometrics. Gait recognition has the potential to allow a device that is with the user to maintain confidence that it is with the same user while the user travels from one location to another. For instance, it could be used to maintain a login session while walking from an office to a conference room or for the duration a smartphone is in the user's pocket while walking around town. While gait is not yet a strong biometric in the matching sense, it can be combined with a strong initial factor to be used only to maintain a session, rather than to initiate one.

Paired Device Proximity

One convenient factor that can be considered in an authentication decision is whether another trusted device that belongs to the user is present. This could be the user's phone, tablet, Bluetooth Low Energy tags, NFC tags, and many other emerging wireless devices. With each device that must be present the attacker must work harder to piece together the correct context for an attack to succeed.

Wearables

"Wearable computing provides a powerful specific instance of device proximity that can be done with very low friction to the user."

Wearable computing provides a powerful specific instance of device proximity that can be done with very low friction to the user. Wearables can function in one of four possible ways in an authentication system:

1. What you have—wearables can function as a what-you-have factor, leveraging wireless technologies to lower friction to the wearable and providing greater likelihood that the device will be with the user by being a part of their typical routine.
2. Presence factor or proximity factor—wearables can also provide a potential strong presence factor, useful for lowering the friction for further authentications after an initial strong authentication. Given wearables have the potential to be strongly attached to the user, they represent more accurately that the same person is present than a phone or tablet.
3. Biometrics—many emerging biometric technologies, such as ECG/EKG or bioimpedence, are naturally aligned with wearables, potentially allowing for a completely frictionless authentication factor in the future that is passive to the user.[27,28]
4. Alternative knowledge proof—wearables also create a new space for knowledge proofs for users, such as custom gestures, new user interfaces, and behavioral biometrics.[29,30]

For more information on how biometrics can be combined with wearables, see the article entitled "A Survey of Biometrics for Wearable Devices."

Seamless and Continuous Authentication

Putting all these emerging technologies together we'll now cover our vision of the best combination of user experience and security for authentication.

Continuous and Passive Authentication

"The growing variety of authentication methods provides opportunities to authenticate continuously and passively."

The growing variety of authentication methods provides opportunities to authenticate continuously and passively.

Many factors can be integrated into a context aware agent that continuously collects and evaluates individual factors. Factors such as paired device proximity, face, voice, and gait can be collected passively and opportunistically without interrupting the user, thus maintaining high confidence that the authenticated user is in fact interacting with or near the device.

Continuous Presence

Going beyond continuous authentication, we envision that various human presence techniques can be used to extend authentication confidence even when the update of individual factors is not authenticating to a specific person, that is, a continuous presence model. This allows us to incorporate simpler and more power-efficient sensors into our authentication model, such as infrared proximity detection, face detection, and keyboard/mouse/touchscreen activity, extending the confidence of an earlier authentication event. For more information on user-passive authentication, see the article in this edition titled "Estimating the Utility of User-Passive Authentication for Smartphone Unlock."

"…various human presence techniques can be used to extend authentication confidence…"

Implementing Step-Up Authentication

In an ideal model the user would always authenticate using the strongest possible authentication factors. However, for the foreseeable future we still believe the most secure factors will remain inconvenient for the user to employ, and hence they will not want to use them on a regular basis. In our model we include the possibility of implementing step-up authentication, which allows the user to authenticate initially at a certain level of security that is convenient for them for their most common usages. Should the user need to access a service that requires a heightened level of security, they can be prompted to authenticate at the higher level at that time. From that point forward the device will maintain that higher confidence level using passive or continuous presence factors.

Adding a Policy Engine

Once we have a set of active, passive, and presence factors available to the device, we are able to perform authentication and presence monitoring. However we must combine this with a policy engine that can determine whether the requirements for a given authentication request have been met. This policy engine can reside locally on the device inside a trusted execution environment such as Intel Software Guard Extensions, or it can reside remotely on an identity service (see the article "Trusted Execution Environment for Privacy Preserving Biometric Verification" in this issue of the *Intel Technical Journal*). The role of the policy engine is to match the authentication requirements of the service provider to the capabilities of the device. We envision the policy engine should also incorporate user requirements, in order to satisfy user policies such as restrictions on which factors to use or explicit requests not to provide identity information to certain service providers. Once the requirements of the user and service provider are met, the policy engine is able to generate an assertion of user identity to the relying party in order to grant access.

"The role of the policy engine is to match the authentication requirements of the service provider to the capabilities of the device."

The Ideal User Experience

The combination of these ingredients enables the best possible user experience and security combination: to separate the authentication of the user to the

"While that confidence remains high enough, the device or identity service can continue to provide strong assertions of the user's identity…"

device from service provider authentication requests. By doing so the user is able to authenticate actively only once, and then the device will maintain confidence in that authentication for an extended period of time. While that confidence remains high enough, the device or identity service can continue to provide strong assertions of the user's identity to relying service providers as the user interacts with them. The identity service will immediately cease to provide assertions of identity as soon as the device is separated from the user, and optionally can notify service providers of the user's absence. The use of trusted execution environments and technologies in client devices will enable this experience while alleviating user privacy concerns associated with the increased use of sensors.

Challenges and Opportunities Ahead

The multifactor authentication approach, described above, promises to simplify personal access to information and services, freeing people from the burden of passwords. Successfully achieving broad adoption will require industry standards, mechanisms to ensure privacy and security, and ultimately systems that people can trust and that deliver a great user experience.

Standards and Initiatives

As the need for secure and easy authentication has grown, so has the level of interest in defining interoperable standards. While some standards had already emerged prior to 2001 (such as fingerprinting), the events on 9/11 jumpstarted standardization efforts in the interest of national security.[10]

National Institute of Standards and Technology (NIST)

In 2001 the National Institute of Standards and Technology (NIST), formerly known as the National Bureau of Standards (NBS), was given the mandate to accelerate biometric standards definition. They had already defined the standard for fingerprint encoding in 1986 (ANSI/NBS-ICST 1-1986) and have updated this specification with many new revisions since then, adding traditional encoding standards for face, iris, DNA, and other biometrics. A later revision defined XML encoding.[11] The current version of the specification is ANSI/NIST-ITL 1-2011. NIST has also authored SP800-63, the Electronic Authentication Guideline, which covers the remote authentication of users interacting with government IT systems over open networks. It defines technical requirements for identity proofing, registration, tokens, management processes, authentication protocols, and related assertions.[32] The United States agencies that use these specifications are the Department of Homeland Security, the FBI, the Department of Justice, the Department of Defense, and the intelligence agencies. The standards have been critical to foster the open exchange of biometric data between agencies and to ensure interoperability. Any company providing biometric solutions to the United States federal government are required to apply these standards.

"The standards have been critical to foster the open exchange of biometric data between agencies and to ensure interoperability."

Federated Identities: SAML, OAuth, and OpenID

While NIST was leading the definition of Biometric standards, other industry consortia, such as the Organization for Advancement of Structured Information Standards (OASIS) were developing standards to implement federated identities. Similar to an enterprise single-sign-on (SSO) implementation, federated identities link the identities that a user has across several identity management systems. When a relying party can use an authentication from an identity management system that it trusts, the relying party is freed up from managing a set of credentials (such as user ID and password) for every user. The three dominant protocols for federated identity are SAML, OAuth, and OpenID.

"…federated identities link the identities that a user has across several identity management systems."

The Security Assertion Markup Language (SAML) is an XML-based standard for exchanging authentication data between parties, typically a user, a service provider (SP), and an identity provider (IdP). A user requests a service from the SP, the SP requests and obtains an assertion from the IdP. To make the assertion, the IdP authenticates the user, although SAML does not specify the method of authentication. Once the assertion is given, the SP decides whether to grant access. SAML 1.0 was published in 2002 and SAML 2.0 became an OASIS standard in 2005.

OAuth is also an open standard but is intended to provide authorization as opposed to authentication. A typical scenario is when a user logs into a service using "Login with Facebook." In the background, an identity provider generates a limited scope OAuth token that authorizes the service to access the user's Facebook data. Thus a user can authorize third-party access to their Facebook resources without having to share their Facebook credentials. OAuth 1.0 was published in April 2010 and revision 2.0 in October 2012.

OpenID is an open standard defined by an industry consortium called the OpenID Foundation. Their authentication standard is called OpenID Connect and is essentially an identity layer built on top of OAuth 2.0. With OpenID, a user can establish an account with an identity provider and then use that identity provider to access any web resource that accepts an OpenID authentication. As with SAML, a variety of authentication mechanisms can be used.

National Strategy for Trusted Identities in Cyberspace (NSTIC)

While NIST standard described above is primarily aimed at federal agencies, the National Strategy for Trusted Identities in Cyberspace (NSTIC) addresses authentication solutions in the private sector. Established in 2011, NSTIC is a White House initiative to foster collaborative efforts between the private sector, advocacy groups, public sector agencies, and other organizations to improve the privacy, security, and convenience of online transactions.[12] The realization of the strategy vision is an "Identity Ecosystem" that follows four guiding principles:

"…the National Strategy for Trusted Identities in Cyberspace (NSTIC) addresses authentication solutions in the private sector."

- Identity solutions will be privacy-enhancing and voluntary
- Identity solutions will be secure and resilient

- Identity solutions will be interoperable
- Identity solutions will be cost-effective and easy to use

Three years after being launched, NSTIC is making headway, with multiple pilots underway that are reporting positive results.[13]

Fast Identity Online (FIDO)

The FIDO Alliance is an industry consortium formed in July 2012 to address the lack of interoperability among strong authentication devices. Their mission is to "change the nature of online authentication" by defining technical specifications that will eventually be recognized as standards and by setting up industry programs that adopt the specifications and standards.[14] To date FIDO has released two specifications, the *Passwordless User Experience* (UX) based on a Universal Authentication Framework (UAF) and the *Second Factor Experience* based on the Universal Second Factor (U2F) protocol. Some of the members of FIDO include PayPal, Google, Microsoft, and Lenovo. PayPal and Samsung recently announced a collaboration that enables Samsung Galaxy S5 users to make payments online using the Galaxy S5 fingerprint reader, thus becoming FIDO's first authentication deployment.[15] FIDO relies on the foundation standards published by the OASIS committees and considers their methodology to be complementary to OpenID.[16]

"Multifactor authentication, more specifically the gathering and storage of biometric data, raises serious privacy concerns."

Privacy Concerns and User Acceptance

Multifactor authentication, more specifically the gathering and storage of biometric data, raises serious privacy concerns. "Biometric technologies don't just involve collection of information *about* the person, but rather information *of* the person, intrinsic to them."[17] This can be threatening to people because that information can be used to control them by monitoring their location, monitoring their activities, limiting access to services, and basically denying them the freedom of anonymity. "The explosion of computers, cameras, sensors, wireless communication, GPS, biometrics, and other technologies in just the last 10 years is feeding what can be described as a surveillance monster that is growing silently in our midst."[18] Where broad use of biometrics would arguably make governments more efficient and cost-effective, it also gives them tremendous power over individuals as in George Orwell's *1984*. As people are wary of this, any solution proposing to make their lives easier and "password-free" through biometrics will have to establish a very strong trust relationship. Users must be convinced that their biometric data will be kept secure, will be shared with no one, and will only be used for the authentication mechanisms for which they opted in. Users will also want the ability to opt out at any time and have their biometric records permanently deleted.

"Users must be convinced that their biometric data will be kept secure,…"

Technologies may help allay user's fears to some extent. For example, Apple's new Touch ID feature protects a user's encrypted fingerprint data in a "secure enclave."[19] Other techniques include biometric template salting, one-way transforms, and match-on-device.[26] These are all designed to thwart attempts to access the biometric information for anything other than the intended purpose.

Broad user acceptance will also depend on the kinds of biometrics that are employed. Users have varying levels of trust based on the method used. Table 1 shows survey results of 1000 biometric users and nonusers on their perceptions about which biometrics are most effective at securing their personal information.

Type of Biometric	Biometric user mean (sd)	Biometric nonuser mean (sd)	Total mean (sd)
Facial recognition	3.60 (1.13)	3.93 (1.06)	3.71 (1.12)
Hand geometry	3.76 (1.06)	3.92 (1.21)	3.82 (1.11)
Gait recognition	2.45 (1.10)	2.78 (1.28)	2.56 (1.17)
Voice recognition	3.60 (1.05)	3.27 (1.26)	3.48 (1.13)
Fingerprints	4.45 (0.86)	4.31 (0.90)	4.40 (0.87)
Key stroke recognition	2.53 (1.25)	2.27 (1.40)	2.44 (1.29)
Signature recognition	2.20 (1.18)	3.13 (1.30)	2.53 (1.29)
Iris/retinal scans	4.45 (0.83)	4.68 (0.54)	4.53 (0.75)
Group means	3.38 (1.06)	3.54 (1.12)	3.43 (1.10)

Note: Ratings were collected using a Likert scale from 1 to 5. 1 = Not at all safe, 2 = Somewhat safe, 3 = Neither safe nor unsafe, 4 = Somewhat safe, and 5 = Very safe.

Table 1: Comparison of User Perceived Safety when Using Various Biometric Methods [20] (Source: America Identified: Biometric Technology and Society, 2010)

Average ratings assigned by biometrics users and nonusers: "How safe do you feel each of the following types of biometrics is as a way to protect your personal records from access by unauthorized persons?"

Fingerprints and iris/retinal scans are perceived as fairly safe, followed by hand geometry and face recognition. Refer to the article "Biometrics from the User Point of View" in this issue of the *Intel Technical Journal*, to learn more about user acceptance of various methods and cover design principles.

"Fingerprints and iris/retinal scans are perceived as fairly safe,…"

Industry Adoption and Enabling of MFA

According to IDC, the size of the identity and access management market in 2010 was nearly USD 4 billion.[21] Another source estimates "The global multi-factor authentication (MFA) market which includes different types of authentication and applications is expected to reach $5.45 billion by 2017 at an estimated CAGR of 17.3 percent from 2012 to 2017."[22] This is clearly an area of significant technology investment with a healthy growth rate. With this amount of investment and rate of growth, how will we avoid creating a hodge-podge of solutions vying for attention from users and relying parties and with enrollment options that confound the problem rather than simplifying it? The same IDC report provided this essential guidance: "Vendors and their ecosystem partners need to collaborate ... to effectively solve the current and emerging issues in digital identity." One of these big issues is the call for standardization and trusted frameworks. It is in our collective best interest for solution providers to adhere to and provide interoperable solutions based on standards such as those being developed by NSTIC and FIDO. Since these standards don't dictate the kinds of factors

"…the size of the identity and access management market in 2010 was nearly USD 4 billion."

or how they are collected, there is still a tremendous amount of room for innovation. Users will have a choice of vendor solution and still be assured that they will be able to reach most, if not all, of their favorite internet services.

Summary

"…passwords alone are unusable as a security method for our online identity needs."

"What you know, what you are, and what you have appropriately combined offer far more powerful assurance to service providers that you are who you claim to be."

It has become painfully obvious that passwords alone are unusable as a security method for our online identity needs. There are far too many to remember, the rules are too complex, and the ability of adversaries to obtain and exploit them is growing stronger. Multifactor authentication, particularly the use of biometrics, promises to be a much better alternative. We described a vision that takes advantage of our increasingly capable and sensor-rich computing devices that not only simplifies factor collection and processing with an improved user experience, but also raises the security assurance for relying parties. *What you know*, *what you are* and *what you have* appropriately combined offer far more powerful assurance to service providers that you are who you claim to be. Combining a multifactor authentication approach with a federated identity ecosystem should make it much more convenient and cost effective for relying parties, freeing them from managing passwords, biometrics, endpoint devices, and step-up authentication approaches. Broad acceptance and adoption of this solution will have challenges that include advances in biometric technologies, user acceptance and trust for biometrics, common standards for federated identity across the ecosystem, and cooperation among industry leading identity provider solutions. However, these challenges are manageable and the level of research and investment leaves us with the hope that the technologies will evolve quickly and improve the authentication experience for everyone.

References

[1] NBC News, "Password Hackers Propel Identity Theft," 2005, http://www.nbcnews.com/id/8408391/ns/nbc_nightly_news_with_brian_williams/t/password-hackers-propel-identity-theft/#.U2HEKvldV8E.

[2] Winstead, Charles L., "Identity Theft," National Security Cyberspace Institute, http://www.nsci-va.org/WhitePapers/2010-12-08-Identity%20Theft-Winstead-final.pdf.

[3] KPMG International, "Cyber Crime—A Growing Challenge for Governments," 2011, https://www.kpmg.com/Global/en/IssuesAndInsights/ArticlesPublications/Documents/cyber-crime.pdf.

[4] Rowat, Mohit, "Social Engineering: The Art of Human Hacking," Infosec Institute, 2013, http://resources.infosecinstitute.com/social-engineering-art-human-hacking/.

[5] McMillan, Robert, "The World's First Computer Password? It Was Useless Too," *Wired*, 2012, http://www.wired.com/2012/01/computer-password/

[6] Hiscott, Rebecca, "The Evolution of the Password—and Why it's Still Far From Safe," Mashable.com, 2013, http://mashable.com/2013/12/30/history-of-the-password/.

[7] Kemp, Tom, "The Problems with Passwords," Forbes.com, 2011, http://www.forbes.com/sites/tomkemp/2011/07/25/the-problems-with-passwords/.

[8] Raphael, JR. "LastPass CEO Explains Possible Hack," *PC World*, 2011, http://www.pcworld.com/article/227268/lastpass_ceo_exclusive_interview.html.

[9] SplashData.com, "Worst Passwords of 2013," 2013, http://splashdata.com/press/worstpasswords2013.htm.

[10] Tilton, Catherine, "Getting Started—Biometric Standards," Planet Biometrics, 2011, http://www.planetbiometrics.com/article-details/i/499/.

[11] NIST Standards History, ANSI/NBS-ICST 1-1986, 2014, National Institute of Standards and Technology (NIST), http://www.nist.gov/itl/iad/ig/ansi_standard-history.cfm.

[12] White House, National Strategy for Trusted Identities in Cyberspace (NSTIC), 2011, http://www.whitehouse.gov/sites/default/files/rss_viewer/NSTICstrategy_041511.pdf.

[13] Grant, Jeremy, "My heart bleeds for better identity solutions, my brain is excited by the progress," NSTIC Notes, 2014, http://nstic.blogs.govdelivery.com/

[14] FIDO Alliance, "About the FIDO Alliance," 2014, http://fidoalliance.org/about.

[15] Matick, Suzanne (FIDO Alliance), "The FIDO Alliance Announces First FIDO Authentication Deployment—PayPal and Samsung Enable Consumer Payments with Fingerprint Authentication on New Samsung Galaxy S5," Wall Street Journal, 2014, http://online.wsj.com/article/PR-CO-20140224-911048.html.

[16] FIDO Alliance, FAQ, 2014, https://fidoalliance.org/about/faq.

[17] Clarke, Roger, "Biometrics and Privacy," 2001, http://www.rogerclarke.com/DV/Biometrics.html#Thr.

[18] Nelson, Lisa S., *America Identified: Biometric Technology and Society* (The MIT Press, 2010), Quoted from Barry Steinhardt, director of the Technology and Liberty Project at the American Civil Liberties Union.

[19] Apple, "iPhone 5s: About Touch ID Security," 2014, http://support.apple.com/kb/ht5949.

[20] Nelson, Lisa S. America Identified: Biometric Technology and Society (The MIT Press, 2010).

[21] IDC, "Worldwide Identity and Access Management 2011–2015 Forecast: The Three Cs—Cooperation, Collaboration, and Commitment—Are Key for Identity-Driven Cloud," 2010.

[22] Semiconductor and Electronics, "Multi-Factor Authentication Market—By Model/Type [Two, Three, Four & Five-Factor], Application & Geography – Forecasts (2012 – 2017), *Research and Markets*, 2012, http://www.researchandmarkets.com/reports/2339811/multi-factor-authentication-market-by-modeltype.

[23] Komanduri, Saranga, Richard Shay, Patrick Gage Kelley, Michelle L. Mazurek, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Serge Egelman. 2011. Of passwords and people: measuring the effect of password-composition policies. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11), ACM, New York, NY, USA, 2595-2604. DOI=10.1145/1978942.1979321 http://doi.acm.org/10.1145/1978942.1979321

[24] Elcomsoft, "Secure Password Managers" and "Military-Grade Encryption" on Smartphones: Oh, Really?, 2012

[25] Bright, Peter, "RSA finally comes clean: SecurID is compromised," *Ars Technica*, June 6, 2011, http://arstechnica.com/security/2011/06/rsa-finally-comes-clean-securid-is-compromised/

[26] Teoh Beng Jin, Andrew and Lim Meng Hui, "Cancelable biometrics," Scholarpedia, 5(1):9201, (2010).

[27] Nymi, http://www.getnymi.com/.

[28] Cornelius, Cory, Jacob Sorber, Ronald Peterson, Joe Skinner, Ryan Halter, and David Kotz, "Who wears me? Bioimpedance as a passive biometric," In the *Proceedings of the 3rd USENIX Workshop on Health Security and Privacy* (HealthSec), Bellevue, Washington (August 6–7, 2012).

[29] Ring: Shortcut Everything, https://www.kickstarter.com/projects/1761670738/ring-shortcut-everything?ref=live

[30] Fin, https://www.indiegogo.com/projects/fin-wearable-ring-make-your-palm-as-numeric-keypad-and-gesture-interface

[31] Trusteer, SIM-ple: Mobile Handsets are Weak Link in Latest Online Banking Fraud Scheme, March 13, 2012, http://www.trusteer.com/blog/sim-ple-mobile-handsets-are-weak-link-latest-online-banking-fraud-scheme

[32] NIST Special Publication 800-63-1, *Electronic Authentication Guideline*, NIST, December 2011, http://csrc.nist.gov/publications/nistpubs/800-63-1/SP-800-63-1.pdf.

[33] Jain, A. K., A. Ross, and S. Prabhakar, "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, Jan. 2004.

[34] Intel Identity Protection Technology, http://www.intel.com/content/www/us/en/architecture-and-technology/identity-protection/identity-protection-technology-general.html.

[35] Two Factor Auth List, http://twofactorauth.org/

[36] Cookie Re-Use in Office 365 and Other Web Services, http://samsclass.info/123/proj10/cookie-reuse.htm.

[37] Security Assertion Markup Language, https://www.oasis-open.org/standards#samlv2.0.

[38] OpenID Specifications, http://openid.net/developers/specs/.

[39] The OAuth 2.0 Authorization Framework, RFC 6749, http://tools.ietf.org/html/rfc6749.

[40] Google Account, https://accounts.google.com/.

[41] Facebook Connect, http://www.facebook.com/help.php?page=730.

Author Biographies

Jason Martin is a staff research scientist in the Security Solutions Lab and manager of the Authentication Research Team at Intel Labs. He leads a team of diverse researchers with the goal of improving user experience and security together. Jason's interests include authentication and identity, trusted execution technology, wearable computing, mobile security, and privacy. Prior to Intel labs he spent several years as a security researcher performing security evaluations and penetration tests on Intel's products. Jason holds a BS in Computer Science from the University of Illinois at Urbana-Champaign. He can be reached at Jason.Martin@intel.com.

Anand Rajan is director of the Emerging Security Lab at Intel Labs. He leads a team of senior technologists whose mission is to research novel security features that raise the assurance of platforms across the compute continuum (*cloud to wearables*). The topics covered by his team span trustworthy execution

environment, mobile security, identity and authentication, cryptography, and security for emerging paradigms (IOT, wearables). Anand has been a principal investigator for Intel's research collaboration with academia, government, and commercial labs in the area of Trustworthy Platforms. Anand was an active member of the IEEE WG that crafted the P1363 (public-key crypto) standard. Anand and his team developed the Common Data Security Architecture specification, adopted as worldwide standard by The Open Group. His team was also instrumental on several security standardization efforts (for example, PKCS#11, BioAPI, UPnP-Security, and EPID). Prior to joining Intel in 1994, Anand was technical lead for the Trusted-UNIX team at Sequent Computer Systems and worked on development and certification of a TCSEC B1-level Operating System. He can be reached at Anand.Rajan@intel.com.

Bob Steigerwald joined Intel in 2006 and is a Program Manager in Intel's Safe Identity division, leading product development efforts in multifactor authentication. In his prior role at Intel, Bob led a team of engineers who worked with software vendors to optimize software for energy efficiency, authored "Energy Aware Computing", a book through Intel Press, and architected the Intel Technology Journal edition on Sustainable Intelligent Systems. Bob has extensive background in software product development, software engineering process improvement, and program management. He also spent 15 years in the US Air Force and was an Associate Professor of Computer Science at the USAF Academy. Bob holds a BSCS from the USAF Academy, an MSCS from the University of Illinois, an MBA from Rensselaer Polytechnic Institute, and a PhD in Computer Science from the Naval Postgraduate School. He can be reached at Bob.Steigerwald@intel.com.

BIOMETRICS FROM THE USER POINT OF VIEW: DERIVING DESIGN PRINCIPLES FROM USER PERCEPTIONS AND CONCERNS ABOUT BIOMETRIC SYSTEMS

Contributors

Pablo Piccolotto
Argentina Software Design Center
Intel Corporation

Patricio Maller
Argentina Software Design Center
Intel Corporation

"Understanding user experiences in the domain of biometrics requires looking beyond the hard data on the different factors performance and error rates."

The user experience of biometric authentication relies heavily on perceptions and judgments of users about the different factors in use. Even biometric factors in use and exploration for a long time, such as face recognition and fingerprint reading, have a halo of unfunded beliefs and personal perceptions about their security level, intrusiveness, and ease of use. Understanding user experiences in the domain of biometrics requires looking beyond the hard data on the different factors performance and error rates.

This article presents a survey-based analysis of user perceptions about security in the use of different biometric factors that are matched to the result of current research on actual security performance. In particular, we discuss factors whose perceived security is different from objective data and derive new research hypotheses in the field of user experience to account for the difference. The study involves the analysis of the surveyed population based upon their biometrics familiarity and the examination of variance on the answers.

Finally, new hypotheses are characterized as interaction design principles, whose application may influence the perception of security beyond data of actual performance. We focus on interaction style and experience journeys for a couple of reference implementations.

Introduction

This article summarizes the results of quantitative user research about personal perceptions in the area of biometric authentication focusing on robustness, security, and preferences of use.

Identity verification for access control, also known as one-to-one comparison or authentication, has been traditionally based on something that a person knows (PIN, password) or a something a person has (key, magnetic or chip card), but the rapid advance of technology is introducing biometrics into the mainstream. Biometrics are based on the principle of measurable physiological and behavioral characteristics such as fingerprint, facial characteristics, voice patterns, or even the way a person walks. Each method has advantages and disadvantages and some of them are more reliable, secure, easy-to-capture, and less intrusive than the others.[1]

"Each method has advantages and disadvantages and some of them are more reliable, secure, easy-to-capture, and less intrusive than the others."

This investigation will focus on the most used physiological biometric factors:

- *Iris recognition* is a technique that uses patterns of color and shape in the iris to confirm a person's identity. Iris scanning devices are not easily fooled

and this approach has demonstrated to be one of most accurate and secure authentication methods, but there are issues that affect these particular technologies. For instance, the sensors are costly and the eye must have a certain degree of lighting to allow the sensor to capture the iris. There is potential for failure when enough light is not available.

- *Facial recognition* is a technique that uses unique facial features to identify an individual. Even though this method is quite inexpensive to implement because most solutions use standard built-in cameras (or a relatively inexpensive webcam) to work, it has been demonstrated that face recognition systems can be easily fooled by the use of a printed photo, a mask, or even video of the person. Current solutions have trouble identifying people under poor lighting conditions and detecting liveness, a necessary condition to provide a competitive level of security.
- *Voice recognition* is a technique that uses a voice print to analyze how a person says a particular word or sequence of words unique to that individual. This method has two major drawbacks: enrollment and security. During enrollment, the speaker's voice is recorded and typically a number of features are extracted to form a voice print (also known as a voice template or model). This process may take several minutes and the uniqueness of the voice as compared to digital fingerprints and iris scans is not as accurate. However, in low to medium risk situations, the risk is acceptable, taking also into consideration the fact that it´s more affordable than other techniques because it only uses a microphone, and that it´s available in phone conversations.
- *Fingerprint recognition* is a technique that uses the distribution of the ridge endings and bifurcations on the finger to confirm a person's identity. This technique, which has been in use for many years by law enforcement and other government agencies, is regarded as a reliable unique identifier. On the other hand, it also has some drawbacks: setting up and enrolling the fingerprint has always been a cumbersome process, and once set up, fingerprint sensors don't always reliably read fingerprints. Too often, fingerprint scanners require users to make several swipes before the system recognizes the individual. Many solutions in the market have solved spoofing attacks and liveness detection by adding thermal sensors to their scans, but there are still many ways to trick the system.

"Current face recognition solutions have trouble identifying people under poor lighting conditions and detecting liveness, a necessary condition to provide a competitive level of security."

"The human brain struggles to retain more than seven numbers in short term memory. Adding letters, cases, and symbols makes a password even more difficult to remember."

The study also included perceptions about 8-character human-generated passwords, to calibrate biometric factors perceptions with a non-biometric authentication factor of mainstream adoption. This technique involves the use of a mixed-case password that is eight characters long and contains a numeral and a symbol, which has long been considered strong enough, even by many IT departments and most applications nowadays. After all, it is one of 6.1 quadrillion combinations, and would take a reasonably fast computer nearly a year to crack.

The human brain struggles to retain more than seven numbers in short-term memory.[2] Adding letters, cases, and symbols makes a password

even more difficult to remember and users tend to select words and names that have some personal meaning. In a recent study of 6 million actual user-generated passwords, the 10,000 most common passwords would have accessed 98.1 percent of all accounts.[3] The prevalence of common passwords makes it even easier for hackers to crack passwords. But even more worrisome than nonrandom passwords is password reuse. The average user has 26 password-protected accounts, but only five passwords, which means that an average user tends to use the same password for at least five different situations.[4]

There is extensive research on all five methods, although the perspectives from which they are analyzed vary from very technical, such as acceptance and rejection rates, to more subjective, such as attitudes or usability. To compare perceptions and objective data, this investigation will select the level of security of each method.

Technical Security and Circumvention Ranking

Security is a risk-management strategy that identifies, controls, eliminates, or minimizes uncertain events that can adversely affect system resources and information assets. A system's security requirements depend on the application's requirements (the threat model) and the cost-benefit analysis. Properly implemented biometric systems are an effective deterrent to attackers.[5] The most common threats are *spoofing* attacks, performed by an individual who attempts to forge the biometric trait. This is particularly easy when there is not a sophisticated way to detect liveness. Liveness testing is a critical aspect of biometrics that nobody seems to have gotten right so far. Liveness is unique to biometrics and sets a technological challenge that goes against its value proposition of convenience, because it frequently implies extra actions from the user or a higher interaction friction than introducing a traditional 8-character password.

"Liveness is unique to biometrics and sets a technological challenge that goes against its value proposition of convenience."

We will also use the term *circumvention*, which reflects how easily the system can be fooled using fraudulent methods.

The integrity of biometric systems (such as assuring that the input biometric sample was indeed presented by its legitimate owner and that the system indeed matched the input pattern with genuinely enrolled pattern samples), is crucial.[6] While there are a number of ways an offender may attack a biometric system, there are two very serious criticisms against biometric technology that have not been addressed satisfactorily:

"One potential problem with biometric factors is that they are not "secrets" in the way that passwords or tokens are."

- *Biometrics are not secrets:* One potential problem with biometric factors is that they are not "secrets" in the way that passwords or tokens are. This means that it could be possible for a hacker to present a photo or video to fool a facial recognition system, to present a wax cast of a fingerprint to a reader, or to play back a recording of a voice to a voice recognition system. It may even be possible to intercept the biometric data from the reader and replay it later, bypassing the biometric sensor. Integrity of

the reference template is core to building a secure biometric solution. But many people are concerned about privacy of biometric information, which implies that preserving the confidentiality of templates is also important.

- *Biometric patterns are not revocable:* Unlike a password, biometric characteristics such as fingerprints cannot be revoked or changed. This can pose a serious problem if a hacker successfully compromises the database housing the biometric credentials. Some biometric systems may deal with this challenge by uniquely distorting or transforming the biometric template when it is stored, and transforming or distorting the biometric in the same way during the match process. If a hacker compromises a fingerprint template database, users can then re-enroll and distinct templates can be generated by using a different distortion or transformation.

The industry has developed different ways to mitigate replay attacks, being liveness detection the preferred mechanism. For example, some voice recognition systems require users to authenticate by asking them to speak a series of random words, preventing them from using a previously recorded voice sample. Similarly, face recognition systems may attempt to detect blinking to ascertain that the image in front of the camera is not a photograph. Sophisticated fingerprint readers also measure heat or electrical conductivity to establish that the finger is "alive."

Previous research has shown that iris recognition is the most secure method reaching 262X better rates than fingerprint recognition[7] (the second factor in our list), followed by facial recognition, and then voice recognition. Table 1 establishes a comparison among the different factors.

"Previous research has shown that iris recognition is the most secure method reaching 262X better rates than fingerprint recognition (the second factor in our list), followed by facial recognition, and then voice recognition."

Method	Coded Pattern	Misidentification rate	Security	Application
Iris recognition	Iris pattern	1/1,200,000	High	High security facilities
Finger printing	Fingerprints	1/1,000	Medium	Universal
Facial recognition	Outline, shape and distribution of eyes and nose	1/100	Low	Low security facilities
Voice printing	Voice characteristic	1/30	Low	Telephone service

Table 1: Comparison list of different biometric factors
(Source: International Journal of Computer Applications, 2011)

Table 2 illustrates the crossover accuracy of each biometric method.[7]

Biometrics	Crossover accuracy
Iris scan	1:131,000
Fingerprints	1:500
Facial recognition	1:100
Voice dynamics	1:50

Table 2: Crossover accuracy of different biometric factors
(Source: International Journal of Computer Applications, 2011)

An interesting study[8] defines a list of seven attributes that are useful to compare any biometric factor. The study poses that as long as it satisfies the following requirements, any human physiological and/or behavioral characteristic can be used as a biometric characteristic:

- *Universality:* each person should have the characteristic.
- *Distinctiveness:* any two persons should be sufficiently different in terms of the characteristic.
- *Permanence:* the characteristic should be sufficiently invariant (with respect to the matching criterion) over a period of time.
- *Collectability:* the characteristic can be measured quantitatively.

However, in a practical biometric system (a system that employs biometrics for personal recognition), there are a number of other issues that should be considered, including:

"A practical biometric system should meet the specified recognition accuracy, speed, and resource requirements, be harmless to the users, and be sufficiently secure to various fraudulent methods and attacks to the system."

- *performance*, which refers to the achievable recognition accuracy and speed, the resources required to achieve the desired recognition accuracy and speed, as well as the operational and environmental factors that affect the accuracy and speed;
- *acceptability*, which indicates the extent to which people are willing to accept the use of a particular biometric identifier (characteristic) in their daily lives;
- *circumvention*, which reflects how easily the system can be fooled using fraudulent methods.

A practical biometric system should meet the specified recognition accuracy, speed, and resource requirements, be harmless to the users, be accepted by the intended population, and be sufficiently secure to various fraudulent methods and attacks to the system. Table 3 shows how each biometric factor ranks in those categories.

Biometric identifier	**Universality**	**Distinctiveness**	**Permanence**	**Collectability**	**Performance**	**Acceptability**	**Circumvention**
Face	H	L	M	H	L	H	**H**
Fingerprint	M	H	H	M	H	M	**M**
Iris	H	H	H	M	H	L	**L**
Voice	M	L	L	M	L	H	**H**

Table 3: Comparison list of different biometric factors using seven biometric attributes (H = High, M = Medium and L = Low)
(Source: International Journal of Computer Applications, 2011)

Methodology

The proposed methodology was designed to quantify users' perceptions on the use of biometric systems. To accomplish this objective, we developed a survey instrument for data collection. The aim of the survey was to explore the relationship between actual and perceived security of several biometric authentication methods.

"The aim of the survey was to explore the relationship between actual and perceived security of several biometric authentication methods."

Some of the research hypotheses reflected in the survey are:

- H1: Perceptions about security of biometric methods are aligned with objective data
- H2: Password-based authentication is perceived as a secure option
- H3: Less-known methods are perceived as less secure
- H4: The more critical a password is, then the more probable for a user to remember it

An online questionnaire was distributed among friends and family of employees of a software development company. Although there was no retribution for answering, participants entered a raffle for a symbolic prize (dinner for two). The sample size was $n = 166$, balanced in gender, and focused on ages 25 to 45.

Results

Results are summarized in the form of hypothesis.

H1: Perceived Security Ranking

Survey respondents ranked the presented factors from the most secure to the least secure supporting H1, with an alignment with the bibliographic review:

1. Iris recognition
2. Fingerprint
3. Facial recognition
4. Voice recognition

Figure 1 illustrates the orders or magnitude of security of each biometric method that does not correspond to the assertiveness of the survey answers (as it was referenced in the *technical security and circumvention ranking*).

In another level of analysis, biometric factors differ in an order of magnitude in security, which should make a ranking activity (if perceptions are actually aligned with reality) a trivial task. However, only 30 percent of the answers had the correct order, and 24 percent had more than one error. The error rate in the case of iris recognition compared to fingerprint raises to 45 percent, accounting as well for the lack of exposure to sophisticated methods.

"Only 30 percent of the answers had the correct order, and 24 percent had more than one error."

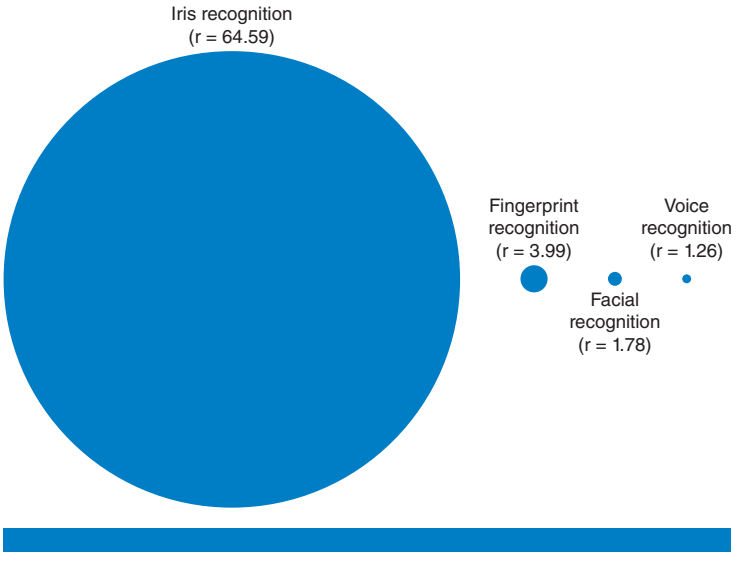


Figure 1: Comparative infographic of objective security amongst the four biometric authentication factors referenced in the study (r = radius).
(Source: Intel Corporation, 2014)

Taking into account only the answers where iris recognition was not ranked as the most accurate method, which would represent the less technically savvy population, we find a higher dispersion and error rate.

Overall, results reject H1 on the basis that the error rate ordering elements a magnitude apart do not reflect accurate perception of biometric methods accuracy.

"Password-based authentication appears in the anwsers as a valid alternative to biometric factors, being ranked in 53 percent of the answers in places comparable to face or voice recognition."

H2: Password-Based Authentication

Password-based authentication appears in the answers as a valid alternative to biometric factors, being ranked in 53 percent of the answers in places comparable to face or voice recognition. The number of respondents that considered passwords as a better alternative to all biometric methods was very small, with 4 percent. The number considering passwords as the least secure alternative, on the other hand, was one of the strongest components with 42 percent.

There was no significant effect of technical savviness on perception of password methods security ($F = 1.04$, $p > 0.42$). Although further research is needed on this front, data gathered is insufficient to reject H2. However, there is a general consensus about the weaknesses of password-based methods contrasted with biometrics. Human-generated passwords can be as secure as one could desire, based on privacy behaviors and choices about length, complexity, and unpredictability.

H3: Less-Known Factors are Perceived as Less Secure

One of the less-used factors presented was iris recognition, with no exposure of participants beyond perceptions build from media exposure to the factor.

Iris recognition was perceived as the most secure method in 53 percent of the answers (only 7 percent of the cases as the worst option).

In the case of voice-based authentication, another biometric factor with little diffusion, the case is exactly the opposite, where 62 percent of the answers placed it as the less secure biometric factor.

Overall, H3 is rejected, showing that perceptions about biometric factors are constructed independently from the level of exposure to technology In fact, when participants were asked directly about the most and least secure method, results were consistently similar.

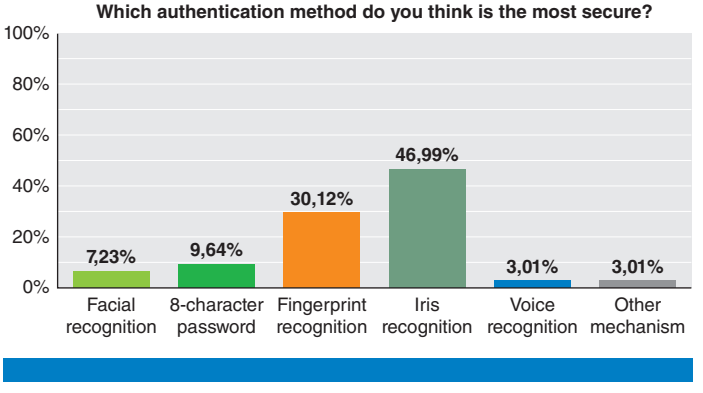


Figure 2: Ranking of perceived security among a given set of authentication methods
(Source: Intel, 2014)

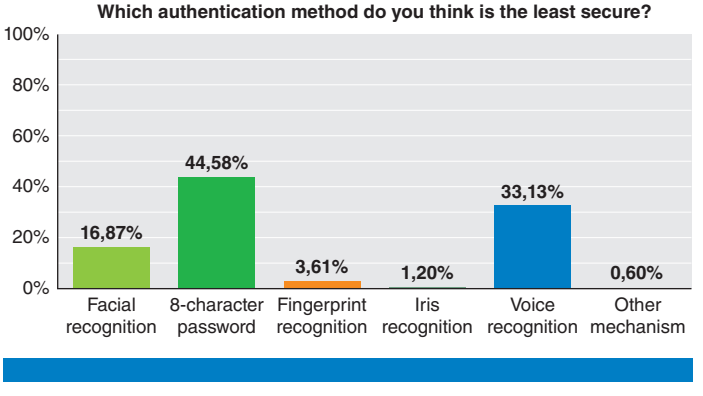


Figure 3: Ranking of perceived insecurity among a given set of authentication methods
(Source: Intel, 2014)

H4: Important Passwords are Easier to Remember

In contrast to the intuitive notion, another interesting finding of our research reveals that the password participants fear the most to lose is also the password they tend to forget more frequently. The password for an online banking account is the password participants are most worried about losing (58.43 percent of the participants agreed on this) and at the

"In contrast to the intuitive notion, another interesting finding of our research reveals that the password participants fear the most to lose is also the password they tend to forget more frequently."

same time, it is the password participants are most likely to forget. Online banking accounts are the less frequently used passwords (compared to email, Social Networks, and OS login) and since sessions eventually expire, they are the most difficult to remember from the user point of view (see Figures 4 and 5).

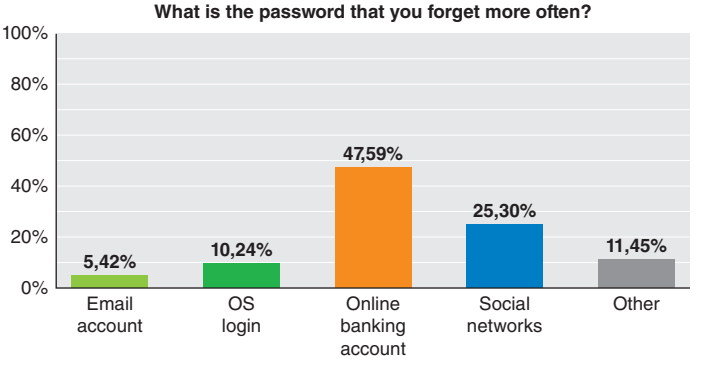


Figure 4: Ranking of passwords users are most likely to forget. (Source: Intel, 2014)

Figure 5: Ranking of passwords users fear the most to lose. (Source: Intel, 2014)

"New research hypothesis leverage perceptions in the design of an overall holistic experience."

"Users tend to give human attributes such as motivations, beliefs and feelings to technology, reacting as if it were a human."

Characterization of New Hypotheses

In this section, we derive new research hypotheses from the results of the study. The objective of these new hypotheses is the generation of knowledge on how perceptions about biometric factors are constructed, and therefore, how to leverage those perceptions in the design of an overall holistic experience. The proposed hypotheses revolve around four main pillars, described here.

Anthropomorphization of Biometrics

Users tend to give human attributes such as motivations, beliefs and feelings to technology, reacting as if it were a human.[9]

It is a fact that humans are better equipped to be recognized by others through their face, voice, or meaningful information they know about each rather than by random pass phrases combining letters, numbers and awkward characters. Nowadays, authentication via biometric verification is becoming increasingly common in corporate and public security systems, consumer electronics, and point-of-sale applications. Increased security requires increased complexity in terms of access; stronger passwords and more authentication points represent a huge usability barrier that biometrics tends to solve. The driving force behind biometric verification has been convenience, making it the most transparent of experiences, if only the implementation is correct.

"The driving force behind biometric verification has been convenience, making it the most transparent of experiences."

A new framing for the rejection of H1 could be set under the principle of anthropomorphism, where factors that appear more natural in human beings (like face recognition) rank higher in perception than they actually are according to reported data.

Red-Light Effect

People modify their behavior when they are being observed. The use of a camera, hence the name red-light effect, can be a great deterrent to rogue behavior in a variety of domains.[10]

Some biometric factors reflect exactly this principle, such as face recognition, while some others can present variations such as voice recognition or fingerprint authentication. Factors that only require a passive authentication from the user can actually increase perceived security compared to those that need the user to actively engage with the device to trigger authentication.

Intrusiveness

Slightly contrasting with the consciousness hypothesis, in many situations intrusive signals can interfere with a higher-priority primary task. In those situations, most people want to turn off intrusive signals that don't indicate emergencies, for instance, by turning off notifications. Some biometric factors are in nature less demanding of user attention, such as face recognition, and therefore more suitable for interactions where a more continuous authentication is required. Other methods, such as fingerprint recognition, alter the mental flow by requiring a physical interaction, therefore augmenting the overall cognitive dissonance of the task.

"In most cases, authentication is an interruption in the user´s primary task flow, and a disruption to working memory."

In most cases, authentication is an interruption in the user's primary task flow, and a disruption to working memory.[11] The greater the demands on working memory from the authentication process, the greater the risk of forgetting aspects of the task at hand. Working memory is the mental process by which information is temporarily stored and manipulated in the performance of complex cognitive tasks. The capacity of working memory is limited, and varies between individuals. Models of working memory describing a multicomponent system including a phonological loop and visuospatial scratchpad were introduced in the early 1970s[5] and have decades of empirical support. The *phonological loop* stores and rehearses verbal and other auditory information, while the *visuospatial scratchpad* manipulates visual images.

"Identity theft is the major concern associated with facial recognition according to approximately 33 percent of the surveyed participants."

Common Concerns

Identity theft is the major concern associated with facial recognition according to 32, 53 percent of the surveyed participants (see Figure 6). Privacy is the second major concern and it has a lot to do with the fear of having a camera probably observing you all the time and not only in the work environment but also at home. The third concern is related to the slow performance. Finally, only a small group of respondents have expressed their concern about the lack of security, inducing the hypothesis of a "good enough" security level for most common situations.

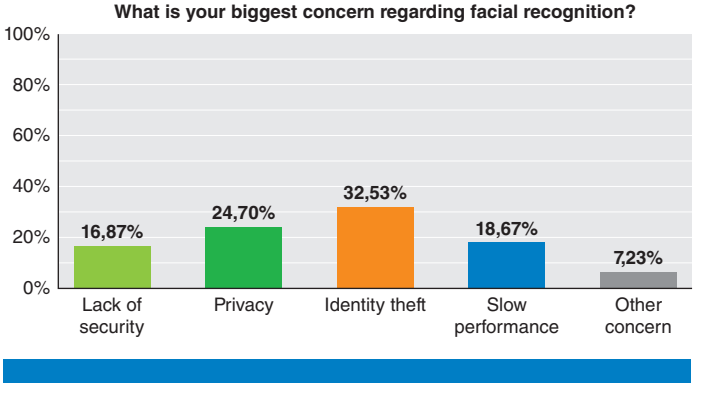


Figure 6: Most common concerns regarding facial recognition (Source: Intel, 2014)

Besides their fears, users didn´t express a lot of discomfort with the idea of using the built-in camera of the device so as to recognize them (Figure 7).

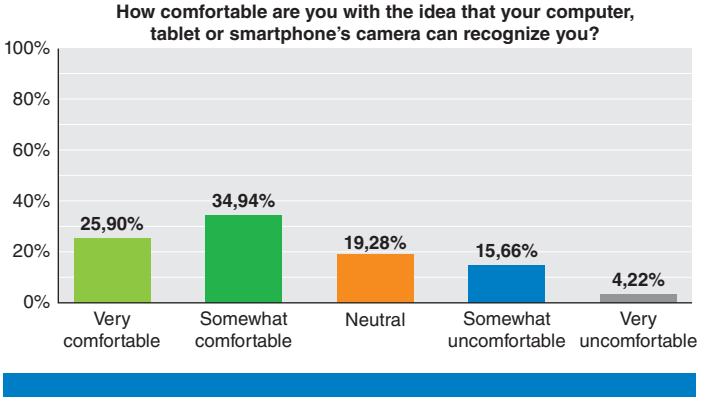


Figure 7: Level of comfort regarding facial authentication in personal devices (Source: Intel, 2014)

"The design of user experiences is a multifaceted problem, so complex in the number of factors that results are better achieved using an iterative approach."

Design Principles

The design of user experiences is a multifaceted problem, so complex in the number of factors that results are better achieved using an iterative approach when developing the product by observing the reaction of actual users, to get

feedback into the construction process again.[12] However, choosing the right set of fundamentals depending on the target population, domain, and type of product can significantly speed up the overall result. These fundamentals, also known as design principles or experience attributes, are a cornerstone of the generation of meaningful experiences.

In this section, we frame the new hypotheses as design principles that, in the context of biometrics, can help test in products how security perceptions are shaped.

Be Secure and Convenient

The value to users of using biometric elements of how you are versus what you know is convenience. It is key that in the tradeoff with security, this convenience is not lost. Biometric systems will always have the benchmark of text passwords, in performance and cognitive load, so the interaction has to be more convenient than traditional methods. The consequences for interaction design affect the determination of performance measures, authentication flows, and incremental authentication.

"Biometric systems will always have the benchmark of text passwords, in performance and cognitive load, so the interaction has to be more convenient than traditional methods."

Don't be Invasive

Clearly contrasting "what you know" methods, biometrics put the user, and user data, on the spot. The idea of a loss of privacy or potential theft of biometric data is extremely disturbing. Similarly, a daunting corporate figure physically scrutinizing the user affects the interaction style and the levels of security.

"The idea of a loss of privacy or potential theft of biometric data is extremely disturbing."

Don't be too Easy

Gathering some biometrics may appear as seamless, too easy, when matched with the value of the data they are providing access to.[13] If the only sign of interaction to access a user's financial records is a camera being turned on, then the ease of the interaction may ironically diminish the perception of security.

Especially under sensitive conditions, users expect to pass through a security door before entering the vault. That's a natural expectation, and it makes people feel safer with the solution. In other words, users want to be aware that the authentication process is happening even though they don't want to do anything about it.

"Especially under sensitive conditions, users expect to pass through a security door before entering the vault."

Conclusions

This article presents a user experience perspective on the interpretation of data about perceptions of biometric authentication factors. A compilation of actual performance of biometric factors is matched with the results of a survey to detect differences in perceptions the level of security, intrusiveness and easy-to-use in the context of authentication.

Although survey results show a preference of biometric methods over traditional password-based authentication, the perceived difference in security is somehow distorted from effective data from research. The particular differences are

examined and interpreted in light of other components of the user experience, such as perceived threats, convenience, and anthropomorphization.

Finally, hypotheses for divergence in perceptions were categorized under focus areas, which become design principles for biometric-based products. The nuances of interaction style and holistic experiences are also illustrated with reference implementations of experience design.

The Future

"This research shows significant differences in security perception of biometric factors when matched against lab results."

This research shows significant differences in security perception of biometric factors when matched against lab results. The differences create new hypotheses about how the user experience of a biometric authentication should perform. Construction of user perceptions is a holistic task and follows an iterative approach for which the use of design principles is well suited. More research is needed in this direction to determine which hypotheses are actually valid, and which design principles will move forward a wide adoption of authentication using biometric factors.

"It is likely that we will see biometrics initially replacing security questions or used in passive/unconscious authentication (such as to verify user´s identity when calling a financial assistant or 911)."

It is likely that we will see biometrics initially replacing security questions or used in passive/unconscious authentications (such as to verify user´s identity when calling a financial assistant or 911). So far, our study shows that only a small portion of the whole universe of users has adopted at least one biometric authentication technology, the so-called "early adopters," which represent less than the 13.5 percent of all users. It will take some time for the different existing products and software solutions to finally replace their traditional 8-character password authentication method. Everything indicates that we will see both technologies coexisting during the next decade or so.

References

[1] Tripathi, K. P., "A Comparative Study of Biometric Technologies with Reference to Human Interface." *International Journal of Computer Applications,* 2011, vol. 14, no 5, p. 10–15.

[2] Miller, George A., "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information," *Psychological Review* 63 (2): 81–97. doi:10.1037/h0043158

[3] Burnett, Mark "10,000 Top Passwords" 2011.

[4] Leyden, John, "Lazy password reuse opens Brits to crooks' penetration," *The Register*, July 20, 2012.

[5] Prabhakar, Salil, Sharath Pankanti, and Anil K. Jain, "Biometric Recognition: Security and Privacy Concerns," *IEEE Security and Privacy,* 2003.

[6] Jain, Anil K., Sharath Pankanti, Salil Prabhakar, Lin Hong, Arun Ross, and James L. Wayman, "Biometrics: A Grand Challenge,"

Proceedings of the 17th International Conference on Pattern Recognition, 2004, ICPR 2004, (Vol. 2) August 2004.

[7] Roselin, Vanaja, E. Chirchi, and L. M. Waghmare, "Iris biometric recognition for person identification in security systems," *International Journal of Computer Applications* (0975 - 8887) Vol. 24 No.9, June 2011.

[8] Jain, Anil K., Arun Ross, and Salil Prabhakar, "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, (Vol. 14, Issue 1) June 2004.

[9] Kynsilehto, Minna and Thomas Olsson, "Exploring the use of humanizing attributions in user experience design for smart environments," at the European Conference on Cognitive Ergonomics, Helsinki, Finland, 2009.

[10] Ernest-Jones, Max, Daniel Nettle, and Melissa Bateson, "Effects of eye images on everyday cooperative behavior: a field experiment," *Evolution and Human Behavior*, 2011, vol. 32, no 3, p. 172–178.

[11] Trewin, Shari, Cal Swart, Larry Koved, Jacquelyn Martino, Kapil Singh, and Shay Ben-David. "Biometric authentication on a mobile device: a study of user effort, error and task disruption," *Proceedings of the 28th Annual Computer Security Applications Conference*, ACM, 2012, p. 159–168.

[12] Tidwell, Jenifer, *Designing Interfaces* (O'Reilly Media Inc., 2010).

[13] Väänänen-Vainio-Mattila, Kaisa, Jarmo Palviainen, Santtu Pakarinen, Else Lagerstam, Eeva Kangas, Alessandro Deserti, Francesco Zurlo, and Francesca Rizzo, "User perceptions of Wow experiences and design implications for Cloud services," DPPI:ACM, 2011. - ISBN 978-1-4503-1280-6, p. 63.

Author Biographies

(José) Pablo Piccolotto is a senior software engineer currently acting as a user interaction designer in a security project jointly developed by McAfee and Intel. He joined Intel in 2008 and since then he has been working at the Argentina Software Design Center (ASDC). He has authored four patents in the name of Intel and 17 invention disclosure forms so far.

He graduated in 2007 with honors from the Aeronautic University Institute in Argentina. He also holds an MBA and a postgraduate degree in Engineering Management from the National Technological University. He is currently pursuing a doctorate in the National University of Córdoba.

In addition to his career at Intel, Pablo is a postgraduate university professor at the Blas Pascal University, and a part-time researcher and professor at

the Aeronautic University Institute. He participated as a lecturer in several conferences, seminars, workshops, and training events on topics such as innovation, entrepreneurship, user experience, usability, intellectual property and software validation. He can be contacted at jose.p.piccolotto@intel.com or pablopiccolotto@gmail.com.

Patricio Maller holds a computer science degree and a Master of Science in computer sciences (2000) with focus on human-computer interaction. He was a Fulbright Scholar at The University of Alabama between 1998 and 2000, completing research on the application of socio-cognitive theories to the acceptance of IT technology.

Patricio is a senior interaction designer and UX leader in a security project jointly developed by McAfee and Intel. He joined Intel in 2006 and since then he has been working at the Argentina Software Design Center (ASDC). Patricio also worked at Motorola, and the educational initiative educ.ar, leading a complete redesign.

Patricio has authored many articles related to processes Agile and UX, and is currently a researcher at the Aeronautic University Institute. He can be reached at patricio.maller@intel.com or pmaller@gmail.com.

A SURVEY OF BIOMETRICS FOR WEARABLE DEVICES

Contributors

Cory Cornelius
Intel Labs

Christopher N. Gutierrez
Purdue University

Biometrics are characteristics about ourselves that we use to recognize each other. Whether it is the structure of our face or the way we walk, we can measure these physiological or behavioral signals and use statistical methods to distinguish amongst users. Rather than "something you know"—as in traditional password-based authentication systems—these biometric-based methods enable systems to authenticate you based on "something you are." In this survey we describe the current state-of-the-art of biometric techniques and evaluate their use in authentication systems with a particular focus on client devices and emerging wearable devices. We examine the academic literature and commercial offerings of biometrics and summarize each biometrics' utility over a variety of criteria and provide guidance for their use in client and wearable devices.

Introduction

For many systems it is often useful for the system to know who is interacting with it. They can prevent unauthorized users from accessing sensitive data (for example, in which an adversary Alice tricks Bob's sensor into divulging his activity data to her smartphone), correctly describe who is using the system, or personalize the experience for that user, attaching such an identity a method of recognizing who is interacting with the system. To do this, we can use biometrics, which have advantages over passwords ("what you know") and security tokens ("what you have"). Biometrics cannot be lost, forgotten, easily stolen, or shared, which are common issues with password or security token systems. They are also resilient to "shoulder surfing" where an adversary steals credential information by observing how the user authenticates to a system.

"Biometrics cannot be lost, forgotten, easily stolen, or shared, which are common issues with password or security token systems."

Categorization

Physiological biometrics use some characteristic of your physiology to identify you. These tend to be the biometrics people are most familiar with: fingerprint, hand geometry, facial recognition, iris or retinal recognition. Physiological characteristics range from noninvasive characteristics like facial features and hand geometry to more invasive characteristics like the impression of a finger, the makeup of DNA, or the structure of the iris. These types of biometrics typically require a sensor attached to the subject or require the subject to place a limb on a sensor.

On the other hand, *behavioral* biometrics use some characteristic of your behavior to identify you. Behavioral characteristics include things like the dynamics of using a keyboard, the acoustic patterns of the voice, the

mechanics of locomotion, and how one signs a signature. In contrast to a physiological biometric, behavioral biometrics can exhibit wide within-subject variation since they are sensitive to things like mood. Likewise, they also tend to be easier to collect since they generally do not require the subject to be interrupted.

These categories are not mutually exclusive. For example, a voice-recognition biometric has both physiological and behavioral aspects. Your voice is shaped by your vocal tract (the larynx, pharynx, oral and nasal cavities), however your current behavior can also affect your voice. For example, your current state of mind (for example, being excited or nervous) can alter your vocal tract and therefore your voice.

Recognition

The usefulness of biometrics relates to their ability to be used to recognize a person for some population. The size of the population is important since some sensors, while seemingly unfit for distinguishing large populations, maybe be able to distinguish smaller populations. Given a population, biometrics can be used in one of two ways: *identification* and *verification*. Before identification or verification can occur, the system first needs to collects many biometric samples from a population. The combination of the biometric samples and the subject's identity forms a *biometric template* for each subject.

"The size of the population is important since some sensors, while seemingly unfit for distinguishing large populations, maybe be able to distinguish smaller populations."

Identification

Identification is a one-to-many matching. The system uses a pre-chosen set of biometric templates to determine the identity of any subject in that population. That is, some unknown subject would present themselves to the system, and it is the system's job to determine which subject that is from the population. A biometric system accomplishes by first sampling the unknown subject's biometric. It then examines its database of biometric templates and finds the biometric template that best matches that subject's biometric sample. The system then asserts that the identity of the unknown subject is the identity of the biometric template that best matches that unknown subject's biometric sample.

Verification

Verification, on the other hand, is a one-to-one matching. That is, a subject presents an identity (such as a name), and it is the job of the biometric system to verify that identity. When a subject presents themselves to the system and asserts their identity, the system retrieves the biometric template associated with that identity from the database. It then samples the subject's biometric and matches that subject's biometric sample with the biometric template. If the biometric sample matches the biometric template (for some matching metric), then the asserted identity is verified. For example, one might use a generative model to learn the distribution of a subject's biometric and select a threshold to accept a new biometric sample according to likelihood. Notice that there is no assumed population; rather, it should reject everyone else in the world.

"Unlike a password or security token system, biometric authentication is dependent upon pattern recognition to determine whether the biometric data presented belongs to the desired user."

"Ideally the rate of these errors would be minimized in order to have a reliable biometric system."

Evaluation Criteria

Unlike a password or security token system, biometric authentication is dependent upon pattern recognition to determine whether the biometric data presented belongs to the desired user. Because pattern recognition methods are probabilistic, biometrics often output a matching value or probability (for example, the presented biometric data belongs to the user with 97-percent probability) rather than a match or no match value as in password or security token systems.

Given such a value, a system can choose a threshold at which they accept (or conversely, reject) a biometric sample. Choosing too low a threshold would cause the system to routinely authenticate invalid users, while choosing too high a threshold would cause the system to incorrectly reject a valid user. Ideally the rate of these errors would be minimized in order to have a reliable biometric system. In practice, however, there is a tradeoff between these two errors that a system designer can choose. It might be more desirable, for example, to make sure all invalid users are correctly rejected at the expense of rejecting some valid users (sacrificing usability in the name of security), or vice versa. Because biometrics leave the choice of this threshold to system designers, evaluations of biometrics often show the rates of these errors for a variety of chosen thresholds.

Performance Metrics

The *false accept rate* is the rate at which the system incorrectly accepts an impostor as another individual. The false accept rate directly impacts the security of the system. The *false reject rate* is the rate at which the system incorrectly rejects a truthful claim of identity. The false reject rate directly impacts the user experience of the system, requiring a valid individual to make multiple attempts in order to successfully authenticate. The *equal error rate* (EER) is the rate at which the false accept rate equals the false reject rate. As such, this metric combines the false accept rate and false reject rate into a single metric. However, it is important to remember that the false accept rate and false reject rate can be tuned by choosing different thresholds. When comparing different biometrics, comparing their respective equal error rates is useful but does not capture different choices of thresholds.

Population Size

For all of these metrics, it is important to keep in mind the population they were sampled from. The size of the population, for example, will determine how unique the biometric actually is. It is believed, for example, that fingerprints are universally unique. To validate such a claim, one would need to collect the fingerprint of every human being, which is impossible. Instead, biometric researchers sample a population in order to arrive at some metrics about the uniqueness of the biometric. The size and makeup of the population they sampled is important to keep in mind when evaluating a biometric. In some systems, like those used for evidence in a court case, it is desirable to ensure that the biometric being used is universally unique. In other systems such a stringent requirement might not be necessary. In many wearable devices, for example, the number of users who will use the device over the course of the device's lifetime is small or may be known *a priori*.

Comparative Methodology

To evaluate these biometrics, we adopt the qualitative evaluation framework proposed by Bonneau et al.[7] Their evaluation highlights various dimensions of user authentication systems unified under a framework of 25 different benefits spanning usability, deployability, and security.

We briefly describe each benefit in Table 1 and highlight any assumptions we make in order to properly evaluate wearable biometric schemes. Each metric is rated as "offers the benefit," "does not offer the benefit," or "almost" offers the benefit, as indicated with the "quasi-" prefix. We also assume that each biometric scheme is implemented using best practices since a poor implementation could kill any scheme.

Usability Benefits	
Memorywise-Effortless	The user does not need to memorize any secrets
Scalable-for-Users	Whether the scheme requires additional user burden for each additional account (for example, passwords are not Scalable-for-Users since it requires a unique password per user account)
Nothing-to-Carry	The user does not need to carry a physical device in order to authenticate. We make the assumption that the user always carries the wearable device.
Physically-Effortless	The authentication does not require physical effort beyond touching the device. Quasi-Physically-Effortless are schemes that require additional effort but is natural for a user (speaking, walking, or interacting with a touch surface).
Easy-to-Learn	The user can intuitively learn how to use the scheme with little direction.
Efficient-to-Use	The time to gather the data to authenticate the user is acceptably short and consistent. We define "acceptably short" as roughly the same amount of time as it takes to type an eight-character password.
Infrequent-Errors	The scheme authenticates an honest user that provides the proper credentials without a probability of rejection.
Easy-Recovery-from-Loss	A user can easily recover from compromised or forgotten credentials without significant latency or issue.
Deployability Benefits	
Accessibility	Physical (not cognitive) disabilities or illness does not prevent the user from authenticating.
Negligible-Cost-per-User	The cost of implementing the scheme is negligible to the user and online service provider (does not require special hardware or infrastructure to utilize the scheme).
Server-Compatible	The verifier does not need to change existing authentication systems.
Browser-Compatible	Users who use a HTML5-compliant browser with Javascript enabled do not need to install additional plugins in order to authenticate. Quasi-Browser-Compatible means that the authentication data can be sent over the browser with a common API but not necessarily used for authentication (a slight variation to Bonneau et al. definition).
Mature	The scheme is available commercially with extensive testing and common open standards. Quasi-Mature schemes have limited market presence and is in the transition from research to practice with compelling results and user studies.
Nonproprietary	Open-source implementations and techniques, methods, and standards are publicly available.

(Continued)

No-Additional-Sensors-Required	Various biometric systems utilized commonly available sensors. A biometric that has No-Additional-Sensors-Required utilizes sensors that are commonly available on a modern smartphone. This list includes accelerometer, gyroscope, microphone, capacitive touch surface, and RGB camera. If a wearable biometric requires a one of these sensors, it could request the smartphone to capture the information on its behalf.
Security Benefits	
Resilient-to-Physical-Observation	Attackers cannot impersonate a valid user through observation.
Resilient-to-Targeted-Impersonation	An attacker cannot use personal information impersonate a valid user
Resilient-to-Throttled-Guessing	An attacker guessing a user's credential is bounded by the verifier using techniques such as throttling repeated guesses.
Resilient-to-Unthrottled-Guessing	Constrained only by computational resources, an attacker cannot guess data used for authentication.
Resilient-to-Internal-Observation	An attacker cannot intercept the user's input from within the device. That is, the user's authentication credentials are resilient to an adversary that has malware installed on the user's device
Resilient-to-Leaks-from-Other-Verifiers	Information leakage from verifiers does not help an attacker impersonate a user.
Resilient-to-Phishing	An attacker cannot impersonate a valid verifier to collect authentication credentials to impersonate the user.
Resilient-to-Theft	The scheme is not vulnerable to theft of a physical object. For biometrics, we expand the definition to include biometrics that are not easily sampled without the attacker being in physical contact with the user.
No-Trusted-Third-Party	The authentication system does not rely on a trusted third party
Requiring-Explicit-Consent	The authentication process cannot be started without explicit user consent.
Unlinkable	Verifiers cannot collude to determine whether the same user is present in multiple online services.

Table 1: Twenty five dimensions of user authentication systems unified spanning usability, deployability, and security using Bonneau's framework.[7].(Source: Cory Cornelius and Chris N. Gutierrez, 2014)

"There are many biometrics one could survey, so we limit ourselves to those biometrics that are suitable for integration into a wearable or client device."

Evaluation of Biometrics

There are many biometrics one could survey, so we limit ourselves to those biometrics that are suitable for integration into a wearable or client device. We examine gait-, voice-, face-, electrocardiogram-, electroencephalogram-, bioimpedance-, ear-, and touch-based biometrics. Table 2 shows the evaluation of these biometrics under the comparative methodology described in the previous section.

All of the biometrics we surveyed offer the usability benefit of *Memory-wise Efficient* (biometrics do not require memorization of secret information), *Scalability-for-Users* (users use the same biometric information across all web accounts), *Nothing-to-Carry* (assuming that the user always carries the wearable), and *Easy-to-Learn* (the biometrics we surveyed are intuitive to use). All the biometrics surveyed lack the following deployability benefits: *Infrequent-to-Errors* (none of the biometrics offer 100-percent accuracy), *Easy-Recovery-from-Loss* (it is insecure to reuse the same biometric when it is compromised),

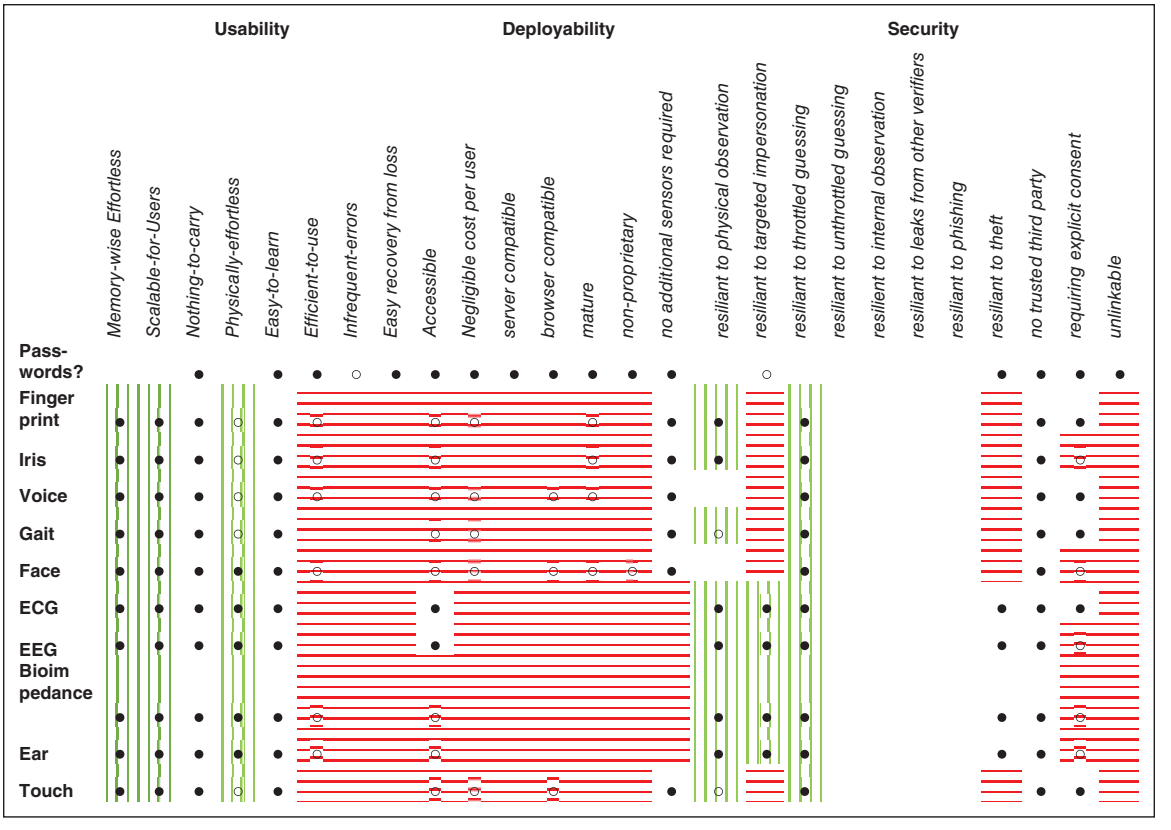


Table 2: Comparative Evaluation of the Selected Biometric for the 25 Benefits Described in Table 1. (Source: Cory Cornelius and Chris N. Gutierrez, 2014)

Server-Capable (server side software does not typically support biometric authentication). These biometrics also lack the following security benefits: *Resilient-to-Unthrottled-Guessing* since an attacker can feasibly sweep across the entire feature space, *Resilient-to-Internal-Observation* because an attacker can steal raw biometric data when the device is compromised, *Resilient-to-Leaks-from-Other-Verifiers* since the biometric data is the same across all verifiers, colluding verifiers can identify whether the same user is present, and *Resilient-to-Phishing* because an attacker can still trick a user into providing a biometric to a malicious verifier. One benefit that is common across all these biometrics is *No-Trusted-Third Party* since the user can interact with the verifier directly. In the remaining sections, we briefly describe each biometric and provide the reasons for our evaluation choices for each biometric.

Gait Recognition

Gait recognition is a behavioral biometric that attempts to learn the distinctive characteristics of how a person walks. Humans have been known to be able to distinguish people based on their gait. It is possible, for example, for humans to distinguish a particular person even when all other distinguishing features are removed. Thus gait recognition systems seek to capture this same process.

"Gait recognition is a behavioral biometric that attempts to learn the distinctive characteristics of how a person walks."

Recent techniques for recognizing humans use accelerometers and gyroscopes and make this biometric more easily integrated into a wearable device.

Accelerometers capture the acceleration of the limb they are attached to, while gyroscopes capture its rotational velocity. These sensors are able to sensor in three orthogonal dimensions. When combined, they yield a 6-dimensional vector that is capable of capturing everything about the motion of a particular human.

Typically, these sensors are placed at the subject's waist, wrists, or ankles in order to capture motion related to the subject's gait. In order to properly capture the subject's gait, the sensors needs to be sampled regularly at a sufficient rate. From this signal, minima and maxima are extracted in order to segment a gait into cycles corresponding to the forward movement of each leg. From these cycles, various time-based and frequency-based features are extracted to form a gait sample. The primary goal of this feature extraction phase is to extract some distinctive features about a subject's gait. A secondary goal is to extract features that are invariant to a variety of factors. For example, the orientation of the sensors will affect how the subject is sensed. Likewise, subjects might walk faster or slower depending on many factors and ideally features would be invariant to speed or pace.

"Because gait recognition using sensors is a relatively young biometric, the EER for many in-the-lab-based studies is in the 20–30 percent range."

Because gait recognition using sensors is a relatively young biometric, the EER for many in-the-lab-based studies is in the 20–30 percent range.[6] Likewise, the size of the population in many of these studies is relatively small (< 100). The major issue to overcome is that gait biometrics is sensitive to many external and internal factors. For example, the kind of shoe a subject is wearing affects recognition.

"In comparison to other biometrics suitable for wearable devices, gait recognition requires more effort."

Usability

In comparison to other biometrics suitable for wearable devices, gait recognition requires more effort. A user is required to walk for a few seconds in order to be accurately identified. This is clearly more physical effort than typing in a password or touching a sensor. However, we consider walking to be a natural action and thus label gait recognition as *Quasi-Physically-Effortless*. Capturing a user's gait requires them to walk for a few seconds, which is considerably slower than passwords and other biometrics. We therefore consider gait recognition as not *Efficient-to-Use.*

Deployability

Gait recognition is *Quasi-Accessible* as certain disabilities or physical conditions will limit access to certain users. Gait recognition is not *Browser-Capable* as robust accelerometer and gyroscope data is not typically transmitted through the browser. We claim that gait recognition is not *mature* as the usage of accelerometers and gyroscopes to recognize a user's gait has not been extensively studied.

Security

We consider gait recognition as *Quasi-Resilient-to-Physical-Observation*. We assume that a malicious observer would require significant effort to properly reproduce the accelerometer and gyroscope data. To the authors' knowledge, spoofing a user's gait (from data gathered through physical observation) either

through human impersonation or building a machine to mimic a user's gait is an open research question. Gait recognition is not *Resilient-to-Targeted-Impersonation*. Information about a user such as height, age, sex, weight and so on could feasibly be gathered online. It may be also possible to obtain a video of a user walking via social networks. This information could be used to make an educated guess of how a user walks. A user's gait is not *Resilient-to-Theft* since it may be possible for a user to steal the device and attempt to mimic a user's gait. Gait requires a user's explicit consent since it requires a user to walk to capture the biometric. Gait recognition does not require additional sensors since the accelerometers and gyroscopes are commonly found on smartphone.

Voice Recognition

Voice recognition is a physiological biometric that distinguishes people based on the way they talk. It is easy, for example, to distinguish biological sex solely based upon pitch since biological females tend to have higher pitch versus biological males. The theory of voice recognition is that the vocal tract of each human shapes ones voice in a manner that is uniquely distinguishable. Humans, for example, are very good at recognizing each other using audio recordings only.

"Voice recognition is a physiological biometric that distinguishes people based on the way they talk."

In order to capture a subject's voice, voice recognition systems typically use a microphone of some sort to capture sound. These microphones usually sample the subset of frequencies that humans are capable of producing (up to 4 kHz). Experimental setups typically involve a microphone placed directly in front of the subject's mouth. However, so long as the microphone can hear the subject's voice unimpeded by any filters then it should be possible to recognize the subject. So, for example, it should be possible to integrate a microphone into a wearable device like a bracelet. The major concern with the integration of a microphone is the energy requirements necessary to capture the large bandwidth a voice occupies.

Given an audio signal, most voice recognition systems extract features from the frequency domain of this signal. Before the subject's voice can be recognized, an important first step is to segment the audio signal into voiced and non-voiced sections. This allows the system to disregard those sections of the audio signal that do not include human voice. The most popular features attempt to model the human hearing system. These features map and weight frequencies in a way that corresponds to frequency response of the human hearing system. From this frequency mapping and weighting other features can be extracted to form features that are invariant to certain factors. For example, all microphones exhibit different frequency response characteristics and it is important to account for this. Likewise, one must account for noise or ambient sound in the signal. More recent research has attempted to model higher-level characteristics of each subject's voice. For example, how humans say certain words including the phonemes that make up voiced words or the duration of certain phonemes.

There are two primary types of voice recognition systems: text-dependent and text-independent. Text-dependent systems rely upon each subject voicing a

small corpus of words, while text-independent systems do not rely upon any corpus of voiced words and are more general. As one would expect, text-dependent systems tend to perform better than text-independent systems. Some systems also incorporate a so-called "universal background model" that attempts to model human speech in general. This allows the system to learn the difference between a particular subject's voice and a general subject's voice, which increases performance.

"Text-dependent systems tend to perform in the sub-5 percent EER range and in ideal conditions under 1 percent. Text-independent systems perform between 10–25 percent EER for moderate-sized populations (>100)."

Text-dependent systems tend to perform in the sub-5 percent EER range and in ideal conditions under 1 percent. Text-independent systems perform between 10–25 percent EER for moderate-sized populations (>100). In the text-independent case, it has been shown voice recognition systems perform on par with humans. However, these systems need to be very sensitive to the type of microphone used and whether multiple types of microphones are used, while humans are not.

Usability

Voice recognition is *Quasi-Physically-Effortless* since most users have the ability to talk. The user's voice is *Quasi-Efficient-to-Use.* The amount of time needed to read a specific word or phrase may or may not be longer than typing in a password (or presenting other biometrics).

Deployability

Further, voice is *Quasi-Accessible* as the voice may be affected by illness or injury. We consider voice biometric as *Quasi-Negligible-Cost-Per-User* as most mobile devices are equipped with microphones (and can be easily integrated into wearable devices). HTML5 allows voice to be captured via web browser thus we consider voice recognition as *Quasi-Browser-Capable.* Although not as extensively researched as passwords, voice recognition has had a quite a few years of development and thus we consider it *Quasi-Mature*. Since microphones are typically found on smartphones and microphones can be placed on wearable devices, we state that voice recognition has *No-Additional-Required-Sensors*.

Security

Since voice can easily be recorded and replayed, we claim that voice recognition is not *Resilient-to-Physical-Observation.* It may be feasible for an adversary to gather a voice sample online and attempt to impersonate a user and thus not *Resilient-to-Targeted-Impersonation*. Since a voice can easily be recorded and replayed, we state that voice is not *Resilient-to-Theft.* The user must be talking in order for the system to recognize the user's voice so we state that voice biometric *Requires-Explicit-Consent.*

Face Recognition

Face recognition is a physiological biometric that humans often use to recognize people. Because humans are able to recognize faces effortlessly, faces are a natural candidate for a biometric. There are a variety of unique facial features as a result of genetic and environmental factors. But provided with even a relatively distorted representation of a person's face (such as a cartoonish drawing), humans are capable of recognizing an individual.

In most face recognition systems, a camera is used to capture an image of the face. The quality of the camera will affect the performance of the system. Likewise, the position of the camera will affect whether the subject is in the field of view. Like a microphone, cameras require large bandwidth and in wearable devices become a concern due to energy constraints. With the recent commodification of depth sensors, newer research has examined how RGBd cameras can improve the performance of face recognition systems. Further, researchers have examined how radiation other than visible light could be integrated into face recognition systems.

Given an image, typically the first job of a face recognition system is to localize the subject's face for segmentation. Once segmented, the system localizes prominent features on the subject's face, like the eyes, mouth, and nose. The system can then compute the geometric relationship of these prominent features to form features that are invariant to environmental factors like illumination and orientation. Other features can be computed that are invariant to facial expression. Some face recognition systems utilize the entire segmented image as a feature; however these systems often require many images of the subject to account for a variety of factors. It is also possible to combine both feature sets in a hybrid fashion.

Face recognition systems in controlled settings tend to perform very well (<1 percent EER). However, in more realistic conditions, the performance drops to somewhere between 5 and 10 percent EER for large populations (>1000). Face recognition systems, however, perform better than humans for recognizing subjects from a large population. System performance degrades even more when factors like distance to camera and occlusions are taken into account.

Usability

If a camera is properly oriented, face recognition is *Physically-Effortless*. Face recognition can identify a user's face within seconds (given ideal conditions), which is comparable to the amount of time a user takes to type in a password. We thus claim that face recognition is *Quasi-Efficient-to-Use*.

Deployability

We consider face recognition as *Quasi-Accessible* as facial hair, glasses, or physical injury can cause issues. Since digital cameras are widely available, we state that facial recognition is *Quasi-Negligible-Cost-Per-User*. As with voice, HTML5 allows video streaming through the browser so we claim that face recognition is *Quasi-Browser-Compatible*. Face recognition has been researched for several years and a number of commercial products have hit the market. We thus consider facial recognition as *Quasi-Mature*. Unlike other biometrics that we surveyed, we consider facial recognition as *Quasi-Non-Proprietary* as OpenCV contains facial recognition biometric functionality. An adversary can easily capture a user's face with a digital camera when they are physically present. Since digital cameras are widely deployed, face recognition has *No-additional-Required-Sensors.*

"An adversary can easily capture a user's face with a digital camera when they are physically present."

Security

We consider face recognition as not *Resistant-to-Physical-Observation* as an adversary can easily capture a user's face with a digital camera when they are physically present. Further, face recognition is not *Resistant-to-Targeted-Impersonation* since a user's face can typically be found on various social networking websites. Similar to voice recognition, an adversary can capture an image of a user's face and thus is not *Resilient-to-Theft*. Since a user's face is always present, we claim that face recognition does not require *Explicit-Consent*.

Electrocardiogram Recognition

Electrocardiogram (ECG) recognition is a physiological biometric that leverages the unique characteristics of the human heart. Electrocardiography is well known in the medical industry for the usage as a diagnostic tool for heart-related conditions. It is used to sample the electrical impulses the heart uses to pump blood throughout the cardiovascular system. As a biometric, the assumption is that these electrical impulses can be uniquely identifiable across humans and is affected by factor such as the size and position of the heart and the anatomy of the chest.

"…electrical impulses can be uniquely identifiable across humans and is affected by factor such as the size and position of the heart and the anatomy of the chest."

ECG is calculated by measuring the electrical signal of the heart with respect to time. To record the electrical activity of the heart, electrodes are placed on the surface of the skin. Electrode sensors can be placed on variety of locations on the human body. Research indicates that ECG signals with respect to biometric usages are invariant to sensor location. Sensor location includes neck, chest, finger, back, toe, and wrists.

In typical ECG recognition systems, the samples of the heart's electrical signal are collected at 50–500 Hz. The amplitude of this signal is typically affected by the material electrodes (for example, gel versus dry), moisture on the skin, and hair at the measurement location. In medical settings, a 10-electrode, 12-lead system is used to record the heart, where the electrodes are placed on a patient's chest, arms, and legs and are pre-gelled to acquire a strong coupling and thus better electrical signal. However, the number of electrodes necessary to measure ECG in a biometric application can be much less and without gel or medical-grade sensors.

Medical classification of electrical activity from the heart is well understood. There are five parts of the electrical signal labeled P, Q, R, S, and T. The P wave is relatively small in comparison to the other waves and consists of positive polarity, low frequency (10–15 Hz) and lasts about 120 milliseconds. The largest wave, called the QRS complex, lasts about 70–110 milliseconds in a normal heartbeat and exhibits frequencies between 10 and 40 Hz. The T wave is the final wave and has a midrange amplitude in comparison to the other sub waves. The markers of these parts are well defined and can be easily segmented.

Given a signal with labeled P wave, QRS complex, and T wave, distinguishing features can be extracted from this signal. Various attributes such as width,

amplitude, slope, curvature, and direction of waves can be calculated given these points. Additionally, ECG recognition systems can use rhythm analysis to examine the intervals of time between features to compute features such as heart rate variability, instantaneous heart rate, and others. It is also possible to extract spectral features.

Electrocardiogram recognition systems are sensitive to the placement of the electrodes on the body. At the palms, ECG performs at 5–10 percent EER for medium-sized populations (<100). Unlike many other biometrics, ECG is relatively difficult to spoof or capture from a subject. It also has the advantage that acquisition is mostly unobtrusive and can be continuous. It can also be used to detect liveness. However, ECG is sensitive to the activity being performed (that is, heart rate increases with physical activity) and some people have heart-related anomalies that could hinder performance. There are also privacy concerns about the nature of ECG-related data since other factors like arousal or emotional state can be inferred from the data.

Usability

ECG is *Physically-Effortless*—sampling typically requires touch two electrodes for a certain duration. However, ECG recognition is not *Efficient-to-Use* since it takes several seconds to sample.

Deployability

ECG recognition is widely *Accessible* due to the fact that every user must have a heartbeat. Since ECG recognition requires, at minimum, a couple of electrodes, we consider it not *Negligible-Cost-per-User.* It is not *Browser-Capable* since it is not standard to send ECG information over the web browser. Further, ECG is not mature since it is a fairly new biometric that has not been studied extensively and has not been widely available commercially. Sensors to capture an ECG signal are not typically available so A*dditional-Sensors-Required.*

Security

We consider ECG to be *Resilient-from-Physical-Observation* since it is nontrivial to reconstruct an ECG signal for a given user without being in physical contact. Likewise, ECG is *Resilient-to-Targeted-Impersonation* since ECG data for a particular user is typically inaccessible from the general public. Further, ECG recognition requires special hardware in order to authenticate a user so we consider it *Resilient-to-Theft.* Finally, ECG recognition *Requires-Explicit-Consent* since acquiring a signal typically consists of a pair of electrodes that a user must touch or hold for a set of time.

"…it is nontrivial to reconstruct an ECG signal for a given user without being in physical contact."

Electroencephalogram Recognition

Electroencephalography (EEG) is a physiological biometric that measures the electrical activity of the brain. The theory is that the configuration of neurons and synapses are configured uniquely for each individual and as such the electrical properties of this configuration will vary from individual to individual. EEG systems have long been used in the medical

industry to diagnose epilepsy, sleep disorders, and other brain-related pathologies.

The process for collecting EEG data is similar to ECG data collection: rather than collecting electrical activity from the heart, the electrodes are placed near the brain. We imagine these electrodes could be integrated into a hat or headband. In medical contexts, 10 to 20 different electrode locations are used on the scalp; however for biometric purposes usually two electrodes are used. These electrodes are placed near the front of the head where there is no hair. The signal is sampled at a relatively fast rate of 256 Hz.

Unlike ECG, there is no distinctive shape present in an EEG signal. However, EEG signals are classified according to the frequency present in the signal. The frequencies present depend upon the subject's current state. For example, being excited will exhibit a different signal than when the subject is calm. As such, many EEG recognition systems require the user to be in a calm state or require training samples for many different states. From this signal many different spectral features are computed.

The performance of two-electrode ECG recognition systems is 5–10 percent EER for medium-sized populations (<100). Like ECG, EEG is difficult to spoof or capture, and when integrated into a wearable device can be captured continuously and unobtrusively. The EEG signal can also be used for liveness detection. In real world situations, EEG recognition systems might require more training samples than other biometrics to account for the different kinds of electrical activity of the brain. Inducing a mental state in the subject degrades the user experience.

Usability

EEG recognition is *Physically-Effortless* since the device would be worn on the head and the signal could be sampled passively. However, EEG recognition is not *Efficient-to-Use* since the time to sample the single takes significantly longer than typing a password.

Deployability

Since EEG signals are always present (disregarding cognitive injury or disability), we claim that the EEG recognition is *Accessible*. Sensors to sample EEG signals are not widely available thus EEG recognition is not *Negligible-Cost-per-User* and *Additional-Sensors-Required.* Browsers by default are not capable in capturing EEG signals therefore it is not *Browser-Capable.* EEG biometrics is relatively new and not commercially available so we state that EEG recognition is not mature.

"…EEG recognition requires electrodes on the surface of a user's head, it is difficult for an adversary to gather any EEG data without physical contact."

Security

Since EEG recognition requires electrodes on the surface of a user's head, it is difficult for an adversary to gather any EEG data without physical contact. Therefore we claim that EEG recognition is *Resilient-to-Physical-Observation*, *Resilient-to-Targeted-Impersonation,* and *Resilient-to-Theft*. However, since EEG

recognition is intended to be worn on a user's head and sampled passively, it does not *Require-User-Consent.*

Bioimpedance Recognition

"Bioimpedance is a relatively new physiological biometric that recognizes subjects based on their response to a small electrical current."

Bioimpedance is a relatively new physiological biometric that recognizes subjects based on their response to a small electrical current. Depending upon the frequency of the current, different parts of the human anatomy will resist and/or store this current thereby altering it. These alterations can be measured. Thus, bioimpedance hypothesizes that the anatomy of humans are diverse enough to be unique. It is easy to imagine, for example, that bioimpedance across the whole body is unique; however the bioimpedance can easily change due to hydration or weight loss/gain. Thus discovering the correct part of the body to measure bioimpedance that is suitable for a biometric is an open research question.

To measure bioimpedance, a small alternating current must be injected into the body. Typically, bioimpedance recognition systems inject a variety of alternating currents at different frequencies because the anatomy responds differently for different frequencies. Typically one uses electrodes in contact with the skin to inject this current. Once injected, the system then needs to sense how the anatomy responds. There are two primary ways of sensing this: bi-polar and tetra-polar sensing. In bi-polar sensing, the same electrodes that were used to inject the current are also the ones that are used to measure the change in voltage. Tetra-polar sensing, on the other hand, uses a separate set of electrodes to sense the change in voltage. Tetra-polar sensing is immune to the contact impedance present in a bi-polar system, however at the cost of more hardware and electrodes. It is, however, not enough to have just two or four electrodes in a bioimpedance recognition system. Typically one employs multiple electrodes in order to fully sample the location of interest. (One can think of this as a kind of tomography.) For example, at the wrist, one would use eight electrodes to adequately sample the anatomy of the wrist. Thus, multiple impedance values can be sensed from different pairs of electrodes.

Impedance is composed of resistance and reactance and is typically represented as a complex value. Because of this, impedance needs to be decomposed into real values for recognition systems. The naïve method is to create a feature that is resistance and reactance concatenated together. Depending upon the noise in the sampling process, this may be adequate. In practical systems, it is necessary to account for noise and thus more robust features are usually extracted by fitting some model to both the resistance and reactance parts. The benefit of bioimpedance is that the features are relatively simple to compute.

For bi-polar sensing at the wrist, bioimpedance recognition systems perform at the 10–20 percent EER. Tetra-polar sensing should lower this value further. For bi-polar sensing across the body (wrist-to-wrist), bioimpedance performs at the 1–5 percent EER. However, both of these results are relatively

small populations (<10). Performance also degrades with perspiration and environmental factors like humidity and temperature. However, one might be able to account for these factors with more sophisticated models. Like face recognition systems, a subject's bioimpedance might change over time and thus require periodic re-enrollment. It is currently unknown what this period is for different areas of the body.

"Since the sensors can be integrated into a bracelet, we claim that bioimpedance recognition is Physically-Effortless…"

Usability

Although the optimal location is an open research question, recent systems measures bioimpedance at the wrist. Since the sensors can be integrated into a bracelet, we claim that bioimpedance recognition is *Physically-Effortless* since the system can measure a sample without user interaction. Bioimpedance takes several seconds to measure so it is *Quasi-Efficient-to-Use.*

Deployability

Bioimpedance recognition can alter due to physical injury so *Quasi-Accessible.* Sensors and the form factor to measure bioimpedance is not widely available so it not a *Negligible-Cost-per-User.* Using Bioimpedance as a biometric is fairly new so it is not *Browser-Capable,* not *Mature*, and *Requires-Additional-Sensors.*

Security

Since bioimpedance recognition requires physical contact to the surface of the skin, it is infeasible for an adversary to gather bioimpedance information without physical contact. Thus, it is *Resilient-to-Physical-Observation, Resilient-to-Targeted-Impersonation,* and *Resilient-to-Theft.* Since the bioimpedance sensors are worn and sampled passively, it *Quasi-Requires-User-Consent*.

Ear Recognition

Ear recognition is a physiological biometric that examines the structure of the ear. Traditionally this biometric has relied upon image-based techniques to recognize subjects.[5] More recent research has examined acoustic-based techniques to sense both the inner and outer parts of the ear.[4] One could imagine, for example, headphones or earbuds integrating this biometric to identify the person wearing them. Because acoustic-based ear recognition seems the most promising way of accomplishing this biometric in a wearable, we survey that technique only.

Acoustic methods for sampling the ear require a speaker and microphone. The speaker generates a tone at some frequency and the microphone picks up this tone as altered by the either the inner or outer ear. Different frequencies yield will provide better discrimination based upon the structures of the ear being examined. One can view the ear as a kind of impulse response filter to the generated tone.

Given this impulse response, the phase is often removed and only the amplitude at each frequency is used. Removing the phase shift reduces intra-subject variability.

Acoustic-based ear recognition systems performed at 1–10 percent EER depending upon the kind of the device used to measure the biometric (headphones, earbuds, mobile phone). However, these results are for a relatively small population (<50).[4] The benefit of the acoustic-based ear recognition system is that it can be easily integrated into some wearable devices and it would be difficult for an adversary to capture the biometric or spoof the biometric.

"…acoustic-based ear recognition system is that it can be easily integrated into some wearable devices and it would be difficult for an adversary to capture the biometric or spoof the biometric."

Usability

Ear recognition is *Physically-Effortless* because the user does not need to interact with the device—just have the sensors placed in/around the ear. It is Quasi-*Efficient-To-Use* since the time to capture the impulse response is quick.

Deployability

Recognizing a user's ear is *Quasi-Accessible* since it may not be applicable to those who have physical injury in or around the ear. Similar to other biometrics, transmitting acoustic ear data is nontrivial through a web browser so it is not *Browser-Capable*. Ear recognition is not *mature* and requires more extensive study to be widely deployable.

Security

Since we assume acoustic-based recognition, it may be difficult for an adversary to capture acoustic data on a user's inner ear. We therefore conclude that ear recognition is *Resistant-to-Physical-Observation*, *Resistant-to-Targeted-Impersonation*, and *Resistant-to-Theft*. Acoustic-based ear recognition is *Quasi-Requiring-User-Consent* since an earpiece may capture the sample passively without the user's knowledge.

Touch Recognition

Touch dynamics is a behavioral biometric that analyzes the way a user interacts with a capacitive touch surface. The inspiration for touch dynamic biometrics builds from other user-interface–based biometrics such as keyboard dynamics and mouse dynamics. Touch dynamics extracts features from user interaction on a capacitive touch surface. Features are drawn from single input methods such as a touch, tap, or swipe, or multitouch gestures such as pinch or two-finger swipe.

Touch dynamics consisting of single-input touch gestures are shown to identify users when coupled with gyroscope and accelerometer. User touch screen interaction usually includes tap, scroll, and fling (such as flipping between pages on an e-book app). The features extracted from these interactions include *touch* features such as screen coordinates, duration, and touch pressure as well as *reactionary* features, which include device positioning and amplitude changes caused by the user touching the device. The advantage of this biometric is its quasi-passive sampling. As the user interacts with the system, the confidence of identifying the user increases (after 10 user interactions, the FAR and FRR are less than 1 percent). Other work utilizes multitouch gestures as a behavioral biometric. These systems tend to perform in the 5–10 percent EER range.

"It is possible to have a touch interface without a screen, and whether these interfaces also contain distinctive sets of features is an open research question."

There are a number of issues to consider when utilizing touch dynamics on wearable or client devices. They usually require additional inputs to accurately identify users. Some methods[1] require measurements from gyroscope and accelerometer sensors. Other methods couple touch dynamics with a pattern unlock[2], which may be infeasible on small touch screens. It is possible to have a touch interface without a screen, and whether these interfaces also contain distinctive sets of features is an open research question.

Usability

Touch dynamics recognition is *Quasi-Physically-Effortless* since it requires a user to interact with a device's touch surface. It is not *Efficient-to-Use* because it may require the user to input multiple touch events in order to properly recognize the user.

Deployability

Touch dynamics is *Quasi-Accessible* because certain physical injuries may prevent a user from interacting on a touch surface. It has *Negligible-Cost-Per-User* since touch surfaces are commonly found on mobile devices. Further, touch dynamics is *Browser-Capable* because touch events could be sampled through a web browser. However, touch dynamics at this time is not *Mature* and requires large scale user studies.

Security

Touch dynamic recognition is *Quasi-Resilient-to-Physical-Observation* because an adversary may be able to observe multiple touch inputs and gestures in attempts to learn the behavior. Further, touch dynamic recognition is not *Resilient-to-Targeted-Impersonations* because information such as handedness (whether a person is left or right handed) may be acquired by an adversary and used to guess a user's touch gestures. It is not *Resilient-to-Theft* because an adversary may learn a user's touch dynamics through observation. Since the biometrics requires user interaction with a touch surface, it *Requires-User-Consent.*

Conclusion

Of the biometrics we surveyed, no one biometric stands out as a clear winner for wearable or client-based devices. They each need to be evaluated in the context of some application. It may even be beneficial to fuse multiple biometrics to provide better security, usability, or deployability in your application. We hope that this survey can provide you with the guidance to decide which biometric is appropriate for your application.

Complete References

[1] Bo, C., L. Zhang, and X Y. Li, "SilentSense: Silent User Identification via Dynamics of Touch and Movement Behavioral Biometrics," ArXiv preprint arXiv:1309.0073.

[2] Angulo, J. and E. Wästlund, "Exploring Touch-screen Biometrics for User Identification on Smart Phones," *Privacy and Identity Management for Life IFIP Advances in Information and Communication Technology*, Volume 375, 2012, Pages 130–143.

[3] Sae-bae, N., K. Ahmed, K. Isbister, and N. Memon, "Biometric-Rich Gestures: A Novel Approach to Authentication on Multi-touch Devices," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, Pages 977–986.

[4] Akkermans, A., T. Kevenaar, and D. Schobben, "Acoustic ear recognition for person identification," in *Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, 2005, Pages 219–223.

[5] Hurley, D. J., M. S. Nixon, and J. N. Carter, "Force field feature extraction for ear biometrics," *Computer Vision and Image Understanding*, Volume 98, Number 3, June, 2005, Pages 491–512.

[6] Nickel, Claudia, Mohammad O. Derawi, Patrick Bours, and Christoph Busch, "Scenario test of accelerometer-based biometric gait recognition," in the Third International Workshop on Security and Communication Networks, May 2011.

[7] Bonneau, Joseph, Cormac Herley, Paul C. van Oorschot, and Frank Stajano, "The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes," in *Proceedings of the 2012 IEEE Symposium on Security and Privacy*, Pages 553–567.

Author Biographies

Cory Cornelius is a research scientist in Intel Labs working on security and privacy research. He recently completed his doctorate at Dartmouth College in Computer Science. His thesis is titled "Usable security for wireless body-area networks." His research interests include wearable biometrics for both authentication and health monitoring. He can be reached at Cory.Cornelius@intel.com.

Christopher Gutierrez is a third-year computer science PhD student at Purdue University. He is a research assistant in the Dependable Computing Systems Laboratory and is currently researching methods to detect and prevent spear phishing attacks. His research interests include authentication, advanced persistent threat protection, and digital watermarking. He can be reached at gutier20@purdue.edu.

ESTIMATING THE UTILITY OF USER-PASSIVE AUTHENTICATION FOR SMARTPHONE UNLOCK

Contributor

Micah Sheller
Intel Labs

"…user-passive authentication requires data that may not always be available at the time of authentication."

The sensing capabilities of smartphones present new opportunities for novel user-passive authentication methods. However, user-passive authentication requires data that may not always be available at the time of authentication. For example, speaker verification requires the user to speak. This article estimates the utility of four passive authentication methods relevant to smartphones—gait recognition, speaker verification, touch dynamics recognition, and phone/PC collaboration—by examining data we collected on five users over one month. Specifically, we monitored the availability of data required for our chosen passive authentication methods: user walking for gait recognition, someone talking for speaker verification, user touching the screen for touch dynamics recognition, and user PC unlocked for phone/PC collaboration. We compare these availability intervals against phone unlock events to determine whether the related passive authentication could have been used to unlock the phone. This article presents our initial findings toward understanding the potential value of these passive authentication methods for unlocking smartphones. We estimate that for our users, a combination of gait recognition, touch dynamics recognition, and phone/PC collaboration could reduce user-active phone authentication for phone unlock by approximately half. We also estimate that another one-quarter of phone unlocks occur roughly eight minutes after passive authentication was last available.

Introduction

The wealth of data available to smartphones about user behavior, environment, and other networked devices, hereafter *user context* or *context*, presents opportunities to develop new methods to relieve the burden of user authentication for smartphone access, such as PIN or password lock screens. Behavioral biometrics solutions such as gait recognition[1][3], speaker verification[2][5], and input dynamics[2][4] continue to mature. Trusted devices can collaborate, exchanging information about their respective locations and detected users, or acting as authentication factors.

Authentication methods require a credential to check, regardless of whether they require user attention. The utility of a passive authentication method, that is, a method requiring no user interaction, relates to how often the credential is *available* to be verified, hereafter *available*. For example, gait recognition provides no value to a system if the user never walks when the system needs an authentication. Just as gait recognition requires the user to be walking, speaker verification requires the user to be talking, touch screen dynamics recognition requires the user to be touching the screen, and trusted device collaboration requires the devices have relevant information available

to share (for example, one trusted device might authenticate the user on behalf of another).

We conducted a study of five users for approximately one month (136 total days of user data) to estimate the *availability* of gait recognition, touch screen dynamics recognition, speaker verification, and phone-PC device collaboration. We installed custom monitoring software on their smartphones to detect conditions necessary for each of these passive authentication methods. The detection conditions do not guarantee *availability*, however, but they serve as good estimations for *availability* upper bounds. To estimate an *availability* upper bound for gait recognition, we monitored whether the user was walking. To estimate an *availability* upper bound for touch recognition, we recorded when the user touched the screen. To estimate an *availability* upper bound for speaker verification, we monitored whether someone could be heard talking (we were restricted to general human speech detection for privacy reasons). To estimate an *availability* upper bound for PC authentication collaboration, we analyzed the users' Windows* security logs to determine when the users' PCs were in use. We then measured how often the users authenticated to their phones during a period of estimated *availability* as a first estimate of the utility of the related passive authentication method.

"We conducted a study of five users for approximately one month (136 total days of user data) to estimate the availability of gait recognition, touch screen dynamics recognition, speaker verification, and phone-PC device collaboration."

Our second estimation of the utility of these passive authentication methods involved examining the relationships between phone unlocks and past *availability,* that is, if the user was not walking at the time of unlock, we determined how long ago the user had been walking. Past authentications have value. In a typical system, a password is entered at the start of the session and the session considered valid as long as the system detects the user is still present, such as detecting user input. We believe future authentication models will combine passive authentications and improved user presence detection to maintain validity of past passive authentications in order to avoid user-active authentications. For example, if a system passively authenticates a user, say by gait recognition, then successfully detects that user's presence for the next five minutes, that gait recognition could be considered valid for that entire period of positive user presence detection.

"We believe future authentication models will combine passive authentications and improved user presence detection to maintain validity of past passive authentications in order to avoid user-active authentications."

To understand how past passive authentications might help avoid user active authentications, we present the distributions of time since we last estimated a context was *available* at the time of phone unlock (only for those unlocks that occurred when the context was estimated as *unavailable*). For example, we show that 37 percent of one user's phone unlocks occurred less than six minutes after detecting walking (the 37 percent does not include unlocks that occurred while walking). Thus, a system capable of maintaining the validity of a past authentication for six or more minutes could potentially use gait recognition to passively authentication an additional 37 percent of that user's sessions than if it did not make use of past authentications.

We also present our method for detecting each of our contexts: user walking, someone talking, user touching screen, user PC unlocked.

"…a system capable of maintaining the validity of past authentications for eight minutes could potentially replace 75 percent of our users' phone unlock authentications with one of our four passive authentication methods."

We measured a user average 50 percent of phone unlocks occurring during one of our *availability* estimation contexts, implying that about half of our users' phone unlocks could be covered by one of our passive authentication methods. We also found that half of the remaining phone unlocks occurred within a user average eight minutes of one of our *availability* estimation contexts, implying that a system capable of maintaining the validity of past authentications for eight minutes could potentially replace 75 percent of our users' phone unlock authentications with one of our four passive authentication methods.

Method

This section describes our data collection, goals, subjects (hereafter users), assumptions, software, and context detection method. We collected a total of 136 days of good data, approximately 27 days per user.

For this research, we focused on the hours between 7 am and 7 pm, Monday through Friday, because these hours cover our users' typical workdays. We leave nights and weekends for future work, as they imply considerably different user behavior patterns and threat models.

The user population for this study was limited to a small group of five Intel research scientists, all males between 22 and 50 years old, who were involved, aware, and informed about the sensor data collected. Though lacking diversity, the users' awareness of the research space and familiarity with the prototype tools involved assisted this feasibility study with collecting a more reliable data set.

We supplied each user with a Nexus 5 smartphone running Android 4.4.2, loaded with custom software to gather and upload measurements. We ensured our application would not require users charge more often than an overnight charge to minimize impact on user behavior.

We instructed each user to adopt the phones as their personal devices in order to better capture genuine user behavior. Additionally, the users were given the devices early to acclimate prior to data collection.

In addition to the data gathered from the users' phones, we conducted short interviews with each user while examining his data. These interviews focused on verifying the surface validity of the data collected and augmenting our understanding of user differences.

"Our study sought to estimate the utility of gait recognition, touch recognition, speaker verification, and phone-PC device collaboration as passive authentication methods…"

Goal

Our study sought to estimate the utility of gait recognition, touch recognition, speaker verification, and phone-PC device collaboration as passive authentication methods for the purpose of eliminating user-active authentications at phone unlock, such as PIN entry. We wanted to understand this utility in the context of two kinds of system: systems that require current authentications and systems that may accept past authentications as valid, according to some policy (defining such policies was out of scope for this study).

Assumptions

To avoid troubleshooting devices, we assumed all phone failures resulting in data gaps, such as OS crashes, resulted from a bug in our software. Thus, we omitted these data gaps from our analysis, as opposed to treating them as natural gaps in data availability.

We assumed that user behavior during the first week of ownership differs from behavior after the first week, especially when acclimating to a new version of Android, as our subjects did. For this reason, we gave our subjects their phones at least one week before we began data collection.

"We assumed that user behavior during the first week of ownership differs from behavior after the first week, especially when acclimating to a new version of Android, as our subjects did."

Limitations

We required each subject use a Nexus 5 phone running the commercial build of Android 4.4.2 to avoid validating our software on a wide range of phones and OS builds.

Our touch detection does not work while the phone screen is off. Our touch detector does not detect what kind of touch event occurred, such as a tap or a swipe.

We originally included passive camera data collection for the front-facing camera, but removed it due to stability problems.

Recording audio disables speech-to-text, so we limited our audio sampling to one second every ten seconds.

Our subjects were all drawn from among our coworkers and were all male. Thus, our demographics are restricted to men in their 20s to 40s working at Intel, in the same building and in the same research domain.

Logging Software

For walking detection, we leveraged Android 4.4.2's sensor batching interface to allow collection from the Android step counter sensor even when the phone slept.

We logged touch events via a transparent view drawn over the screen and configured with FLAG_WATCH_OUTSIDE_TOUCH, an Android API level three setting that allows the software to receive notification whenever a touch event occurs outside the software's UI. These outside touch events contain no detail, such as touch coordinates or whether the touch was a drag or tap, we believe in order to prevent software such as key loggers. This lack of detail limited our data to touch counts only. We logged these counts at 10-second intervals only while the phone screen was on. We did not miss samples, but our logging rate limited our timing precision to 10-second granularity.

We logged Android system Intents for screen on, screen off and phone unlock.

We used Android's AudioRecord API to record audio every 10 seconds for 1 second.

We configured our software to run as a service, starting on boot and automatically restarting in the case of a crash.

User Context Definitions and Detection Methods

This section explains how we defined/detected our user contexts, that is, how we mapped our input data to the related user context intervals.

All context detection intervals were cropped in the event that the phone logging software was not running.

User Walking

We defined *user walking* as any span of greater than ten seconds in which all steps are at most three seconds apart. We based these timing choices on recommendations from in-house domain experts and literature review.[1][3] Though gait recognizers may require more than ten seconds of data, we intended *user walking* to serve as an *availability* upper bound for gait recognition, so we chose a more generous value.[1]

"Though gait recognizers may require more than ten seconds of data, we intended user walking to serve as an availability upper bound for gait recognition, so we chose a more generous value."

User Touching Screen

We defined *user touching screen* as any consecutive samples of more than one touch each. Our software was limited to a polling rate of 10 seconds. Therefore, touch events as long as 20 seconds apart could have been considered consecutive, depending on where the events fell within their respective 10-second polling windows.

Someone Talking

To detect *someone talking*, we captured audio for one second every 10 seconds. We then applied an in-house audio environment classifier to emit a binary value of whether human speech was detected. Note that human speech can come from many sources: other people, TV, radio, and so on, and our solution does not distinguish voices from different individuals.

Our sampling limitations required that we interpolated speech detection intervals based on only one second of data every ten seconds. Our interpolation method was to extend the value of each sample across the time window up to the next sample.

User PC Unlocked

PC unlocks were not detected directly by the phone. Rather, we collected security logs from each user for the dates of the study. These security logs hold the times when the PC was either unlocked or locked. We defined *user PC unlocked* as the time between a pair of unlock/lock security log timestamps. All of our PCs ran either Microsoft Windows* 7 or Microsoft Windows 8.

Results

"Our data consists of 136 weekdays from 7 am to 7 pm each day, spread across five different users."

Our data consists of 136 weekdays from 7 am to 7 pm each day, spread across five different users. In this section, we present the percentages of phone unlocks that occurred during each context. We also present a distribution of time between each context and phone unlock for those cases where the context was not detected at the time of the phone unlock.

The percentages of phone unlocks that occur during a detected context gives us a baseline for the utility of that context's related passive authentication mechanism. For example, one user was walking 27 percent of the time that he unlocked his phone. This implies that passive gait recognition could potentially reduce that user's phone unlock authentication burden by 27 percent.

The distribution of time since last detected *availability* at time of phone unlock estimates the potential value of past authentications. For example, for one user, 37 percent of his phone unlocks occurred while not walking, but still less than six minutes after his last detected *user walking* context. This implies if the system could maintain the value of an authentication for six minutes, passive gait recognition could potentially reduce that user's phone unlock authentication burden by 64 percent (27 percent from current gait recognitions, 37 percent from past gait recognitions).

Percentage of Phone Unlocks during Detected Contexts

Figure 1 shows how often the user unlocked his phone during our detected *availability* estimation contexts.

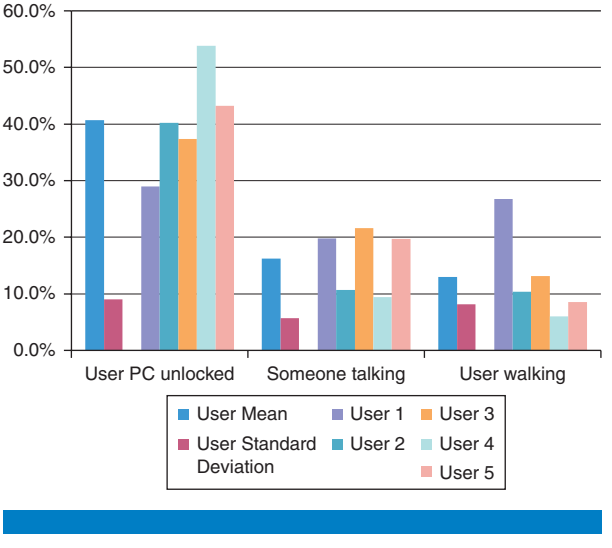


Figure 1: Percentage of phone unlocks that occurred during *user PC unlocked*, *someone talking*, or *user walking*
(Source: Intel Corporation, 2014)

Note that touch is omitted because we could not detected it while the phone was locked, as mentioned earlier.

On average from among users, *user PC unlocked* covered 40.7 percent of phone unlocks, more than twice the user averages for *user walking* and *someone talking*, which we estimated at 13 percent and 16.2 percent, respectively. These values imply that for passive authentications immediately prior to phone

"On average from among users, user PC unlocked covered 40.7 percent of phone unlocks, more than twice the user averages for user walking and someone talking, which we estimated at 13 percent and 16.2 percent, respectively."

unlock, of our four passive authentication methods, our users would benefit most from phone/PC collaboration. They also imply that gait recognition and speaker verification would be poor standalone options for such authentications. We expect that few authentication solution designers would consider the kind of paradigm shift implied by passive authentication for such a small reduction in PIN entries.

"…we see the highest variation among users in our user walking context, with a standard deviation of over half the mean."

Looking at the user standard deviation values in Figure 1, we see the highest variation among users in our *user walking* context, with a standard deviation of over half the mean. In the user interviews, we asked users whether they take frequent walk breaks, to which three answered yes, and whether they often use their phones while walking, to which two answered yes. User 1 was the only user to answer yes to both questions and recorded the highest percentage of phone unlocks during *user walking.* Similarly, user 4 was the only user to answer no to both questions recorded the lowest percentage of phone unlocks during *user walking*. Users 2 and 5 both answered yes to frequent walking and no to phone use while walking. User 3 answered no to frequent walking and yes to phone use while walking.

Our data shows a user standard deviation for percentage of phone unlocks during *user PC unlocked* of less than one-fifth of the mean. This is considerably less variation than *user walking*. From our interviews, our user variation for phone unlocks during *user PC unlocked* matched our answers to whether the user often used his phone while walking. Those that answered yes (users 1 and 3) averaged 33.2 percent, while those that answered no (users 2, 4, and 5) averaged 45.8 percent.

Our data shows a user standard deviation for percentage of phone unlocks during *someone talking* of approximately one-third of the mean. All of our users work in a cubicle environment in the same building. In our user interviews, we recorded which users sat in cubicles next to a primary aisle, that is, near the center of the floor or near an entry or exit, and which users did not. Location of cubicle matched the differences between users in percentage of phone unlocks during *someone talking*: those that sat next a primary aisle averaged 20.4 percent, while that that did not averaged 10.1 percent.

Our passive authentication methods need not be used standalone. Figure 2 shows the same kind of data as Figure 1, but for various unions of our *availability* estimation contexts used together. For example, the category *user walking* or *someone talking* shows the percentages of phone unlocks that occurred during either of the two contexts.

In addition to the set of all three *availability* contexts, we also show data for the set without *someone talking* and the set without *user PC unlocked*. We chose to show these because text-independent speaker verification and phone/PC authentication collaboration are immature relative to gait recognition.

"All of our users recorded over 50 percent of phone unlocks during one of our availability contexts."

All of our users recorded over 50 percent of phone unlocks during one of our *availability* contexts. When we omit *someone talking*, our user average drops

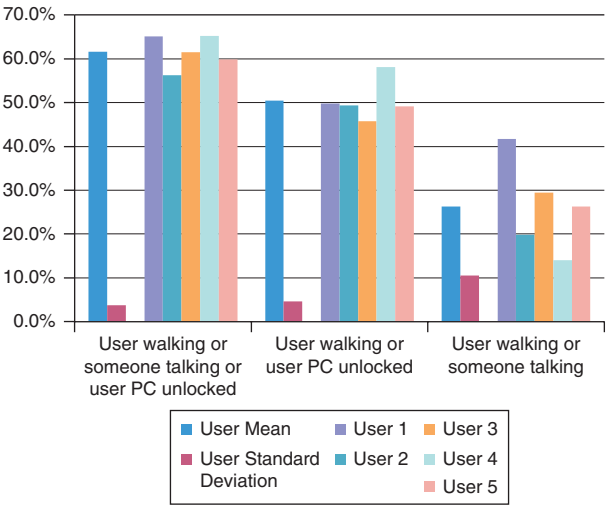


Figure 2: Percentage of phone unlocks that occurred during various unions of user PC unlocked, someone talking, or user walking
(Source: Intel Corporation, 2014)

from 61.6 percent to 50.4 percent. When we omit *user PC unlocked*, our user average drops to 26.3 percent.

These drops show that our *availability* contexts at time of phone unlock overlap less than randomly. For example, our value of 61.6 percent for the set of all three contexts is approximately 5 percent greater than random, which we calculate would be approximately 56.7 percent by the multiplication rule for independent events. This implies that gait recognition, phone/PC collaboration, and speaker verification each tend to address different user patterns, such that there is little redundancy between them.

We see some overlap between *user walking* and *user PC unlocked*: the sum of the user averages for *user walking* and *user PC unlocked* are slightly higher than the user average for the union of *user walking* and *user PC unlocked*: 53.7 percent and 50.4 percent, respectively. We believe this represents times when users forgot to lock their PCs before walking away from them.

Figure 2 shows much lower user standard deviation values (relative to the mean) than Figure 1. This suggests that a passive authentication model that uses each of our authentication methods would perform much more consistently across users than a model that relies on only one of our authentication methods.

From our user interviews, we found that our users who reported that they often used their phones while walking, users 1 and 3, also recorded the lowest percentage of phone unlocks covered by *user PC unlocked*. We believe this explains the low user standard deviation for phone unlocks during *user walking* or *user PC unlocked* of less than one-tenth the mean. The differences in user behavior between the two contexts offset each other.

"This implies that gait recognition, phone/PC collaboration, and speaker verification each tend to address different user patterns, such that there is little redundancy between them."

"This suggests that a passive authentication model that uses each of our authentication methods would perform much more consistently across users than a model that relies on only one of our authentication methods."

"In models where passive authentication must occur immediately prior to phone unlock, our estimates imply that using any of our passive methods alone would be insufficient and would vary considerably between users, while the combination of gait recognition, speaker verification, and phone/PC collaboration shows considerable promise for all of our users."

In models where passive authentication must occur immediately prior to phone unlock, our estimates imply that using any of our passive methods alone would be insufficient and would vary considerably between users, while the combination of gait recognition, speaker verification, and phone/PC collaboration shows considerable promise for all of our users.

Time Since Context at Phone Unlock

For systems able to use passive authentications some time prior to phone unlock, such as a system capable of monitoring whether the same user is still present, our analysis must also consider *availability* intervals that occur before phone unlock. Figure 3 shows a key metric for estimating the value of such systems. For example, the sub-chart of *user walking* tells us how long ago the user was last walking for cases where he unlocked his phone while *not* walking. This allows us to estimate how many phone unlocks gait recognition could cover if our system accepted gait recognition results from some maximum time in the past.

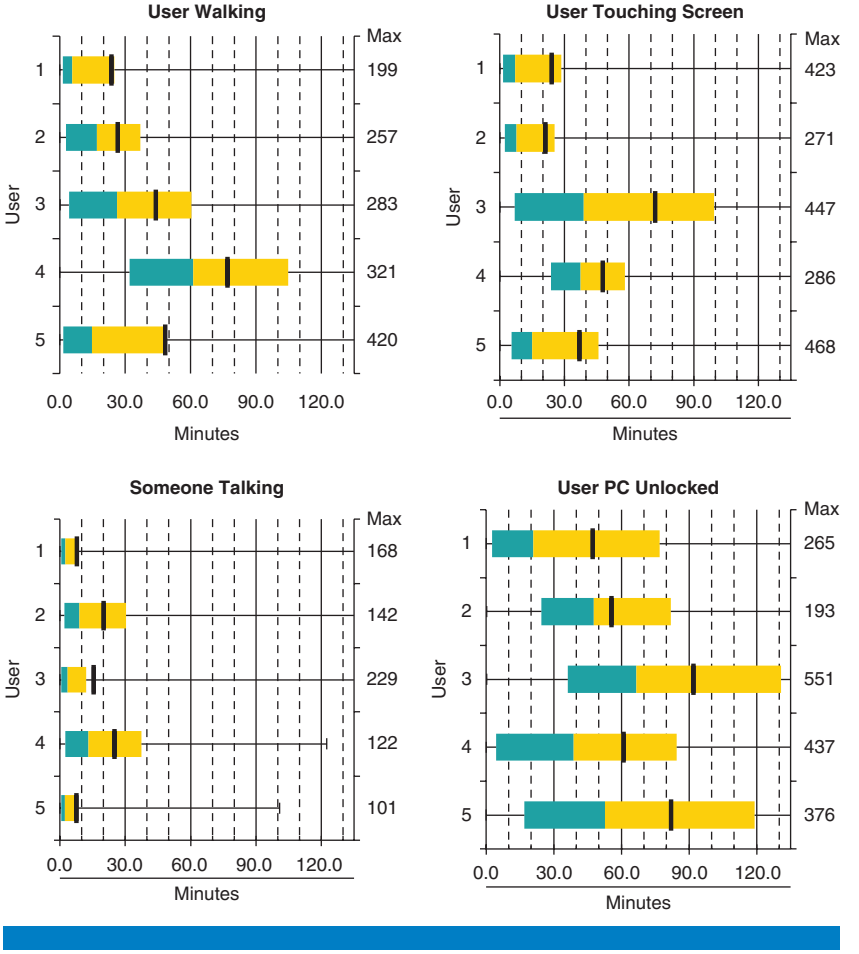


Figure 3: Quartile boxplots of time since context at phone unlock for phone unlocks occurring when the context was not detected
(Source: Intel Corporation, 2014)

Note that these distributions are for unlocks not already covered at the time of unlock.

The *someone talking* subplot shown in Figure 3 shows a remarkably similar distribution between users 1 and 5. From our interviews, we know that these users sit close to one another, but they do not work directly with each other, nor do they talk more than a few times per day. This implies that most of the human speech detected comes from coworkers in nearby cubicles.

Used alone, our data suggests that to cover more than 75 percent of phone unlocks with passive gait recognitions would require maintaining the validity of past authentications as old as one hour, or nearly two hours in the case of user 4. We know this from the third quartiles. We see nearly identical results for touch dynamics recognition, though the outlier user is user 3 instead of user 4.

"Used alone, our data suggests that to cover more than 75 percent of phone unlocks with passive gait recognitions would require maintaining the validity of past authentications as old as one hour,…"

Because *user PC unlocked* already covers a significant number of phone unlocks at the time of unlock, we can achieve a coverage close to 75 percent of phone unlocks by accepting *user PC unlocked* intervals as old as the medians in Figure 3. For example, *user PC unlocked* covers 40.2 percent of user 2's phone unlocks at the time of phone unlock. Thus, for user 2 and *user PC unlocked*, the median of the distribution shown in Figure 3 represents another 30 percent of unlocks in addition to the 40.2 percent, for a total of 70 percent. Even so, we still see that 75 percent coverage again requires accepting authentications as old as one hour for most users.

Again, we turn to using our passive authentication methods in concert, rather than by themselves. Figure 4 shows the same kind of data as Figure 3, but for unions of our contexts.

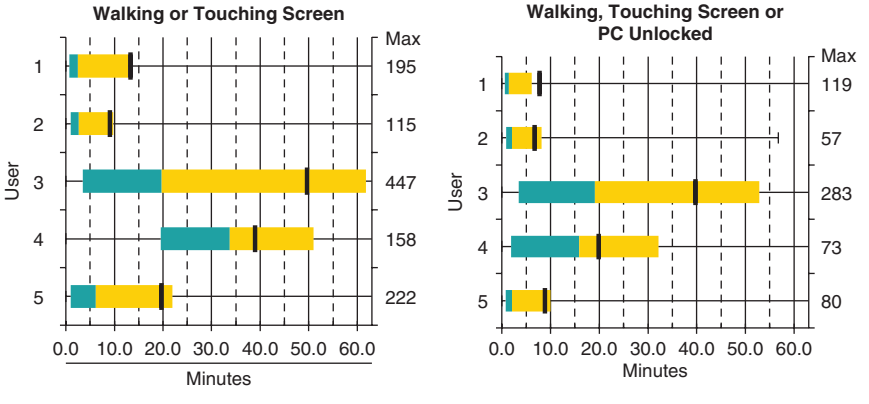


Figure 4: Quartile boxplots of time since one of a context union at phone unlock for phone unlocks occurring when none of the contexts in the union were detected

We omit *someone talking* from these unions due to concerns about the validity of the data. Unlike our analysis at time of phone unlock, this analysis relates more strongly to the frequency of contexts than the duration of contexts. We

believe that in terms of frequency, our data shows that *someone talking* is an unusably high estimation of speaker verification availability.

Like our standalone analysis of *user PC unlocked*, the medians in Figure 4 represent a total phone unlock coverage of approximately 75 percent, as approximately 50 percent of phone unlocks are already accounted for at the time of phone unlock, that is, our results from Figure 2. Similarly, our third quartiles represent approximately 88 percent of phone unlocks.

Figure 4 shows stark contrast between our users who reported themselves as frequently taking walks during the day (users 1, 2, and 5) and our users who reported otherwise (users 3 and 4). However, *user walking* cannot completely explain the difference, as the inclusion of *user touching screen* improves significantly over the standalone *user walking* shown in Figure 3.

"For our frequent walkers, Figure 4 suggests that if the phone can maintain the validity of authentications up to two minutes in the past, a combination of gait recognition, touch dynamics analysis, and phone/PC collaboration could feasibly replace up to 75 percent of user active authentications for phone unlock."

For our frequent walkers, Figure 4 suggests that if the phone can maintain the validity of authentications up to two minutes in the past, a combination of gait recognition, touch dynamics analysis, and phone/PC collaboration could feasibly replace up to 75 percent of user active authentications for phone unlock. If past authentication validity can be maintained for up to ten minutes, this same combination of passive authentication methods could feasibly replace up to 88 percent of these users' active authentications for phone unlock.

For users 3 and 4, Figure 4 suggests the need for maintaining authentication validity for as long as twenty minutes to achieve 75 percent phone unlock coverage, an order of magnitude more than for the other users.

Discussion

"Replacing active authentications with passive authentications presents risk in terms of user confusion."

Replacing active authentications with passive authentications presents risk in terms of user confusion. Inconsistent user experience frustrates users if they cannot understand the reasons for it. We do not believe that a smartphone that appears to randomly require a password 80 percent of the time would be a pleasant user experience. Therefore, we need to understand approximately how many active authentications need to be replaced in order to offset the inconsistent nature of passive authentication. Until then, we rely on intuition to interpret our numbers.

Thus, we make what we believe to be a conservative claim: a replacement rate of 90 percent would provide a good user experience, even if users were unable to understand why they still needed to actively authenticate 10 percent of the time.

Models Using Past Authentications

In this section, we discuss models that allow the use of past authentications for phone unlock, whether by timeout, monitoring that the user is still with the phone, or some other policy.

With a goal of 90 percent in mind, our data suggests that passive authentication on smartphones requires some method for preserving the

validity of passive authentications that occurred some number of minutes in the past. For three of our users, for an authentication model that uses gait recognition, touch dynamics recognition, and PC collaboration, this number of minutes is only ten. If we accept a 75 percent replacement target, this number drops further to two minutes. This seems quite reasonable: two minutes matches the default idle timeout for the smartphones in our study. For the other two users, however, a 90 percent replacement target increases accepting authentications as old as an hour. This would surely require user presence monitoring of some kind.

The three users whose data suggests a two- to ten-minute authentication validity policy report themselves as frequent walkers, while the other two do not. However, walking does not fully explain the difference between these users, as Figures 3 and 4 show tremendous benefit from adding touch dynamics recognition and phone/PC collaboration to gait recognition for these users. This suggests that there is some other common user trait between these users other than walking.

Our study shows that for these models, the frequency of data availability matters as much as the total duration of data availability. We see this in Figures 1 and 3, which together estimate that when used by themselves, gait recognition, touch dynamics recognition, and PC collaboration all require maintaining authentication validity for up to an hour in order to achieve 75 percent *availability* coverage of phone unlocks, despite the self-evident fact that for our users, *user PC unlocked* covers much more of the workday than any of our other contexts.

Models Requiring Immediate Passive Authentication

Our data suggests that for authentication models requiring passive authentications immediately prior phone unlock won't likely cover much more than 50 percent of phone unlocks, and that such models require collaboration with the user's PC. However, such a model might still be feasible if it is consistent. It seems reasonable that users could understand a model where they do not have to enter their phone passwords while sitting in front of one of their unlocked PCs.

"Our data suggests that for authentication models requiring passive authentications immediately prior phone unlock won't likely cover much more than 50 percent of phone unlocks, and that such models require collaboration with the user's PC. However, such a model might still be feasible if it is consistent."

Figure 1 shows that these models get far more value from phone/PC collaboration than our other authentication methods. This suggests that these models benefit most from passive authentication methods when in terms of *availability*, duration matters more than frequency.

In general, our contexts appear to be negatively correlated with respect to phone unlocks. That is, they occur together at time of phone unlock less often than random. For authentication models using multiple passive authentication methods, negative correlation between passive authentication *availability* means better total *availability*, as when one authentication method is unavailable others are more likely to be available. However, if these systems intend to fuse simultaneous passive authentications for the purpose of a

"Thus, from our data, we see that gait recognition, touch dynamics recognition, speaker verification, and phone/PC collaboration work best together as a strategy to increase the availability of passive authentication, as opposed to a strategy to increase the strength of the authentication through multiple simultaneous authentications."

"Phone/PC collaboration requires device locality between that phone and PC. This requirement is nontrivial."

stronger level of authentication, negative correlation *reduces* overall *availability*. Thus, from our data, we see that gait recognition, touch dynamics recognition, speaker verification, and phone/PC collaboration work best together as a strategy to increase the *availability* of passive authentication, as opposed to a strategy to increase the strength of the authentication through multiple simultaneous authentications.

Challenges

Unfortunately, *someone talking* appears to generate frequent false positives for speaker verification in the form of nearby coworker conversations. We see this in the distributions for past *someone talking* for users 1 and 5. These are remarkably similar distributions. The commonality between these users is where they sit. Due to these frequent false positives, we had to omit *someone talking* from our analysis of systems that use past authentications.

Phone/PC collaboration assumes the PC authentication can be trusted throughout the PC session. We believe that a robust phone/PC collaboration system should include some form of passive authentication on the PC.

Phone/PC collaboration requires device locality between that phone and PC. This requirement is nontrivial.

Finally, our *user PC unlocked* suffered from occasional false positives when our users forgot to lock their PCs.

Future Work

Any future studies should include true text-independent speaker verification to replace *someone talking*. Additionally, sampling rates for audio and touch data should be increased.

PC unlock authentications should be analyzed in the same way as we have analyzed phone unlock authentications, that is, estimating when passive authentication methods are *available* on the PC and how the phone might be leveraged to passively authenticate to the PC.

The subject pool should be increased to include a larger variety of subjects, particularly women.

"A larger variety of data should be collected to find methods for monitoring user presence on smartphones for the purposes of maintaining the validity of past authentications."

A larger variety of data should be collected to find methods for monitoring user presence on smartphones for the purposes of maintaining the validity of past authentications.

Additional passive-authentication–related contexts should be studied, such as passive camera for *user face visible* or *user eyes visible*.

Usability research should be undertaken to better define targets for how many active authentications must be replaced to create a positive user experience, especially given the potential passive authentication to degrade user experience due to its inconsistent nature.

Future research should include robust methods for establishing device locality and trust between the phone and PC.

Summary

Our data shows that a combination of gait recognition, touch dynamics recognition, speaker verification, and phone/PC device collaboration could feasibly replace half of active phone unlock authentications (such as PIN or password entries) with passive authentications at the time of phone unlock. If the phone can maintain the validity of past authentications for up to two minutes, our data shows that a combination of gait recognition, touch dynamics recognition, and phone/PC device collaboration could feasibly replace 75 percent of active phone unlock authentications for three of our users. If the window of authentication validity can be increased to ten minutes, this percentage could feasibly increase to 90 percent. For the remaining two users, however, 75-percent passive replacement requires maintaining authentication validity for up to twenty minutes.

"If the phone can maintain the validity of past authentications for up to two minutes, our data shows that a combination of gait recognition, touch dynamics recognition, and phone/PC device collaboration could feasibly replace 75 percent of active phone unlock authentications for three of our users. If the window of authentication validity can be increased to ten minutes, this percentage could feasibly increase to 90 percent."

References

[1] Nickel, C., H. Brandt, and C. Busch, "Classification of Acceleration Data for Biometric Gait Recognition on Mobile Devices," *Proceedings of the Special Interest Group on Biometrics and Electronic Signatures (BIOSIG)*, 2011.

[2] Shi, W., J. Yang, Y. Jiang, F. Yang, and Y. Xiong, "SenGuard: Passive User Identification on Smartphones Using Multiple Sensors," *Proceedings of the IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob),* 2011.

[3] Mäntyjärvi, J., M. Lindholm, E. Vildjiounaite, S. Mäkelä, and H. Ailisto, "Identifying Users of Portable Devices from Gait Pattern with Accelerometers," *Proceedings of the IEEE 30th International Conference on Acoustics, Speech, and Signal Processing (ICASSP),* vol. 2, pp.973–976, 2005.

[4] Frank, M., R. Biedert, E. Ma, I. Martinovic, and D. Song, "Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication" *IEEE Transactions on Information Forensics and Security,* vol. 8, no. 1, pp. 136–148, 2013.

[5] Trewin, S., C. Swart, L. Koved, J. Matino, K. Singh, and S. Ben-David, "Biometric Authentication on a Mobile Device: A Study of User Effort, Error and Task Disruption," *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC)*, pp. 159–168, 2012.

[6] Cahill, C. P., J. Martin, M. W. Pagano, V. Phegade, and A. Rajan, "Client-based authentication technology: user-centric authentication using secure containers," *Proceedings of the 7th ACM workshop on Digital identity management (DIM)*, pp. 83–92, 2011.

Author Biography

Micah Sheller works as a senior software engineer and research scientist in the Intel Labs Security and Privacy Research lab. Micah joined Intel's Digital Home Group in 2005, where he worked on various media server optimizations. He then joined the Mobile Platforms Group, focusing on the USB3 specification, which names him as a technical contributor, and SSD caching technologies. In 2009, Micah joined Intel Labs to contribute to the Intel® SGX™ research effort, particularly in the spaces of media use models and the Intel SGX software runtime. Micah's research currently focuses on new methods for user authentication. Micah has patents pending in several security spaces such as management and sharing of digital information, user authentication techniques, and user authentication policy definition/management. He can be reached at micah.j.sheller@intel.com.

HETEROGENEOUS FACE RECOGNITION: AN EMERGING TOPIC IN BIOMETRICS

Contributor

Guodong Guo
West Virginia University

An emerging topic in biometrics is matching between heterogeneous image modalities, called heterogeneous face recognition (HFR). This emerging topic is motivated by the advances in sensor technology development that make it possible to acquire face images from diverse imaging sensors, such as the near infrared (NIR), thermal infrared (IR), and three-dimensional (3D) depth cameras. It is also motivated by the demand from real applications. For example, when a subject's face can only be acquired at night, the NIR or IR imaging might be the only modality for acquiring a useful face image of the subject. Another example is that no imaging system was available to capture the face image of a suspect during a criminal act. In this case a forensic sketch, drawn by a police artist based on a verbal description provided by a witness or the victim, is likely to be the only available source of a face of the suspect. Using the sketch to search a large database of mug-shot face photos is also a heterogeneous face recognition problem. Thus it is interesting to study the HFR as a relatively new topic in biometrics. In this article, several specific HFR problems are presented, and various approaches are described to address the heterogeneous face matching problems. Some future research directions are discussed as well to advance the research on this emerging topic.

Introduction

"Biometrics is about the identification of humans by their characteristics or traits, which include both physiological and behavioral characteristics."

Biometrics is about the identification of humans by their characteristics or traits, which include both physiological and behavioral characteristics. Physiological traits are related to the body shape, such as face, fingerprint, and iris, while behavioral characteristics are related to the pattern of human behavior, such as the typing rhythm, gait, and voice.

Because of the important and useful applications, such as identity management, law enforcement, and surveillance, biometrics has been an active research topic in the field of computer vision and pattern recognition.

Among various biometric traits, face recognition is one of the most challenging research topics, since there are many possible variations that affect the face matching performance. In traditional face recognition studies, the focus has been on addressing the changes and variations caused by human aging, head pose, illumination, and facial expressions, called A-PIE. Although significant progresses have been made especially for addressing the PIE problems, new challenges are emerging.

One of the emerging topics in face biometrics is matching between heterogeneous image modalities, called heterogeneous face recognition (HFR). This emerging

topic is motivated by the advances in sensor technology development that make it possible to acquire face images from diverse imaging sensors, such as the near infrared (NIR), thermal infrared (IR), and three-dimensional (3D) depth cameras. It is also motivated by the demand from real applications. For example, when a subject's face can only be acquired at night, the NIR or IR imaging might be the only modality for acquiring a useful face image of the subject. Thus it is interesting to study the HFR as a relatively new topic in biometrics.

In this article, several specific problems belonging to HFR will be presented in the section "Heterogeneous Face Recognition Problems," and different HFR algorithms and approaches will be introduced in the section "Heterogeneous Face Recognition Algorithms." Various HFR databases will be described briefly in the section "Heterogeneous Face Databases." Future research directions for HFR are discussed in the section "Some Thoughts on Future Directions. This is followed by "Concluding Remarks."

Heterogeneous Face Recognition Problems

Dictionary.com defines *heterogeneous* as "diverse in kind or nature." In the context of biometrics, heterogeneous face recognition (HFR) is to match face images coming from different modalities.[1] The motivation of the HFR is that face images of the same subject can often be captured by different sensors under different imaging conditions, because of the sensor technology development and broader application requirements.

"In the context of biometrics, heterogeneous face recognition (HFR) is to match face images coming from different modalities."

For example, the sensors can use different spectral bands: visible light spectrum (VIS), near infrared (NIR), and thermal infrared (IR); different content can be acquired: regular two-dimensional (2D) light reflection and three-dimensional (3D) depth data, especially the recently developed RGB-D sensors. Further, the cameras can have different qualities with different prices, for example, high-quality professional cameras, low-quality surveillance or web cameras, or photo scanners; and can be used in different acquisition environments: indoor/outdoor or different weather conditions (sunny, rainy, or snowy).

Therefore, in real applications, the probe and gallery face images may come from different image modalities. For instance, the still face images are usually used for face identity enrollment, while the face images from surveillance video cameras might be used for face matching or search over the still image database.

"Therefore, in real applications, the probe and gallery face images may come from different image modalities."

In this section, various HFR problems are discussed and presented, including both the basic problems that are clearly defined and have been studied in quite a few research works and some other HFR problems that have not been studied extensively.

Basic HFR Problems

The basic heterogeneous face matching problems include VIS vs. Sketch, VIS vs. NIR, VIS vs. 3D, and VIS vs. IR. These specific problems have been

clearly defined in previous research works[1][2], and are commonly admitted by researchers in biometrics.

Among the basic and typical heterogeneous face matching problems, VIS vs. Sketch and VIS vs. NIR are the mostly studied in the literature.

There are also approaches performing heterogeneous matching between thermal IR and VIS face images, for example, Li et al.[3], Choi et al.[4], Klare and Jain[2], and approaches to perform recognition between forensic sketches and visible face images[5][2], which is much more challenging than viewed sketches (drawn while viewing), because the drawn sketches can be obtained based on very limited information about the true identity, resulting in the sketches not being similar to the exact person. Compared to the popular VIS vs. Sketch and VIS vs. NIR, there are far fewer publications on VIS vs. 3D and VIS vs. IR matching, although these problems are also defined clearly as heterogeneous face recognition problems.

"Compared to the popular VIS vs. Sketch and VIS vs. NIR, there are far fewer publications on VIS vs. 3D and VIS vs. IR matching, although these problems are also defined clearly as heterogeneous face recognition problems."

There are several reasons why the problems of VIS vs. Sketch and VIS vs. NIR are more popular than others. One is that the high quality 3D range sensors and thermal IR cameras are still expensive, while the acquisition of NIR face images and face sketches does not need to involve expensive sensors. Thus it is relatively easier to collect data for research and practical applications, involving the Sketch, VIS, and NIR images. Another reason could be that it is more challenging to perform VIS vs. 3D or VIS vs. IR, since the image appearance differences between VIS and 3D or VIS and IR are significantly larger than between VIS and Sketch or VIS and NIR. As demonstrated by Goswami et al.[6], some photometric preprocessing of the images can help a lot to get high accuracies for heterogeneous face matching between VIS and NIR modalities. The matching between VIS and Sketch can also have very high accuracies.[7]

The forensic sketches are more challenging than the viewed sketch[5][2]; that is because the forensic sketches drawn by the forensic artists may not know (or the witness may not remember) the "full" face correctly, and thus the limited information can result in the drawn sketches not characterizing the true person well. In other words, it does not really mean that the sketches and VIS are very different modalities.

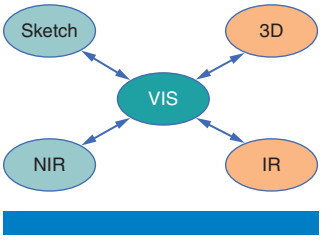


Figure 1: Some typical pairwise, heterogeneous face matching problems (Source: West Virginia University, 2014)

In addition to matching between VIS and other modalities, as shown in Figure 1, there is also heterogeneous matching between any pair of modalities in practice, such as NIR vs. 3D or NIR vs. IR, when the diverse sensors are used more and more in practical applications. To keep the graphic illustration clean, those pairwise matching relations are not shown in Figure 1.

In early studies, researchers usually only dealt with one specific HFR problem, for example, VIS vs. Sketch, while in recent studies, multiple HFR problems were studied to validate the developed methods in different cases.

Not only the basic HFR problems but also some other newly proposed problems can be classified as heterogeneous face matching tasks, which will be introduced next.

Other Heterogeneous Face Matching Problems

Some other face recognition problems in recent studies can be considered as heterogeneous face matching too. These atypical HFR problems include:

1. Matching between face images of different resolutions, that is, high-resolution and low-resolution.[8][9] For this kind of study, some existing face databases were used to "generate" face images at different resolutions. For example, the face images are cropped[9] as 32 32 and then down-sampled to 16 16, 8 8, 6 6, and 4 4. These down-sampled low-resolution face images were up-sampled into 32 32 to mimic the low-resolution face images for their study.
2. Digital photo vs. video frame.[9] Face images can be captured by digital still cameras or extracted from the video sequences captured by video camcorders. The faces from digital photos and video frames can have different resolutions and qualities. Thus face matching between digital photos and video frames can also be considered as a heterogeneous face matching problem.[9]
3. Face recognition with cosmetic changes.[10][11] This can be considered as another heterogeneous face recognition problem. As shown in Figure 2, face images of the same subject may look very different based on whether

"Thus face matching between digital photos and video frames can also be considered as a heterogeneous face matching problem."

Figure 2: Faces with makeup applied (left column) and faces with no makeup (right column) for the same individuals (each row). (Source: Originally shown in Wen and Guo[11], 2013)

"The matching between face images with or without makeup can be considered as another heterogeneous face recognition problem."

facial makeup is applied or not. The matching between face images with or without makeup can be considered as another heterogeneous face recognition problem.

Actually, it has been found that facial cosmetics can change the perception of faces significantly[12] and can bring a great challenge for face matching computationally.[13][14] Motivated by these studies, we have studied how to address the influence of makeup on face recognition based a dual-attributes approach[11], and a correlation-based approach.[10]

Heterogeneous Face Recognition Algorithms

"The key issue for heterogeneous face matching is how to reduce the difference between heterogeneous face images."

The key issue for heterogeneous face matching is how to reduce the difference between heterogeneous face images. Typically, there exist significant facial appearance changes between heterogeneous face images, even though the face images can be aligned well. The differences can be caused by the variety of sensors (for example, different spectral responses), different image acquisition conditions (for example, by physical devices or hand-drawing), or changes by the subjects themselves (for example, applying facial makeup). So the algorithm development for HFR usually focuses on various approaches to reduce the differences between heterogeneous face images of the same subjects.

Despite the significant progress that has been made for face recognition, most face recognition systems are not designed to handle HFR scenarios currently, including commercial off-the-shelf (COTS) systems. Therefore, there is a need and substantial interest for studying heterogeneous face matching problems.[2]

In this section, some representative approaches to HFR will be presented, based on a grouping into different categories.

Transforming One Modality to Another

To reduce the facial appearance differences between two modalities, one category of approaches is to transform the face images from modality *A* to another denoted by *B*, such that face matching can be executed using the "same" modality *B* approximately. This transformation can be in the raw image level or feature level. If it is in the image level, a new image will be synthesized in modality *B*, and then the image comparison is likely to use the same modality *B*; If it is in the feature level, the extracted features from image modality A will be transformed into features in domain *B*, and then compared to the features extracted directly from image modality *B*. This kind of approach is typically used to deal with VIS and sketch matching, where a face sketch can be synthesized from a photograph (or vice versa).[15][16][7] There are also some other methods proposed purely for sketch synthesis[17][18], which may be useful for matching VIS and sketch images.

A representative method to sketch synthesis from face photos is the eigen-transform method[15], which is similar to the eigenfaces method[19], but applied to two image modalities. The key idea is the sketch to be synthesized can be reconstructed based on the linear combination of a set of eigenvectors learned

from training sketch images, and the combination coefficients are equal to those learned from the corresponding face photo reconstruction. Thus, given a face photo, the reconstruction coefficients can be learned first and then applied to the sketch synthesis from sketch eigenvectors. After synthesis, the pseudo-sketch can be used to match against real sketches in the gallery for recognition.

Other approaches[20][21][22] use the idea similar to image analogies[23] to transform one modality to another, such as NIR to VIS or vice versa. One representative method[20] is to use local patches to build a dictionary for VIS and NIR faces separately and learn a linear combination of the nearest neighbors (similar patches) to reconstruct each patch for a given NIR face image. Then the learned linear reconstruction is applied to a new modality to synthesize a virtual VIS face for matching with other VIS images in the gallery.

Photometric Preprocessing

The second category of approaches to HFR is to use photometric preprocessing techniques to normalize the lighting or illumination in face images of each modality so that the differences between heterogeneous face images can be reduced. These preprocessing methods were originally developed to deal with illumination changes in visible light face images, but were then adapted to address the heterogeneous face matching problems, such as VIS vs. NIR face images. For these approaches, the underlying assumption is that the heterogeneity of face images is caused by the lighting or reflection differences in face surfaces.

"The second category of approaches to HFR is to use photometric preprocessing techniques to normalize the lighting or illumination in face images of each modality so that the differences between heterogeneous face images can be reduced."

Goswami et al.[6] gave a good summary of different photometric preprocessing techniques for HFR. Typically there are three different methods for photometric preprocessing, which will be introduced here:

One method is called sequential chain (SQ) preprocessing. It uses a series of steps for face image preprocessing. First, the Gamma correction is executed, which enhances the local dynamic range of the face image in darker regions, while compressing the range in bright and highlight regions. Second, the Difference of Gaussian (DoG) filtering is performed to compress the low frequency or nonessential information while maintaining or enhancing the gradient information that is more useful for recognition. Third, contrast equalization is used to rescale the intensity values globally and reduce the possibility of having extreme values during the processing in previous steps.

Another method is called single scale retinex (SSR). Usually the image intensity value, I, can be modeled as the product of illumination L and surface reflectance R. In the SSR method, the illumination component L is estimated by using the blurred image computed from the original face image. For example, the Gaussian filter can be used to compute the blurred image. Then the reflectance component R can be estimated by subtracting the illumination component from the original image in the logarithm domain. The SSR is applied to different modality images separately to compute the reflectance. The resulted reflectance images are assumed to be similar for heterogeneous face images, and are then used for feature extraction and matching.

The third method is called self quotient image (SQI). The SQI is very similar to the SSR operation. It is defined by the ratio between the original face image and a smoothed version of the original image, without using the logarithm computation. The ratio image is then used for feature computation and matching, replacing the original face image.

"Currently the photometric preprocessing methods are mainly used for VIS vs. NIR face images."

Currently the photometric preprocessing methods are mainly used for VIS vs. NIR face images. As shown in Figure 3, various photometric preprocessing methods can make the NIR and VIS face images look more similar. However, it is not clear if these methods are useful or not for other heterogeneous face matching problems, such as VIS vs. IR or VIS vs. 3D.

Figure 3: The effect of photometric preprocessing on heterogeneous face images (top: VIS, bottom: NIR); left to right: raw images, SQ, SQI, and SSR processing results.
(Source: Originally shown in Goswami et al.[6])

Another issue is that even though the photometric preprocessing can make the face images similar, it still needs feature mapping or other learning methods to further improve the performance for HFR in practice.

Common Subspace Projection

The third category of HFR approaches is to generate common subspaces so that both modalities of face images can be projected into, and the differences between heterogeneous images are expected to be minimized after the projection, as illustrated in Figure 4. New features can be generated after the joint projections into the common space.

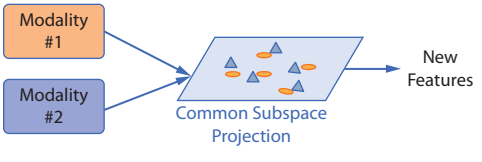


Figure 4: The common subspace projection to build the relationship between two different modalities of data and generate new features to minimize the differences
(Source: West Virginia University, 2014)

Classical methods to generate the common subspaces include the canonical correlation analysis (CCA)[24], and partial least squares (PLS).[25] These methods and their kernel versions for nonlinear mapping have been used for HFR, for example, by Sharma and Jacobs[8], Yang et al.[26], and Yi et al.[27]

Given face images corresponding to two different modalities, the CCA method can learn a pair of directions to maximize the correlation of the original data in the new subspace. The PLS is to learn a latent subspace such that the covariance between latent scores of the data from two modalities is maximized. Both the CCA and PLS methods can have linear mapping and kernel based extensions for nonlinear mapping.

"Both the CCA and PLS methods can have linear mapping and kernel based extensions for nonlinear mapping."

In addition to the classical methods, there are some other recent approaches to compute the common subspace in different ways. For example, Lin and Tang[28] proposed a method called Common Discriminant Feature Extraction (CDFE) for inter-modality face recognition. Two transforms are simultaneously learned to transform the samples in both modalities respectively to the common feature space. The learning objective incorporates both the discriminative power and local smoothness of the feature transformation.

Another method is the coupled discriminant analysis (CDA) by Lei et al.[9], which incorporates constraints such as locality information of the features and discriminative computation similar to the classical linear discriminant analysis (LDA), to improve the performance for heterogeneous face matching. More recently, the kernel-prototype–based similarity measure for HFR[2] was proposed, which pursues the kernel trick by Balcan et al.[29] to represent each face image with a set of training images, serving as prototypes.

Random Subspaces

The random subspace (RS) method by Ho[30] was developed to deal with the small sample size problem in recognition, using the idea similar to the classical bagging[31] and random forests[32] methods. The RS method is also useful to improve and generalize the classification performance, based on sampling a subset of features and classifier training in the reduced feature space. Then multiple classifiers can be learned from the multiple sets of randomly sampled features. These classifiers can be combined together to form a much stronger classifier or recognizer.

"The random subspace (RS) method by Ho was developed to deal with the small sample size problem in recognition, using the idea similar to the classical bagging and random forests methods."

Wang and Tang[33] used the random subspace with linear discriminant analysis (LDA) called RS-LDA for visible light face recognition. Klare and Jain[34] adapted RS-LDA for heterogeneous face recognition, by using multiple samplings of face patches from both VIS and NIR face images. The random subspace is also extended to the kernel prototype similarity measures[2] for HFR.

Dual Attributes

Attributes are a semantic level description of visual traits, as discussed, for instance, by Lampert et al.[35] and Farhadi et al.[36] For example, a horse can be described as four legged, mammal, can run, can jump, and so on. A nice property of using attributes for object recognition is that the basic attributes might be learned from other objects, and shared among different categories of objects.[37]

Facial attributes are a semantic level description of visual traits in faces, such as big eyes, or a pointed chin. Kumar et al.[38] showed that a robust face verification can be achieved using facial attributes, even if the face images are collected from uncontrolled environments over the Internet.

Motivated by the usefulness of facial attributes, a method called dual attributes was recently proposed by Wen and Guo[11] for face verification robust to facial appearance changes caused by the makeup. The key idea is that the dual attributes can be learned from faces with and without cosmetics, separately. Then the shared attributes can be used to measure facial similarity irrespective of cosmetic changes. In essence, dual attributes are capable of matching faces with or without makeup in a semantic level, rather than a direct matching with low-level features.

"The key idea is that the dual attributes can be learned from faces with and without cosmetics, separately."

The dual attributes method by Wen and Guo[11] may be adapted to other heterogeneous face matching problems.

Multiview Discriminative Learning

In the methods introduced above, typically only two modalities are used for HFR. Is it possible to deal with multiple modalities in the formulation? The answer is yes.

For example, the CCA can be extended to a multiview CCA by Rupnik and Shawe-Taylor.[39] Another way is to use the principle of LDA to derive a so-called multiview discriminant analysis (MDA) method by Kan et al.[40] It learns multiple view-specific linear transforms in a non-pairwise manner by optimizing a generalized Rayleigh quotient, that is, maximizing the between-class variations and minimizing within-class variations in a low dimensional subspace. The optimization problem is then solved by using the generalized eigenvalue decomposition technique.

Another method is the generalized multiview analysis by Sharma et al.[41], where the cross-view correlation is obtained from training examples corresponding to the same subjects or identities. This correspondence requirement is not needed in the MDA formulation.[40]

"These multiview analysis methods have been shown to be useful for some heterogeneous image matching problems, such as photo vs. sketch and VIS vs. NIR."

These multiview analysis methods[40][41] have been shown to be useful for some heterogeneous image matching problems, such as photo vs. sketch and VIS vs. NIR.

Heterogeneous Face Databases

To facilitate the study of heterogeneous face recognition, several databases have been assembled. A summary of the existing databases are presented in this section.

CUFS Database (Sketch-VIS)

This database was collected by the Chinese University of Hong Kong. The CUHK Face Sketch Database contains 606 subjects with VIS and sketch face

pairs.[7] There are 1,216 images in total. This is probably the first publicly available database for heterogeneous face matching.

CUFSF Database (Sketch-VIS)

This is an extended version of the CUFS database, containing 1,194 subjects with 2,388 image pairs of VIS and sketch by Zhang et al.[42] The sketch photos were drawn by artists when viewing the original face images for each subject. It is called *viewed sketches* by Klare and Jain[2] in contrast to the forensic sketches.

CASIA-HFB Database (VIS-NIR-3D)

This is probably the first database that contains more than two face modalities, assembled from the Institute of Automation, Chinese Academy of Sciences (CASIA) by Li et al.[43] It has 100 subjects of 992 face images in total. Each subject has four VIS, four NIR, and one or two 3D face images. The cropped face images were provided with the eye coordinates aligned manually. Some baseline results were provided based on direct matching with the classical PCA and LDA features. Later on, the database was extended to 202 subjects just for the VIS and NIR image modalities, resulting in 5,097 face images for VIS and NIR modalities.

"This is probably the first database that contains more than two face modalities, assembled from the Institute of Automation, Chinese Academy of Sciences…"

Cross-Spectral Dataset (VIS-NIR)

This dataset by Goswami et al.[6] contains VIS and NIR face pairs for 430 subjects over multiple sessions, collected from the University of Surrey in the United Kingdom. Different pose angles in pitch and yaw directions were captured for every 10 degrees. Each subject has at least three poses. In total, there are 2,103 NIR and 2,086 VIS face images. Twelve algorithms were provided as the baseline results together with the database, based on the combination of different photometric preprocessing methods, features, and matching techniques.

LDHF-DB (VIS-NIR, Long Distance)

This database by Maeng et al.[44] was collected by the Korea University. It contains 100 subjects at different distances to the cameras. Each subject was captured at distances of 60, 100, and 150 meters, separately, using both VIS and NIR cameras. There are 1,600 face images in total. This dataset emphasizes the long distance acquisition of heterogeneous face images.

"This dataset emphasizes the long distance acquisition of heterogeneous face images."

UND Database (VIS-IR)

The database contains 82 subjects with multiple IR and VIS face images for each subject. The total number of face images in this database is 2,292. It was used by Choi et al.[4] for IR to VIS face recognition.

NPU3D Database (VIS-3D)

The NPU3D database by Zhang et al.[45] contains Chinese VIS and 3D faces, collected at Northwestern Polytechnical University, China, using the Konica Minolta Vivid 910 3D laser scanner. The acquisition distance is about 1.5 meters. There are 300 individuals captured with 35 different scans (various pose, facial expression, accessory and occlusion) per subject. In total, there are 10,500 3D facial surface scans with the corresponding VIS images.

CASIA NIR-VIS 2.0 Database (VIS-NIR)

It contains 725 subjects of 17,580 face images from multiple recording sessions, in which the first session is identical to the CASIA-HFB database. Each subject has 1–22 VIS and 5–50 NIR face images. Different evaluation protocols were also provided with the database as well by Li et al.[46]

Other Databases

There are also some other databases that are either small, seldom used, or just private, such as, for example, the VIS and IR face database collected by the Pinellas County Sheriff's Office and forensic sketches and VIS databases, introduced by Klare and Jain[2].

Some Thoughts on Future Directions

"As an emerging topic in biometrics, HFR has attracted more and more attention recently."

As an emerging topic in biometrics, HFR has attracted more and more attention recently. However, the study of HFR is still in its early stage, and more efforts are needed to advance the field of research. Here some new thoughts are presented, hopefully to inspire new efforts to address the challenging research on HFR.

Identify Which Methods Can Work on Which HFR Problems

There are different modalities to match within HFR, such as Sketch vs. VIS, NIR vs. VIS, and so on. Different algorithms and approaches have been developed, which are typically for one specific HFR problem or two, but not for all. Even though an algorithm can be tested on different HFR problems experimentally, the recognition accuracies could be very different for different HFR problems. For example, an algorithm can get 95-percent accuracy on VIS vs. sketch, but may only achieve 60-percent accuracy when applied to VIS vs. IR. So an issue is raised: which methods can work on which HFR problems? New investigations can be performed to address this issue, and then one can know which methods are appropriate to solve what kinds of HFR problems. It is especially important for real applications of biometrics systems, not just for academic research. A systematic evaluation of the existing (and future) algorithms on each of the HFR problems could be done towards addressing this issue.

Deal with the Degrees of Heterogeneity in HFR

"By defining and measuring the degrees of heterogeneity, one can know just how difficult it is to solve a specific HFR problem: the more heterogeneous, the more difficult to address typically."

Related to the previous issue, another is to study and define the degrees of heterogeneity in various heterogeneous face matching problems. As presented earlier, there are a variety of HFR problems. However, it has not yet been studied just how heterogeneous it could be between two given modalities of face images. By defining and measuring the degrees of heterogeneity, one can know just how difficult it is to solve a specific HFR problem: the more heterogeneous, the more difficult to address typically.

Further, when a new HFR problem is proposed, one can predict how difficult it will be to address it before developing an algorithm to solve it, based on the measure of degrees of heterogeneity. The challenge is how to define and measure the degree of heterogeneity universally over different matching problems.

And also, the measure of the heterogeneity can help classify the existing (and future) algorithms into different categories based on their capabilities to address the HFR problems at different levels of heterogeneity.

Explore New Learning Methods to Solve HFR Problems

As stated above, the study of HFR is still not mature; new algorithms are expected to be developed to improve the recognition performance. In developing new algorithms, one promising direction is to explore learning-based methods. Since it is difficult (if not impossible) to model how the image appearance is changed from one modality to another, example-based learning approaches are probably the only way to study the differences between two modalities and to build the relations between face images in two modalities.

"In developing new algorithms, one promising direction is to explore learning-based methods."

In exploring learning-based methods, one direction is to study the domain adaptation methods to adapt the data from one modality to another. Recently, we have shown that the adaptive support vector machines (A-SVM) by Yang et al.[47] can be applied for action recognition from VIS to IR by Zhu and Guo[48]. Based on this, we can expect that the A-SVM or other domain adaptation methods could be helpful to address the HFR problems.

Collect Larger Databases with Public Access

As stated earlier, some HFR databases have been assembled; however, few of them are large, compared to the homogeneous (same modality) face recognition databases. By collecting larger databases, one can evaluate the algorithm's performance better towards real applications. Further, there are fewer databases for VIS vs. 3D, VIS vs. IR, makeup vs. no makeup, or containing multiple modalities for the same subjects. New databases can be collected to facilitate the study of various HFR problems, rather than just VIS vs. Sketch or VIS vs. NIR.

"By collecting larger databases, one can evaluate the algorithm's performance better towards real applications."

Other HFR Problems

Some typical and atypical HFR problems were introduced earlier. However, new HFR problems can still be identified along with new sensor development or acquisition environment changes.

Further, some existing face recognition problems may be revisited by considering them as HFR. In this way, new ideas may be inspired to address the well-defined problems from new angles. For example, human aging can cause significant facial appearance changes, as shown in Figure 5. Cross-age face recognition is a well-defined, challenging problem. Various methods have been proposed, such as the generative approaches based on age synthesis by Gong et al.[49], Wu and Chellappa[50], Park et al.[51], Ramanathan and Chellappa[52], and discriminative approaches by Yadav et al.[53], Li et al. [54], Ling et al.[55], and Biswas et al.[56] Because of the space limit, it will not be discussed in detail here, but the cross-age face recognition can be considered as a HFR problem as well.

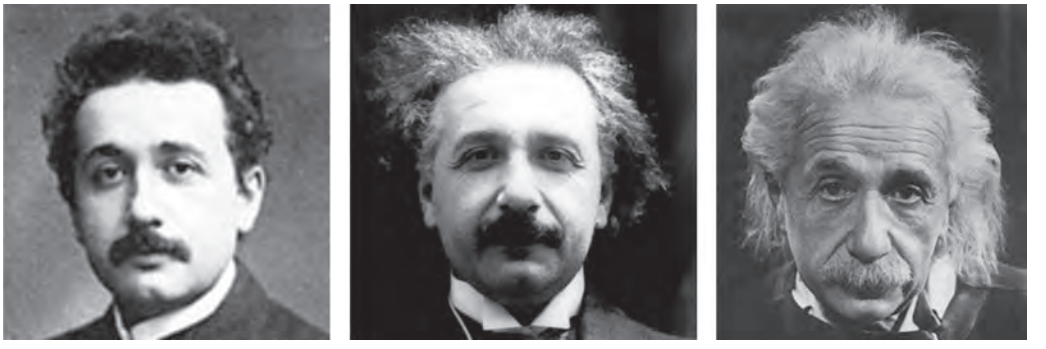


Figure 5: Aging can cause significant facial appearance changes. (Source: Image Search over the Internet, 2014)

Concluding Remarks

An emerging topic in biometrics, called heterogeneous face recognition, has been presented. Several specific HFR problems, both typical and atypical, have been introduced. Some representative approaches to HFR have been described based on a categorization. Various HFR databases have been listed to researchers, and some new thoughts on future exploration of HFR have been introduced as well. Hopefully this article will inspire new research efforts to address the challenging and interesting heterogeneous face recognition problems.

"Hopefully this article will inspire new research efforts to address the challenging and interesting heterogeneous face recognition problems."

Complete References

[1] Li, S., *Encyclopedia of Biometrics* (Springer: 2009).

[2] Klare, B. and A. Jain, "Heterogeneous Face Recognition Using Kernel Prototype Similarities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, No. 6, pp. 1410–1422, 2013.

[3] Li, J., P. Hao, C. Zhang, and M. Dou, "Hallucinating Faces from Thermal Infrared Images," *Proc. Int'l Conf. Image Processing*, pp. 465–468, 2008.

[4] Choi, J., S. Hu, S. Young, and L. Davis. "Thermal to Visible Face Recognition." *Proc. of SPIE*, Vol. 8371, pages 83711L–1, 2012.

[5] Klare, B., Z. Li, and A. Jain, "Matching Forensic Sketches to Mugshot Photos," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 639–646, Mar. 2011.

[6] Goswami, D., C. Chan, D. Windridge, and J. Kittler, "Evaluation of face recognition system in heterogeneous environments (Visible vs NIR)," *2011 IEEE International Conference on Computer Vision Workshops*, pages 2160–2167.

[7] Wang, X. and X. Tang, "Face Photo-Sketch Synthesis and Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 1955–1967, Nov. 2009.

[8] Sharma, A. and D. Jacobs. "Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch," *CVPR*, pages 593–600, 2011.

[9] Lei, Z., S. Liao, A. K. Jain, and S. Z. Li, "Coupled discriminant analysis for heterogeneous face recognition," *IEEE Trans. on Information Forensics and Security*, 7(6), 1707–1716, 2012.

[10] Guo, G-D., L. Wen, and S. Yan, "Face authentication with makeup changes," *IEEE Trans. Circuits and Systems for Video Technology*, DOI:10.1109/TCSVT.2013.2280076

[11] Wen, L. and G-D. Guo, "Dual attributes for face verification robust to facial cosmetics," *Journal of Computer Vision and Image Processing*, NWPJ-201301-82, Vol. 3, No. 1, pages 63–73, 2013.

[12] Ueda, S. and T. Koyama, "Influence of make-up on facial recognition," *Perception*, 39(2):260, 2010.

[13] Dantcheva, A., C. Chen, and A. Ross, "Can facial cosmetics affect the matching accuracy of face recognition systems?" *IEEE Conf. on Biometrics: Theory, Applications and Systems*, Washington DC, USA, 2012.

[14] Chen, C., A. Dantcheva, and A. Ross, "Automatic facial makeup detection with application in face recognition," *Proc. of International Conference on Biometrics* (ICB), Madrid, Spain, June 2013.

[15] Tang, X. and X. Wang, "Face Sketch Recognition," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 50–57, Jan. 2004.

[16] Liu, Q., X. Tang, H. Jin, H. Lu, and S. Ma, "A Nonlinear Approach for Face Sketch Synthesis and Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1005–1010, 2005.

[17] Gao, X., J. Zhong, J. Li, and C. Tian, "Face Sketch Synthesis Algorithm Based on E-HMM and Selective Ensemble," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 4, pp. 487–496, Apr. 2008.

[18] Zhang, W., X. Wang, and X. Tang, "Lighting and Pose Robust Face Sketch Synthesis," *Proc. European Conf. Computer Vision*, 2010.

[19] Turk, M. and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[20] Chen, J., D. Yi, J. Yang, G. Zhao, S. Z. Li, and M. Pietikainen, "Learning mappings for face synthesis from near infrared to visual light images," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 156–163, 2009.

[21] Wang, R., J. Yang, D. Yi, and S. Z. Li, "An analysis-by-synthesis method for heterogeneous face biometrics," *Advances in Biometrics* (Berlin: Springer 2009), pp. 319–326.

[22] Liu, M., W. Xie, X. Chen, Y. Ma, Y. Guo, J. Meng, and Q. Qin, "Heterogeneous face biometrics based on Gaussian weights and invariant features synthesis," *IEEE 2nd International Conference on Computing, Control and Industrial Engineering (CCIE)*, Vol. 2, pp. 374–377, 2011.

[23] Hertzmann, A., C. Jacobs, N. Oliver, B. Curless, and D. Salesin, "Image analogies," *SIGGRAPH*, 2001.

[24] Hotelling, H., "Relations between two sets of variates." *Biometrika* 28, 321–377 (1936).

[25] Wold, H. "Partial least squares," in *Encyclopedia of Statistical Sciences,* edited by S. Kotz and N. Johnson, volume 6, pages 581–591. Wiley, New York, 1985.

[26] Yang, W., D. Yi, Z. Lei, J. Sang, and S. Li, "2D-3D face matching using CCA," *IEEE International Conference on Automatic Face Gesture Recognition*, pages 1–6, 2008.

[27] Yi, D., R. Liu, R. Chu, Z. Lei, and S. Z. Li, "Face matching between near infrared and visible light images," *IAPR International Conf. on Biometric*, pages 523–530, 2007.

[28] Lin, D. and X. Tang, "Inter-Modality Face Recognition," *Proc. European Conf. Computer Vision*, pages 13–26, 2006.

[29] Balcan, M.-F., A. Blum, and S. Vempala, "Kernels as Features: On Kernels, Margins, and Low-Dimensional Mappings," *Machine Learning*, vol. 65, pp. 79–94, 2006.

[30] Ho, T. K., "The Random Subspace Method for Constructing Decision Forests," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[31] Breiman, L., "Bagging Predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[32] Breiman, L., "Random forests," *Machine Learning* 45, No. 1 (2001): 5–32.

[33] Wang, X. and X. Tang, "Random Sampling for Subspace Face Recognition," *Int'l J. Computer Vision*, vol. 70, no. 1, pp. 91–104, 2006.

[34] Klare, B. and A. Jain, "Heterogeneous face recognition: Matching NIR to visible light images," *Int'l Conf. on Pattern Recognition*, 2010, pp. 1513–1516.

[35] Lampert, C., H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 951–958, 2009.

[36] Farhadi, A., I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1778–1785, 2009.

[37] Farhadi, A., I. Endres, and D. Hoiem, "Attribute-centric recognition for cross-category generalization," *IEEE Conf. on CVPR*, pages 2352–2359, 2010.

[38] Kumar, N., A. Berg, P. Belhumeur, and S. Nayar, "Attribute and simile classifiers for face verification," *IEEE International Conf. on Computer Vision*, pages 365–372, 2009.

[39] Rupnik, J. and J. Shawe-Taylor, "Multi-view canonical correlation analysis," *SiKDD*, 2010.

[40] Kan, M., S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," *European Conf. on Computer Vision*, pp. 808–821 (2012).

[41] Sharma, A., A. Kumar, H. Daume III, and D. W. Jacobs, "Generalized multiview analysis: A discriminative latent space," IEEE Conference on Computer Vision and Pattern Recognition (2012).

[42] Zhang, W., X. Wang, and X. Tang, "Coupled Information-Theoretic Encoding for Face Photo-Sketch Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[43] Li, S. Z., Z. Lei, and M. Ao, "The HFB face database for heterogeneous face biometrics research." *IEEE Computer Vision and Pattern Recognition Workshops*, pp. 1–8, 2009.

[44] Maeng, H., S. Liao, S.-W. Lee, and A. K. Jain. "Nighttime face recognition at long distance: cross-distance and cross-spectral matching," *Asian Conf. on Computer Vision*, pp. 708–721. 2012.

[45] Zhang, Y., Z. Guo, Z. Lin, H. Zhang, and C. Zhang, "The NPU Multi-case Chinese 3D Face Database and Information Processing," *Chinese Journal of Electronics*, vol. 21, no. 2 (2012): 283–286.

[46] Li, S. Z., D. Yi, Z. Lei, and S. Liao, "The CASIA NIR-VIS 2.0 face database," *Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 348–353, 2013.

[47] Yang, J., R. Yan, and A. Hauptmann, "Cross-Domain Video Concept Detection using Adaptive SVMs," *Proc. MM*, 2007.

[48] Zhu, Y. and G-D. Guo, "A study on visible to infrared action recognition," *IEEE Signal Processing Letters*, Vol. 20, No. 9, pages 897–900, 2013.

[49] Gong, D., Z. Li, D. Lin, J. Liu, and X. Tang, X. "Hidden Factor Analysis for Age Invariant Face Recognition," *ICCV* 2013.

[50] Wu, T. and R. Chellappa, "Age invariant face verification with relative craniofacial growth model," *Computer Vision–ECCV 2012* (pp. 58–71).

[51] Park, U., Y. Tong, and A. K. Jain, "Age-invariant face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32(5), 947–954.

[52] Ramanathan, N. and R. Chellappa, "Face verification across age progression," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3349–3361, 2006.

[53] Yadav, D., M. Vatsa, R. Singh, and M. Tistarelli, "Bacteria Foraging Fusion for Face Recognition across Age Progression," *Computer Vision and Pattern Recognition Workshops (CVPRW)*, (2013) pp. 173–179.

[54] Li, Z., U. Park, and A. K. Jain, "A discriminative model for age invariant face recognition," *IEEE Transactions on Information Forensics and Security*, (2011) 6(3), 1028–1037.

[55] Ling, H., S. Soatto, N. Ramanathan, and D. W. Jacobs, "Face verification across age progression using discriminative methods," *IEEE Transactions on Information Forensics and Security*, (2010) 5(1), 82–91.

[56] Biswas, S., G. Aggarwal, and R. Chellappa, "A non-generative approach for face recognition across aging," in IEEE Second Int'l Conf. on Biometrics: Theory, Application and Systems, 2008.

Author Biography

Guodong Guo received his BE degree in Automation from Tsinghua University, Beijing, China, in 1991, a PhD in Pattern Recognition and Intelligent Control from the Chinese Academy of Sciences, in 1998, and a PhD in computer science from the University of Wisconsin-Madison, in 2006. He is currently an assistant professor in the Lane Department of Computer Science and Electrical Engineering at West Virginia University. In the past, he has visited and worked in several places, including INRIA, Sophia Antipolis, France, Ritsumeikan University, Japan, Microsoft Research, China, and North Carolina Central University. He won the North Carolina State Award for Excellence in Innovation in 2008, and Outstanding Researcher (2013–2014) and New Researcher of the Year (2010–2011) at CEMR, WVU. He was selected as the "People's Hero of the Week" by BSJB under MMTC on July 29, 2013. His research areas include computer vision, machine learning, and multimedia. He is the author of *Face, Expression, and Iris Recognition Using Learning-based Approaches* (2008), co-editor of *Support Vector Machines Applications* (2014), and has published over 60 technical papers in face, iris, expression, and gender recognition, age estimation, and multimedia information retrieval. He can be contacted at Guodong.Guo@mail.wvu.edu

ON DESIGNING SWIR TO VISIBLE FACE MATCHING ALGORITHMS

Contributors

Cameron Whitelam
Multispectral Imagery Lab,
West Virginia University

Thirimachos Bourlai
Multispectral Imagery Lab,
West Virginia University

Recent advances in facial recognition have trended towards cross-spectrally matching visible gallery face images to probe face images captured under different wavelengths of the electromagnetic spectrum. In this article, we study the problem of matching visible images to images taken in the short-wavelength infrared (SWIR) spectrum, more specifically, the 1550-nm band. There are many benefits to using the SWIR spectrum for face recognition, including covert capturing in nighttime environments as well as imaging through certain environmental conditions such as fog and smoke. However, due to the fact that the moisture in the skin tends to absorb the 1550-nm wavelength, all subjects appear to have dark or black skin tone. Because of the stark contrast between 1550-nm and visible face images, standard face recognition protocols fail to accurately match images captured using sensors operating on different bands. While preliminary work in this area resulted in fairly good performance results, it was determined that a more sophisticated approach could be developed to further improve our original face recognition algorithm in terms of (i) accuracy, (ii) speed, and (iii) adaptability, that is, the proposed algorithm should achieve good results on a wider variety of testing scenarios (diverse face datasets).

More specifically, we study the advantages and limitations of our new proposed cross-spectral matching (visible to SWIR) technique when using an extended set of challenging FR scenarios. The proposed face matching algorithm is a significant improvement when compared to the original algorithm where fused texture-based scores of a large number of photometric normalization combinations between SWIR and visible images were used to achieve satisfactory recognition performance results. Our contributions are threefold. Firstly, multiple databases are considered, which represent different difficult environments, that is, multiband face images were acquired under different lighting conditions and behind different obscurants (multiple levels of tinted glass). Secondly, we demonstrate that the use of a random selection of intensity-based normalization techniques is not necessary. This is because a random combination of such techniques does not have a significant amount of discriminatory information to accurately match one subject's face to another, yielding undesirably low face-matching scores. Thirdly, we demonstrate that a smart selection of a subset of normalization techniques not only results in obtaining more accurate face recognition performance scores, but also drastically decreases the long processing time required to produce even a single face-to-face image match score. Our design also incorporates the usage of parallel processing to further boost the time needed to perform cross-spectral matching. Finally, our experiments indicate that the level of improvement in recognition accuracy is scenario dependent.

"…fused texture-based scores of a large number of photometric normalization combinations between SWIR and visible images were used to achieve satisfactory recognition performance…"

Introduction

The past decade's research efforts in the area of facial recognition resulted in a significant improvement in terms of recognition performance. This can be inferred from the results of the 2010 Multiple-Biometrics Evaluation (MBE) study organized by NIST.[1] In 1993, at a false match rate (FMR) of 0.001 percent, the best performing face matcher had a false non-match rate (FNMR) of 79 percent. According to NIST's MBE, the FNMR has significantly dropped to an impressive 0.003 percent at a false accept rate (FAR) of 0.001 percent. Typically, face recognition algorithms perform well in the visible band of the electromagnetic spectrum (380–750 nm). However, the problem of matching facial images remains a challenge when dealing with difficult and diverse scenarios, including the usage of different capturing sensors (for example 2D, 3D, visible, or IR), large datasets, or having to deal with facial obscurants, as well as pose, illumination, and expression variations. There are many inherent problems that come along with visible face recognition. First and foremost, the effect of illumination variation on visible band images is among the most insidious problems that face matching algorithms need to efficiently deal with. Because of this problem, recent FR trends are leading away from the visible spectrum and heading to different infrared bands: near infrared (NIR: 750–1100 nm)[2][3], short-waved infrared (SWIR: 900-1900 nm)[4], mid-waved infrared (MWIR: 3–5 μm)[5][6], and long-waved infrared (LWIR: 7–14 μm).[7] The main disadvantage of IR-based biometric identification at the present time is the high price of the majority of high end sensors. However, the cost of infrared security cameras has dropped considerably and is now comparable to high end digital single-lens reflex (DSLR) cameras (visible band). For example, in the thermal band, FLIR is now offering LWIR cameras starting at less than USD 3,000, making them more affordable and, thus, researchers can utilize them in several innovative ways. This scenario would have been inconceivable just a few years ago. Affordable IR sensors provide the opportunity to create more challenging databases (larger scale, different sensors operating on the same or different IR bands), allowing also for the development and testing of heterogeneous face matching algorithms where visible images are matched to NIR, SWIR, MWIR, and LWIR face images.

"…recent FR trends are leading away from the visible spectrum and heading to different infrared bands…"

The focus of this work is matching visible to SWIR band face images. There are many benefits when using SWIR camera sensors for the purpose of designing and developing face recognition algorithms. First, the SWIR spectrum allows for covert capture of face images in nighttime environments considering that the illumination source is invisible to the human eye (due to the wavelength being well beyond the visible spectrum). Another advantage of the SWIR band is the capability to see through different types and levels of tinted glass as well as sunglasses.[11] SWIR has a longer wavelength range than NIR and is more tolerant to low levels of obscurants like fog and smoke. Finally, different facial features can be extracted in the SWIR band that can be combined with those extracted in the visible band to create a more accurate and complete representation of a subject's face.[4] Our previous studies determined that this

"…SWIR spectrum allows for covert capture of face images in nighttime environments considering that the illumination source is invisible to the human eye…"

capability resulted in an increase of rank-one identification rates under variable face matching scenarios.[16]

While previous FR studies have mainly concentrated on the visible and NIR bands, FR in the SWIR band, more specifically the 1550 nm wavelength, has received limited attention. Prior work focused on dealing with face datasets assembled under controlled and challenging conditions.[4] However, in uncontrolled scenarios (long range recognition and imaging behind glass operational conditions), there is a need for efficient intelligence and surveillance reconnaissance (ISR) interoperability, that is, operational teams (for example, armed forces) are required to effectively manage, access, and use ISR to improve command and control, and enhance information sharing and situational understanding to improve the effectiveness of operations while minimizing collateral damage in a complex environment. One particular issue that we address in this article is the ability to capture a subject's face behind glass (that can be used in commercial buildings, homes, or vehicles), especially when the glass is tinted. Being able to image a subject behind different types of tinted glass and accurately match them with images from a database of visible images (such as a watch list, do-not-fly list, and so on) is an important step in improving human identification in operational environments.

"Being able to image a subject's face behind different types of tinted glass and accurately match them with images from a database of visible images is an important step in improving human identification in operational environments."

Goals and Contributions

In this article, we propose a new cross-spectral face matching algorithm. It significantly enhances the capability of the original approach proposed by Kalka et al.[4] to match SWIR to visible face images in variable challenging scenarios, including scenarios where face images were captured behind different types of tinted glass. Firstly, in order to evaluate the efficiency of our proposed approach, a database of subjects was assembled behind multiple types of tinted glass and under different lighting conditions (that is, ambient lighting or the usage of SWIR active illumination). Secondly, we determined that our wavelength- and scenario-dependent eye detection algorithm performs very well on all datasets that it was tested on. In addition, experiments using our proposed face matching algorithm show that the use of randomly selected photometric normalization techniques (as proposed in Kalka et al.[4]) is not necessary to improve FR performance. This is due to the fact that certain normalization techniques do not yield enough discriminatory information in the face which, in turn, yields low face match (similarity) scores. Specifically, we demonstrate that the use of only a small subset of more than 45 normalization techniques (and their combinations) available and tested was necessary to increase the overall performance of our face matcher, while drastically reducing the computational time required to perform a single match (that is, using a small subset vs. all possible combinations). Our proposed design also includes the use of parallel processing, which further reduces the time needed to perform a single match. Finally, our experiments show that the level of improvement achieved when using our proposed face matching approach in variable challenging face datasets is scenario dependent.

"…certain normalization techniques do not yield enough discriminatory information in the face…"

The rest of the article is organized as follows. The following section, "Background Information," discusses some background work done in the field of heterogeneous face recognition. "Data Collection" describes our data collection process, while the section "Methodological Approach" offers insights into our methodology. Finally, the section "Experimental Setup" demonstrates experimental results, while the section "Conclusions and Future Work" draws conclusions and discusses our future work.

Background Information

The field of heterogeneous face matching can be broken down into four different categories: NIR-visible, SWIR-visible, MWIR-visible and LWIR-visible matching. Because each band of the IR spectrum reveals different facial characteristics, different face recognition algorithms (including face/eye detection, feature extraction, and matching) must be designed, developed, and used when working in any specific face matching category described above. In the NIR spectrum, Klare and Jain[2] learned discriminative projections using common feature-based representations (LBP and HOG features) as well as linear discriminant analysis (LDA) on both NIR and visible images. In the proposed approach, the authors matched NIR to visible images directly using random subspace projections as well as using sparse representation classification. Zhu et al.[3] proposed the transductive heterogeneous face matching (THFM) method that adapts the NIR-visible matching, learned from a training database, to target images. With the use of their version of a Log-DoG (difference of Gaussian) filtering, along with local encoding and feature normalization, they were able to alleviate the heterogeneous difference between the two spectral bands. The transduction-based approach simultaneously reduces the domain difference and learns the discriminative model for target subjects. This resulted in fairly accurate NIR-visible matching scores.

"…different face recognition algorithms must be designed developed and used when working in any specific face matching category…"

In the category of SWIR-visible face matching, Mendez et al.[7] use nonlinear dimensionality reduction approaches. Global nonlinear techniques, such as Kernel-PCA and Kernel-LDA, as well as local nonlinear techniques, such as local linear embedding and locality preserving projections were compared to their linear counterparts, PCA and LDA. Experiments showed that the use of local nonlinear dimensionality reduction techniques resulted in higher FR matching rates in the SWIR band on two controlled databases. Kalka et al.[4] used a photometric fusion technique that incorporated six different illumination normalization schemes. These techniques were combined in both the SWIR and visible bands to create 36 photometric combinations. A simple summation fusion scheme was then used to determine the final match score. The approach was tested on a set of datasets representing difficult challenging environments including SWIR images taken in an operational setting (that is, in the wild). Experimental results showed that this approach outperformed other texture-based approaches and was dependent on the scenario tested.

In the field of passive IR face matching, Bourlai et al.[5][17][18][19][20][21] use variable schemes, including a texture-based fusion scheme to match MWIR or LWIR

probe to visible gallery face images. Using different databases ranging from 50 to more than 100 subjects, different approaches, including texture-based ones such as local binary and ternary patterns (LBP/LTP), pyramid histogram of gradient (PHOG), and scale invariant feature transform (SIFT) were compared and fused after applying different photometric normalization techniques. Osia and Bourlai[6][17] also match MWIR probe images to visible gallery images. Face features, such as veins, scars, and wrinkles, are first extracted using multiple techniques including a standard fingerprint extraction method, the scale-invariant feature transform (SIFT), and the speeded up robust feature (SURF) method. A fingerprint matcher is then used to match the extracted features (from either the whole face or subregions of the face) from same band face images (visible-visible, MWIR-MWIR and so on).

"…we are discussing the problem of matching SWIR against visible face images captured under variable conditions."

In this work, we are discussing the problem of matching SWIR against visible face images captured under variable conditions. What follows is a description of the face datasets used for the purpose of our study.

Data Collection

To the best of our knowledge, there are no publicly available face databases in the research community that are composed of visible and SWIR face images captured behind tinted glass. In this work, we discuss the database collected at WVU for the purpose of our study and the data collection protocol designed and executed. To simulate an operational environment, where face images need to be captured through tinted glass, a three-sided booth, with a 1-ft. × 1-ft. window slot, was built for subjects to sit behind and have their faces collected by both visible and SWIR cameras. The window slot was set so different types of tinted glass could be easily switched. Both industrial and automotive tinted glass was used for the purpose of this study. The three types of glass that were used can be described as follows:

- Industrial clear glass panel (clear with 0-percent tint)
- Industrial clear glass panel with tinted automotive film applied (80-percent tinted film)
- Industrial tinted glass panel (Solarcool-2 Graylite)

"Each panel allows for the simulation of different operational scenarios…"

The use of these panels was chosen in order to test the performance of heterogeneous face recognition across a wide spectrum of varying levels of tint. Each panel allows for the simulation of different operational scenarios such as imaging through normal glass (0-percent tint), through a vehicle's tinted side window (80-percent tinted film) or through a secure building with highly tinted windows (Solarcool-2 Graylite).

Two different light sources (scenarios) were used to illuminate the subjects' faces while sitting behind the booth and glass panels. In the first scenario, an interior light source (inside the booth with the subject present) was used to illuminate the subject's face. Two 250 W tungsten lightbulbs were positioned in the booth to optimize the illumination on the subject's face without hotspots

and produced ~3.87 kilolux of light. In the second scenario, an external active SWIR illumination source was used. A 1550-nm laser source with a 500 mW light diffuser was positioned outside of the booth and illuminated the glass externally. The SWIR illumination source was set up at an angle from the glass panel in order to minimize the reflections back into the camera. An image of each subject's face behind the glass panel was captured with the two different illumination sources using the following cameras:

"An image of each subject's face behind the glass panel was captured with the two different illumination sources…"

- *Visible Camera:* A Canon EOS 5D Mark II camera was used to capture the visible images. This digital SLR camera has a 21.1-megapixel full-frame CMOS sensor with a DIGIC 4 image processor and a vast ISO range of 100–6400. It also has an auto lighting optimizer and peripheral illumination correction that enhances its capabilities. The Mark II was used to collect RGB images of the subjects at the ground truth level (no glass) as well as when using each glass panel with either of the two illumination sources tested.
- *SWIR Camera:* A Goodrich SU640 camera was used to capture SWIR face images. The SU640 is an indium gallium arsenide (InGaAs) video camera featuring high sensitivity and wide dynamic range. This model has a 640 × 512 FPA with 25 m pixel pitch and >99 percent pixel operability. The spectral sensitivity of the SU640 ranges uniformly from 700–1700 nm wavelength. The response falls rapidly at wavelengths lower than 700 nm and greater than 1700 nm.

A total of 140 subjects participated in our study. When a subject arrived for collection, ground truth images were first taken with no glass panels, when using either the visible or the SWIR camera sensors. Then, the following conditions were considered when using the aforementioned glass panels, the SWIR camera, and different illuminators. In addition, a 100-nm band pass filter, centered at 1550 nm, was placed in front of the SWIR camera to ensure that only SWIR light in that particular waveband was entering the camera.

- 0 percent tint
  - Internal visible illumination
  - External active SWIR illumination
- 80 percent tinted film
  - Internal visible illumination
  - External active SWIR illumination
- Solarcool-2 Graylite
  - Internal visible illumination
  - External active SWIR illumination

Overall, we assembled a total of seven databases under variable scenarios (including the ground truth database where no glass was used between a subject's face and the camera). The final set of databases assembled consisted of the data of 140 subjects, that is, 980 SWIR face images (140 subjects × 7 scenarios) and

"Overall, we assembled a total of seven databases under variable scenarios…"

140 visible (ground truth) face images, totaling 1,020 face images. These images were then used to conduct the heterogeneous face recognition experiments that are described later in the section "Experimental Setup." In the next section, we describe our methodological approach used to deal with the variable challenges of the cross-spectral face matching scenarios investigated in this study.

Methodological Approach

In this section, our methodological approach to perform heterogeneous face recognition is discussed. First, we outline an overview of our automatic face/eye detection algorithm. This allows for each subject's face to be geometrically normalized to the same plane and used for face recognition. Then, we describe our photometric normalization fusion face recognition algorithm. Our method uses a texture-based matching algorithm, local binary patterns (LBP) and local ternary patterns (LTP), as well as our new proposed cross-photometric normalization fusion scheme to accurately match SWIR (probe dataset) to visible face images (gallery dataset). An empirical study is conducted to help reduce the number of photometric normalization techniques used, which, in turn, helps reduce the time required to obtain a face match score. What we discuss in the following sections are the steps used in our proposed heterogeneous face recognition approach.

"An empirical study is conducted to help reduce the number of photometric normalization techniques used…"

Automatic Face and Eye Detection

In general, commercial and academic facial recognition algorithms require that the face images of each individual be standardized (in terms of orientation, interocular distance, masking, and so forth). Typically, feature points of sub-facial regions, more specifically the locations of human eye centers, are used to rotate and translate a face to a standard representation. While this operation can be manually performed by an operator on a limited size dataset, when having to deal with larger databases (from thousands to millions of subjects), manually obtaining the eye locations of each subject, and for each sample per subject available, is not practical. Therefore, an accurate and robust eye detection method needs to be employed since it is expected to have positive impact on FR performance on any typical FR system dependent on eye locations. Our face and eye detection method comprises five main processes: preprocessing, automatic face detection, eye region localization, summation range filtering, and geometric normalization. This leads to an image that is suitable for a face recognition system.

"…the location of the human eye centers are used to rotate and translate a face to a standard representation."

Preprocessing

SWIR images tend to have low contrast in the facial region, especially in the 1550-nm band. Instead of the human skin reflecting those wavelengths back into the camera, the moisture from the skin tends to absorb higher SWIR wavelengths, causing the skin to appear very dark (even for very light-skinned subjects). In order to compensate for this, photometric normalization techniques bring out unique features that are beneficial for face and eye detection algorithms applied to wavelength-specific face datasets.

Because we use a *template matching* scheme to detect the face and eye regions, average templates are needed. Therefore, for the purpose of this study, seven subjects are randomly selected from each capturing scenario and their faces are geometrically normalized, cropped, and averaged together to create an average face template. Then, the eye regions from this template are cropped and used as *average eye templates*. These average templates are, finally, saved and used on all images in the database.

Automatic Face Detection

Because of the unique qualities that SWIR images have, typical face detection algorithms could not be used. Therefore, a template-based face detection algorithm was developed to spatially locate the face region. For each pixel in the query face image, the 2D normalized cross-correlation is computed between the region of that pixel and the average face template. Mathematically, the 2D normalized cross-correlation can be described as:

"Because of the unique qualities that SWIR images have, typical face detection algorithms could not be used."

$$\delta(u,v) = \frac{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]\left[t(x-u, y-v) - \bar{t}\right]}{\left\{\sum_{x,y}\left[f(x,y) - \bar{f}_{u,v}\right]^2 \sum_{x,y}\left[t(x-u, y-v) - \bar{t}\right]^2\right\}^{1/2}} \quad (1)$$

where f is the image, $\bar{t}$ is the mean of the template, and $\bar{f}(u,v)$ is the mean of $f(x,y)$ in the region under the template. Then, the convolution of the image and the average template yields a correlation map. The highest location within the correlation map (the peak) is the location of the face. However, different average templates yield different eye detection results. Because of this issue, multiple average templates (in our case five) are created and used to increase the chance of finding the correct location. The final location of the face can be described with the following formula:

$$\hat{\delta}(u,v) = argmax\,(\delta_x(u,v)) \quad (2)$$

where $\delta_x(u,v)$ is the location of the highest correlation coefficient obtained from average template x (in our case $x = 1,\ldots,5$). Then, $\hat{\delta}(u,v)$ corresponds to the upper left point of the face region, and finally, the face can be cropped to the size of the average templates used. By only using the face area determined by this approach, our subsequent eye detection method only has to search the face area (a much smaller region), instead of the entire image as a whole.

"By only using the face area determined by this approach, our subsequent eye detection method only has to search the face area, instead of the entire image as a whole."

Eye Region Localization

Since the location of the face is now known, the location of the eye regions can be easily determined. In order to further reduce the search space, the face is split into four equal regions (top left, top right, bottom left, and bottom right). Assuming that the face region is found correctly by using the method described above, the right and left eyes should be located in the top right and top left regions respectively. Therefore, to obtain the left and right eye regions, the average eye templates are convolved with their respective quadrants using Equation 1. As stated above, different average eye templates yield different results. Therefore, the process is repeated

multiple times using unique templates to increase the chance of obtaining the correct region. Then, Equation 2 can be used to find the final location of the eye regions.

Summation Range Filter

Although the region of the eye can be easily found, the center of the eye cannot always be determined to be the center of the found region. Therefore, an accurate way of determining the correct center of the eye must be employed. Knowing that the pupil is typically much darker than an iris, and that, in certain conditions, light reflections from an illumination source are available within the eye (typically in the pupil), summation range filters can be used to more accurately determine the center of the eye. The summation range map *S (x,y)* can be described as follows:

"…summation range filters can be used to more accurately determine the center of the eye."

$$S(x,y) = \sum_{x=-1}^{1} \sum_{y=-1}^{1} R(x,y) \tag{3}$$

where

$$R(x,y) = argmax\left(I(x-1:x+1, y-1:y+1)\right) - argmin\left(I(x-1:x+1, y-1:y+1)\right) \tag{4}$$

and where *I(x,y)* is the original cropped eye region. Then, the final eye center is determined to be

$$P(x,y) = argmax\left(S(x,y)\right) \tag{5}$$

This process is repeated for both the right and left eye regions to determine the final locations for the right and left eye respectively.

Geometric Normalization

In order to assist facial recognition systems, the left and right eye locations are used to geometrically normalize the image. By setting a standard interocular distance, the eye locations can be centered and aligned onto a single horizontal plane and resized so all images are similar to each other. This ensures that, if the eyes are found correctly, the left and right eyes are guaranteed to be in the same position every time, an assumption that is made by most facial recognition algorithms. Therefore, all face images are geometrically normalized based on the found locations to have an interocular distance of 60 pixels with a resolution of 130 × 130 pixels. The geometrically normalized images can then be used in our facial recognition system.

"…if the eyes are found correctly, the left and right eyes are guaranteed to be in the same position every time…"

Face Matching Algorithm

In this work, both commercial and research software was employed to perform the face recognition experiments: (1) Commercial software, such as Identity Tools G8 provided by L1 Systems and (2) standard texture-based feature methods. Two different texture-based schemes were used to test our algorithms, namely local binary patterns (LBP) and local ternary patterns (LTP).

In the LBP operator, patterns in an image are computed by thresholding 3×3 neighborhoods based on the value of the center pixel. Then, the resulting binary pattern is converted to a decimal value. The local neighborhood is defined as a set of sampling points evenly spaced in a circle. The LBP operator used in our experiments is described as $LBP^{u^2}_{P,R}$, where P refers to the number of sampling points placed on a circle with radius R. The symbol u^2 represents the uniform pattern, which accounts for the most frequently occurring pattern in our experiments. The pattern is important because it is capable of characterizing local regions that contain edges and corners. The binary pattern for pixels, lying in a circle f_p, $p = 0,1,\ldots,P-1$ with the center pixel f_c, is mathematically computed as follows:

"The pattern is important because it is capable of characterizing local regions that contain edges and corners."

$$S(f_p - f_c) = \begin{cases} 1 \; if f_p - f_c \geq 0; \\ 0 \; if f_p - f_c < 0; \end{cases} \tag{6}$$

Following this a binomial weight 2^p is assigned to each sign $S(f_p - f_c)$ to compute the LBP code,

$$LBP_{P,R} = \sum_{P=0}^{P-1} S(f_p - f_c) 2^p \tag{7}$$

LBP is invariant to monotonic gray-level transformations. However, one disadvantage is that LBP tends to be sensitive to noise in homogeneous image regions since the binary code is computed by thresholding the center of the pixel region.

Consequently, LTP has been introduced to overcome such a limitation, where the quantization is performed as follows:

$$S(f_p - f_c) = \begin{cases} 1 \; if f_p - f_c \geq t \\ 0 \; if \; |f_p - f_c| \leq t \\ -1 \; if f_p - f_c \leq -t \end{cases} \tag{8}$$

The output of this operator is a 3-valued pattern, as opposed to a binary pattern. Furthermore, the threshold t, can be adjusted to produce different patterns. The user-specific threshold also makes the LTP code more resistant to noise.

An Empirical Study on Photometric Normalization

The problem of cross-spectral FR, matching visible to SWIR face images, is very challenging because of the interaction between the electromagnetic waves (visible and SWIR) and the material (in our case, human skin). This results in different reflectance, transmission, and scattering properties. Because of this, contrast, texture, and so on are different when dealing with visible and SWIR images, respectively. Photometric normalization algorithms traditionally have been employed in order to compensate for these changes in illumination, such as shadows and varying light conditions. In this work, we employ six different

" The problem of cross-spectral FR is very challenging because of the interaction between the electromagnetic waves and the material."

photometric normalization techniques in order to facilitate cross-spectral matching. More specifically, we employ the following techniques: contrast-limited adaptive histogram equalization (CLAHE), tangent-based single-scale retinex (TBSSR), log-based single-scale retinex (LBSSR), TBSSR followed by CLAHE (C-TBSSR), LBSSR followed by CLAHE (C-LBSSR), and the Tan and Triggs[9] normalization (TT). Sample images of these photometric normalizations can be seen in Figure 1.

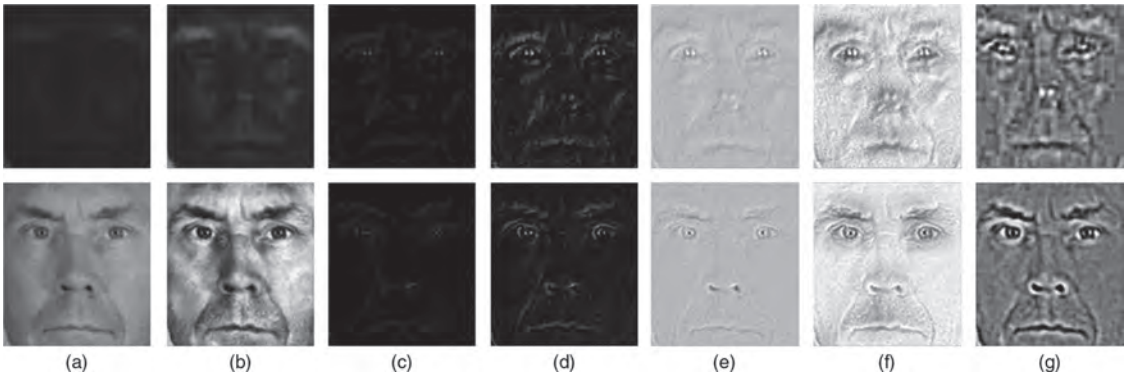


Figure 1: Sample with different photometric normalizations: a) Original data b) CLAHE c) LBSSR d) C-LBSST e) TBSSR f) C-TBSSR and g) TT
(Source: West Virginia University, 2014)

- CLAHE: This technique operates on small local regions (8×8 for our experiments) in the image and applies histogram equalization on each individual region (in contrast to the entire image in regular histogram equalization). In order to increase contrast while decreasing the amount of noise, CLAHE redistributes each histogram so that the height of each bin falls below a predetermined threshold (0.1 in our reported experiments).
- TBSSR: This decomposes the image into two components, illumination $L(x,y)$ (the amount of light falling on the targeted object) and reflectance $R(x,y)$ (the amount of light reflecting off the targeted object). The illumination component is estimated as a low-pass version of the original image, while the reflectance component is obtained by dividing the original image from other illumination images. Therefore, to calculate the TBSSR,

$$R(x,y) = atan\left(\frac{I(x,y)}{L(x,y)}\right) \tag{9}$$

"A common problem with TBSSR is that images tend to become oversaturated or 'washed-out'."

- C-TBSSR: A common problem with TBSSR is that images tend to become oversaturated or "washed out." This can have negative effects on eye detection algorithms. Furthermore, "halo" artifacts may be introduced depending on the scene and scale of value chosen for the Gaussian smoothing function. To diminish the cost of processing speed, we applied the CLAHE approach listed above to TBSSR face images to help compensate for the aforementioned approaches and to increase the contrast of the image.

- LBSSR: By using different nonlinear transformations on the TBSSR, different image representations can be obtained. Therefore, the *atan* in Equation 1 is replaced with a logarithmic transformation, resulting in the following formula:

$$R(x,y) = log_{10}\left(\frac{I(x,y)}{L(x,y)}\right) \quad (10)$$

- C-LBSSR: As described above, the LBSSR can cause over saturation and haloing effects. Therefore, the CLAHE approach was also applied to the LBSSR image to correct the contrast issues mentioned above.
- TT: This photometric normalization[9] incorporates a series of algorithmic steps that allow for the reduction of illumination variations, local shadowing, and highlights, while still preserving the essential elements of visual appearance. These steps include gamma correction (raising each pixel value to a certain value, in this case 2), difference of Gaussian filtering (subtraction of an original image from the blurred version of the same image), and contrast equalization (suppressing larger intensities while maintaining lower intensities).

"…we wanted to determine which combination of photometric normalization algorithms produces the best match scores…"

In this empirical study, we wanted to determine which combination of photometric normalization algorithms produces the best match scores between visible images and 1550-nm SWIR face images. In order to do this, a heterogeneous cross-spectral approach was used. First, all gallery and probe images are photometrically normalized using the techniques described above. Then, each normalization-per-probe image is matched with each normalization-per-gallery image. With the original face image and the six photometric normalizations used ($n = 7$), 49 different photometric combinations are created per match (7 probe representations × 7 gallery representations). This resulted in 49 different match scores for a single probe-gallery match. An overview of this process can be seen in Figure 2.

"The ROC is used to examine the relationship between the true positive and the face positive rate."

Once all probe images are matched to all gallery images, each photometric combination is broken down into their respective genuine (true positive) and imposter (true negative) scores. Then, the *receiver operator curve* (ROC) is computed for all 49 photometric normalizations. The ROC is used to examine the relationship between the true positive and the false positive rate. To quantify which ROC performs better than another, a measure must be taken. In this case, the *area under the curve* (AUC) is used as a measurement to determine which photometric combination performs better than the others. Higher AUCs show combinations that have a wider gap between true positives and false positives, which, in turn, results in higher performance.

After computing the AUCs for all 49 combinations, we can determine which photometric combinations result in higher performance. Sample ROCs and their respective AUCs can be seen in Figure 3.

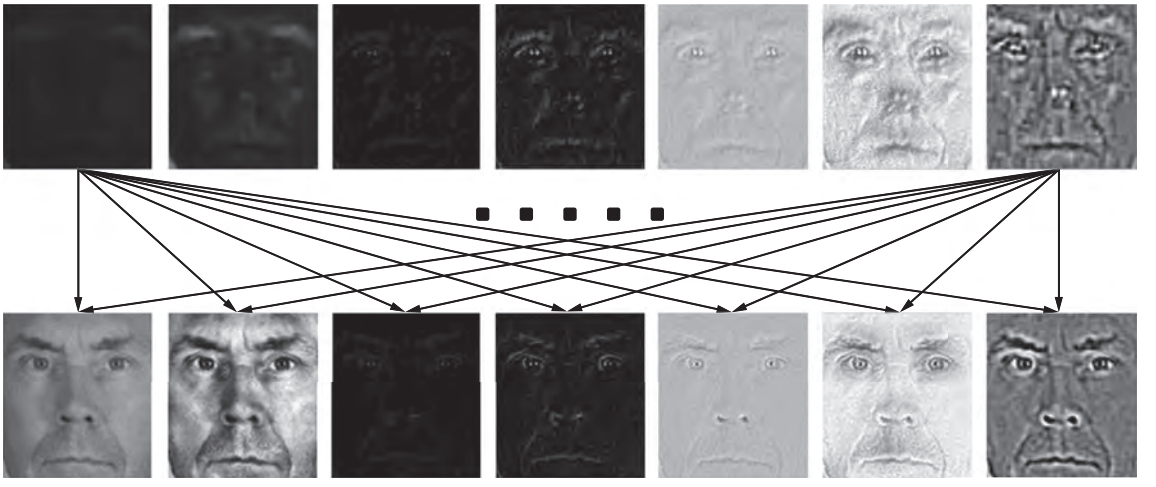


Figure 2: An overview of the cross-photometric empirical study done. Notice that each representation in the gallery set is matched against all representations in the probe set creating 49 combinations.
(Source: West Virginia University, 2014)

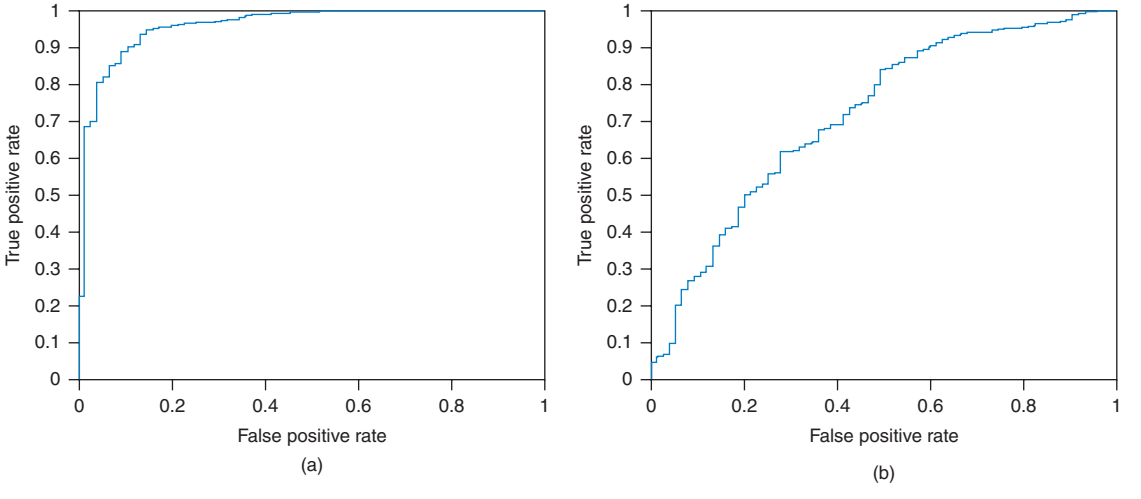


Figure 3: Sample ROCs for two different photometric combinations a) Combination 44: 95.85 percent AUC and b) Combination 2: 71.67 percent.
(Source: West Virginia University, 2014)

Score Level Fusion

Since we know which photometric combinations perform best and which combinations perform worse, based on their respective AUCs, we can take advantage of multiple combinations to further increase the final match score. Simple fusion of all 49 combinations can be performed, as also described by Kalka et al.[4] However, this is not feasible in practice due to the lengthy process of applying all photometric normalization schemes and matching all 49 combinations. Therefore, a second empirical study was conducted to determine

which three combinations, fused together from the top five photometric combinations observed, provide the best matching results for our study. Choosing only three combinations allows for a vast increase in processing time while still maintaining the level of accuracy desired. After the combinations were determined, the testing phase only required these three combinations be used. The final match score S for any probe image is computed by using the following formula:

"Choosing only three combinations allows for a vast increase in processing time while still maintaining the level of accuracy desired."

$$S = \sum_{i=1}^{3} m(Pt_i, Gt_i) \tag{11}$$

Where Pt and Gt are the gallery and probe templates respectively and i represents the photometric normalization combination determined previously. Matching function $m(Pt,Gt)$ corresponds to the matching algorithms listed above, LBP and LTP. An overview of this process is illustrated in Figure 4.

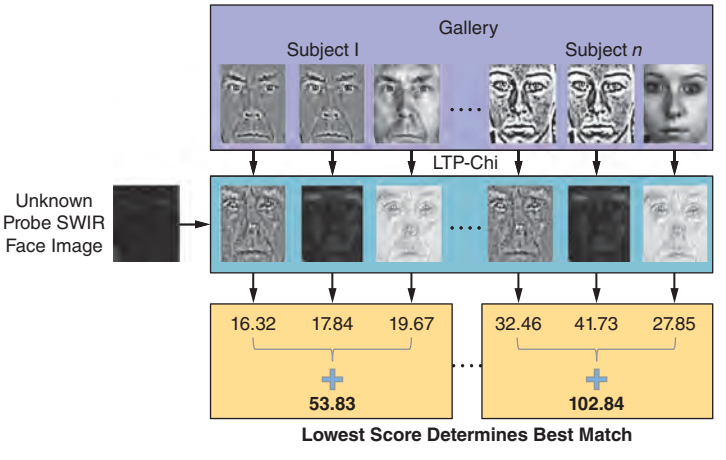


Figure 4: An overview of the score level fusion scheme used to improve cross-spectral face recognition.
(Source: West Virginia University, 2014)

After the completion of this empirical study, it was determined that the following photometric combinations yield the highest rank 1 identification rates:

1. *Gallery:* Tan and Triggs – *Probe:* Tan and Triggs
2. *Gallery:* Tan and Triggs – *Probe:* Contrast-limited adaptive histogram equalization
3. *Gallery:* Contrast-limited adaptive histogram equalization – *Probe:* Tangent-based single-scale retinex

Another advantage of our approach is that instead of performing all 49 photometric normalization combinations as was proposed by Kalka et al.[4], which uses a lot of processing time, our proposed approach only requires three

such combinations. As we will show in the following section, the advantage of this approach is that it manages to increase the performance rate as well as boost the processing speed when compared to the original algorithm described by Kalka et al.[4]

Experimental Setup

Two sets of experiments were conducted to demonstrate the efficiency of our algorithm. First face/eye detection tests were performed on all face datasets. Then, heterogeneous face recognition tests were performed. Our approach is compared with baseline texture-based approaches as well as commercial software (G8 provided by L1 Systems).

Eye Detection Validation

"To test the accuracy of the eye detection method, the normalized average was used."

In order to test the accuracy of our eye detection algorithm, a validation scheme was used. First, we performed our face detection algorithm on each capturing scenario to determine the spatial location of the face region. If the face detection failed, the location was manually marked and subsequently used. To test the accuracy of the eye detection method, the normalized average error was used. This error, indicating the average error between both eyes, was used as the accuracy measure for the found eye locations and can be described as:

$$e = \frac{mean\ (d_{left}, d_{right})}{W} \tag{12}$$

Where d_{left} and d_{right} are the Euclidean distances between the found left and right eye centers with the manually annotated ground truth and w is the Euclidean distance between the eyes in the ground truth image. In the normalized error, $e < 0.25$ (or 25 percent of the interocular distance) roughly corresponds to the width of the eye (corner to corner), $e < 0.10$ roughly corresponds to the diameter of the iris, and $e < 0.05$ roughly corresponds to the diameter of the pupil.

Texture-Based Distance Metrics

"…two different distance metrics were used, the distance transform and the chi-squared metric."

In order to get the final match scores from the LBP and LTP feature vectors, two different distance metrics were used, the distance transform and the chi-squared metric. The distance transform (defined as the distance or similarity metric from image X to image Y) is defined as follows:

$$D(X,Y) = \sum_{Y(i,j)} w(d_x^{K_{Y(i,j)}}(i,j)) \tag{13}$$

Where $K_{Y(i,j)}$ is the code value of pixel (i,j) of image Y, and w is a user controlled penalty function.

The chi-squared distance is defined as follows:

$$\chi^2(n,m) = \frac{1}{2}\sum_{i}^{l} \frac{h_n(k) - h_m(k)}{h_n(k) + h_m(k)} \tag{14}$$

Where h_n and h_m are the two histogram feature vectors, l is the length of the feature vector and n and m are two sample vectors extracted from an image of the gallery and probe sets respectively.

Experimental Results

In this section we discuss the two main experiments we performed. The first one is on eye detection and heterogeneous face recognition, and the second one is discussing the time efficiency (computational complexity) of our proposed fusion approach when compared to original one, and after we incorporate in our design parallel processing.

Eye Detection and Heterogeneous Face Recognition

In order to show the efficiency of our algorithms, we performed both eye detection and heterogeneous face recognition tests on the seven different databases listed above: ground truth (no glass), 0-percent tint with active SWIR and visible illumination, 80-percent tint with active SWIR and visible illumination, and the Solarcool-2 Graylite glass with active SWIR and ambient lighting. One hundred forty subjects were used for one-to-one comparison of visible gallery images to SWIR probe images. We compared our proposed heterogeneous face recognition scheme with multiple algorithms, including the baseline texture-based approaches (LBP, LTP) with both distance metrics, a commercial face recognition algorithm (L1 Systems G8), and the original cross-photometric score level fusion approach described by Kalka et al.[4] An overview of the results of this experiment can be found in *Table 1*.

"We compared our proposed heterogeneous face recognition scheme with multiple algorithms…"

	Eye Detection (% *@e<.25)*	**LBP – χ^2**	**LBP – DT**	**LBP – χ^2**	**LTP – DT**	**L1's G8**	**Kalka et al [4]**	**Proposed Method**
Ground Truth	96.81	62.14	80.14	70.71	86.43	81.43	35.00	**94.26**
0% Visible Lighting	100.00	38.57	50.00	47.86	53.57	56.43	19.29	**67.86**
0% Active SWIR Lighting	97.86	12.86	17.86	19.29	24.29	12.14	5.71	**35.71**
80% Visible Lighting	99.29	26.43	**40.82**	20.41	30.61	6.12	13.57	34.69
80% Active SWIR Lighting	98.57	7.14	12.14	8.57	13.57	4.29	5.00	**23.57**
Solarcool Visible Lighting	36.96	39.29	48.57	52.14	55.00	**69.29**	23.57	61.43
Solarcool Active SWIR Lighting	57.61	0.71	0.71	0.71	0.71	1.43	0.71	**4.29**

Table 1: Experimental results for the proposed eye detection and heterogeneous face recognition algorithm. Eye detection (second column) uses the normalized average error at $e = 0.25$ while the face recognition results show the rank 1 percentage. (Source: West Virginia University, 2014)

In these results, the eye detection (second column) reports the percentage of eyes whose normalized average error is $e < 0.25$. In reference to the face recognition studies performed (columns 3 through 9), the percentage of subjects who obtained a rank 1 identification rate was reported. Note that in all

cases, except for the 80-percent visible lighting and Solarcool visible lighting, our proposed algorithm outperformed all other algorithms.

Time Efficiency

One of the main drawbacks to the algorithm proposed by Kalka et al.[4] is the length of time that it takes to complete a single probe-gallery match. Because the algorithm is essentially repeating the same process 49 times (just with different image representations), the time to match one probe SWIR image to a gallery of visible images is impractical, in an operational standpoint, and grows as the size of the gallery grows. Therefore, in order to increase speed, as well as increase matching accuracy, our empirical study, described earlier in the section "An Empirical Study of Photometric Normalization," was performed to narrow the photometric normalization combinations down from 49 to 3. Although this helps speed up the matching process by approximately 18.5 times, it's still too slow to have any practical matching ability. In order to decrease the process time further, parallel processing was used. Eight cores were used simultaneously to perform the matching algorithm described above. All experiments were conducted on a gallery of 140 subjects. The results for the time efficiency test can be found in Table 2, where we can see the time it takes (in seconds) for a single probe to match a gallery image. All experiments described above were performed on a 64-bit Windows 7 machine with 12 GB of RAM running Intel® Core™ i7 CPU at 3.2 GHz using MATLAB R2012b. The MATLAB Parallel Processing Toolbox was used to test the parallel processing speeds.

"…the time to match one probe SWIR image to a gallery of visible images is impractical in an operational standpoint, and grows as the size of the gallery grows."

Algorithm	Kalka et al.[4]	Proposed	Proposed with Parallel Processing
Avg. Time (sec)	12.059	0.650	0.207

Table 2: Results of the time efficiency test. All times reported are in in seconds for a single probe to match a gallery face image.
(Source: West Virginia University, 2014)

"…the proposed method, when using parallel processing, further speeds up the time it takes to make a single gallery to probe match."

As we can see in Table 2, the proposed method, when using parallel processing, further speeds up the time it takes to make a single gallery to probe match. Also it is clear that by reducing the number of photometric normalizations, our algorithm is much faster and more efficient than the algorithm proposed by Kalka et al.[4]

Conclusions and Future Work

In this article, we studied the advantages and limitations of performing cross-spectral face matching (visible against SWIR) in different challenging scenarios represented in a set of different face databases. We first showed that our eye detection approach performs extremely well on our assembled face databases. Specifically, we managed to achieve an eye detection rate of greater than 96 percent in the majority of the scenarios we investigated. This achievement is

critical in improving the efficiency of the automated face recognition system proposed. Secondly, we proposed an approach that enhances the capability of the original cross-photometric score level fusion proposed by Kalka et al.[4] Our experimental results showed that the use of a fairly small set of photometric normalization combinations is sufficient to yield desirable face recognition scores due to our empirical study that determined the efficiency in matching probe to gallery face images under specific pairs of photometric normalization algorithms. In other words, our study showed that a smaller number of combinations results in an increase of rank-1 identification rate, in addition to an improvement of the computational complexity of the proposed approach, that is, when using a subset vs. a complete set of photometric normalization techniques and their combinations. By using the best three photometric normalizations instead of all 49 combinations tested, the time required for a single gallery to probe face match increased by more than 18.5 times. In addition, by utilizing MATLAB's parallel processing toolbox, we were able to further increase the matching speed by 58 times when compared to the original matching algorithm.

"…by utilizing MATLAB's parallel processing toolbox, we were able to further increase the matching speed by 58 times when compared to the original matching algorithms."

Another benefit of our face matching algorithmic approach is that in all but two scenarios, it outperformed all other face matching algorithms tested. We obtained a rank 1 identification rate of 94.26 percent when using our ground truth data, which is about 2.7 times improvement over the original algorithm[4] and a more than 7-percent improvement over LTP-DT, the algorithm that achieved the second-best rank 1 identification rate. The only scenarios where our proposed algorithm did not outperform all others were when we used the 80-percent visible lighting and the Solarcool visible lighting face datasets.

"We obtained a rank-1 identification rate of 94.26 percent, which is about 2.7 times improvement over the original algorithm…"

For future work we are planning to extend our experiments in the field of heterogeneous face recognition. One of our focus areas will be to test our proposed *enhanced cross-photometric score level fusion algorithm* on face images captured on different bands than the SWIR and visible ones we used in this study. Studies in the near IR and passive IR bands would be beneficial to show the robustness of this algorithm, even when we have to deal with face images affected by different image degradation factors, such as camera and motion noise, or information loss due to the acquisition of face images at long ranges (close to or further than the capabilities of the camera used).

Acknowledgements

This work is sponsored through a grant from the Office of Naval Research (N00014-08-1-0895) Distribution A - Approved for Unlimited Distribution. The authors are grateful to the students and staff at West Virginia University, especially Dr. Bojan Cukic, Dr. Jeremy Dawson, as well as WVU students including Nnamdi Osia, Neeru Narang, and Jason Ice, for their assistance in this work.

References

[1] Grother, P., G. Quinn, P.J. Phillips, "Report on the Evaluation of 2D Still-Image Face Recognition Algorithms," *Nat'l Inst. Of Standards and Technology interagency/internal report (NISTIR) 7709*, 2010.

[2] Klare, B. and A. Jain. "Heterogeneous face recognition: Matching NIR to visible light images," *IEEE International Conference on Pattern Recognition (ICPR)*, pp. 1513–1516, 2010.

[3] Zhu, J. Y., W.S. Zheng, J.H.Lai, and S. Li, "Matching NIR Face to VIS Face using Transduction," *IEEE Transactions on Information Forensics and Security*, 2014.

[4] Kalka, N., T. Bourlai, B. Cukic, and L. Hornak, "Cross-spectral Face Recognition in Heterogeneous Environments: A Case Study on Matching Visible to Short-wave Infrared Imagery," *International Joint Conference on Biometrics (IEEE, IAPR)*, pp. 1–8, 2011.

[5] Bourlai, T., A. Ross, C. Chen, and L. Hornak, "A Study on using Middle-Wave Infrared Images for Face Recognition," *SPIE Biometric Technology for Human Identification IX*, pp. 83711K–83711K, 2012.

[6] Osia, N. and T. Bourlai, "Holistic and Partial Face Recognition in the MWIR Band Using Manual and Automatic Detection of Face-based Features," *IEEE Conf. on Technologies for Homeland Security*, pp. 273–279, 2012.

[7] Mendez, H., C. San Martin, J. Kittler, Y. Plasencia, and E. Garcia-Reyes, "Face recognition with LWIR imagery using local binary patterns," *Advances in Biometrics*, Springer, pp. 327–336, 2009.

[8] Akhloufi, M. A., A. Bendada, and J.C. Batsale, "Multispectral face recognition using nonlinear dimensionality reduction," *Proceedings of SPIE, Visual Information Processing XVIII*, volume 7341, 2009.

[9] Tan, X. and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, pp. 19:1635–1650, 2010.

[10] Bourlai, T., N. Kalka, D. Cao, B. Decann, Z. Jafri, C. Whitelam, J. Zuo, D. Adjeroh, B. Cukic, J. Dawson, L. Hornak, A. Ross, and N. A. Schmid, "Ascertaining human identity in night time environments," *Distributed Video Sensor Networks*, Springer, pp. 451–467, 2011.

[11] Ice, J., N. Narang, C. Whitelam, N. Kalka, L. Hornak, J. Dawson, and T. Bourlai, "SWIR imaging for facial image capture through tinted materials," *SPIE Defense, Security, and Sensing. International Society for Optics and Photonics*, pp. 83530S–83530S, 2012.

[12] Jobson, D., Z. Rahman, and G. Woodell, "Properties and performance of a center/surround retinex," *IEEE Transaction on Image Processing*, pp. 6(3):451–462, 1997.

[13] Whitelam, C., Z. Jafri, and T. Bourlai, "Multispectral eye detection: A preliminary study," *IEEE International Conference on Pattern Recognition*, pp. 209–212, 2010.

[14] Whitelam, C., T. Bourlai, and I. Kakadiaris, "Pupil Detection under Lighting and Pose Variations in the Visible and Active Infrared Bands," *IEEE Workshop on Information and Forensics Security (WIFS)*, pp. 1–6, 2011

[15] Short, J., J. Kittler, and K. Messer, "A comparison of photometric normalization algorithms for face verification," *IEEE Conference on Automatic Face and Gesture Recognition*, pp. 254–259, 2004.

[16] Bourlai, T., "Short-Wave Infrared for Face-based Recognition Systems," *SPIE Newsroom Magazine - Defense & Security*, 2012 [Invited Article].

[17] Osia, N. and T. Bourlai, "A Spectral Independent Approach for Physiological and Geometric Based Face Recognition in the Visible, Middle-Wave and Long-Wave Infrared Bands," *Image and Vision Computing, Journal - Elsevier,* 2014.

[18] Bourlai, T. et al., "Applications of Passive Infrared Imaging to Forensic Facial Recognition," *InfraMation (Thermal Imaging Leading User's Conference - Organized by FLIR), Loews Royal Pacific*, 2013.

[19] Bourlai, T., "Mid-wave IR Face Recognition Systems," *SPIE Newsroom Magazine - Defense & Security*, pp. 1–3, 2013.

[20] Bourlai, T. and B. Cukic, "Multi-Spectral Face Recognition: Identification of People in Difficult Environments," *IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 196–201, 2012.

Author Biographies

Cameron Whitelam received the BS degree in Biometric Systems from West Virginia University. From January, 2011, he started pursuing a PhD in computer engineering in the multispectral imaging lab under the direction of Dr. Thirimachos Bourlai. His main areas of research include automatic eye detection in visible, shortwave infrared, and 3D images. He also focuses on facial recognition algorithms and their performances. In the summer of 2012, he was awarded a prestigious internship at the Office of Naval Research where less than 5 percent of applicants were accepted. He has coauthored numerous conference papers on automatic eye detection and has presented his

work internationally. He has also coauthored a book chapter in *Video Sensor Networks* (published by Springer).

Thirimachos Bourlai holds a BS (M.Eng. Equiv.) degree in Electrical and Computer Engineering from Aristotle University of Thessaloniki (Greece), an MS in Medical Imaging with Distinction and a PhD (Facial Recognition) degree from the University of Surrey (U.K.). He completed his first postdoctoral appointment in 2007 at the University of Surrey and his second in 2009 in a joint project between The Methodist Hospital and the University of Houston, TX (USA), in the fields of thermal imaging and human-based computational physiology. From 2008 to 2012 he worked at West Virginia University, first as a visiting research assistant professor and then as a research assistant professor. Since August of 2012 he has been an assistant professor at WVU, where he is the founder and director of the Multi-Spectral Imagery Lab. He is also an adjunct assistant professor at the School of Medicine, Department of Ophthalmology at WVU. He is actively involved in several applied projects, in the areas of biometrics, biomedical imaging, deception detection, and human computer interaction. The projects have been funded by DoD-ONR, DoD-DTRA, FBI, CITeR, a National Science Foundation (NSF) Industry/ University Cooperative Research Center (I/UCRC), and TechConnect WV. He has published numerous book chapters, journals, magazines, and conference papers. T. Bourlai is also a Senior Member of IEEE and a member of the IEEE Communications Society, the IEEE Signal Processing Society, the IEEE Biometrics Compendium, the National Safe Skies Alliance, the Biometrics Institute Organization, the National Defense Industrial Association, the European Association of Biometrics, and the American Physical Society.

ADDING NONTRADITIONAL AUTHENTICATION TO ANDROID*

Contributors

Ned M. Smith
Software and Solutions Group,
Intel Corporation

Micah Sheller
Intel Labs,
Intel Corporation

Nathan Heldt-Sheller
Software and Solutions Group,
Intel Corporation

In this article we describe how passive authentication factors can be used to improve user authentication experiences. We focus our research on motion, vicinity, and location sensors, but passive authentication is not limited to this set of sensors. We show how their use can also improve security when used in combination with traditional authentication factors. User experiences are better when the device continuously monitors passive authentication factor types because the number of authentication challenges can be reduced for a given set of usage scenarios. We show how multiple factors can be fused and how False Accept Rate (FAR) and False Reject Rate (FRR) relate to confidence. Confidence is useful for determining when it is most appropriate to protect a device resource and to re-prompt the user for authentication credentials. An implementation of our research has been applied to Android* platforms to demonstrate implementation feasibility and usability.

Introduction

Significant change in authentication techniques hasn't happened in computing since the PC revolution put a keyboard in nearly every home in America and across the globe. As the primary input device, it is only natural that passwords would become the primary form of authentication for every application, system, and service. The advent of PDAs, smartphones, and tablets meant people no longer could be tethered to their keyboards. Touch-screen displays turned the screen into an input device and virtual keyboards made it possible for users to continue using their passwords. However, the clunky hunt-and-peck user experience of typed passwords on a touch screen gave way to innovative variations on passwords. *Swipe authentication,* a variation on a PIN, meant users needed only remember a geometric shape and play "connect the dots," which many find easier to remember and input. The innovation in touch screens enabled innovation in authentication techniques.

Sensors are another technology spreading aggressively throughout mobile form factors and in the anticipated Internet of Things. Ubiquitous sensors set the stage for a sea change of innovation in authentication techniques. This article explores several nontraditional authentication techniques using some of the most common sensors available on an average smartphone or tablet computer.

Android Authentication Current Practice

In this section we describe the current state of Android authentication.

Login and Lock Screen

Most are familiar with the current approach to authentication on a smartphone. When the system is powered up or resumes from a sleep state, the user is presented with a *lock screen*. The lock screen presents the user with an authentication challenge—usually a password. But users may configure alternatives such as those discussed in the introduction. A problem with the lock screen concept is some information isn't sensitive and hence is inconveniently accessed only by entering the password. Workarounds have been explored such as an edge swipe that presents the user with a control panel to access the music player, camera, flashlight, battery level indicator and other status. The addition of multi-account support in the KitKat release of Android made it necessary to integrate a method for switching users so the correct lock screen is shown.

"A problem with the lock screen concept is some information isn't sensitive and hence is inconveniently accessed only by entering the password."

Application Access through the Binder

Android users depend heavily on applications being their emissaries in the online world. When the user needs to authenticate to a web service or a remote device, a mobile application is required to collect user credentials and present them to the remote entity. To obtain credentials, apps must communicate with an Android service called the *binder*. The binder grants privileges to applications to act on the users' behalf having access to user credentials.

Web Authentication and Web Single Sign-on

Web services manage user accounts centrally at their respective web sites. Doing so means users have multiple identities spread across the Web, each referring to the same individual but constrained by the account management procedures at each web site. The user experiences cognitive overload keeping track of multiple accounts and credentials. Large social media sites attempt to relieve the cognitive overload and usability inconveniences by offering web single-sign-on (SSO) services. These services work by allowing users to log in first to the social media site; other web sites then accept this single authentication in lieu of their own, thus making subsequent web site accesses transparent.

Mobile applications generally support using a social media login because it gets authentication out of the way upfront so the application is free to focus on the rest of the user experience.

Login Pain Points

A typical Android login experience requires the user to authenticate to the device resulting in the removal of the lockscreen. If users access web services they are prompted again for authentication credentials. Single sign-on helps alleviate the usability pain point, but single-sign-on creates an access token that grants access broadly without re-verifying the user is authenticated. Users must take care to ensure the social media site login strength is equal to or stronger than the user's most sensitive sites. Many financial institutions have refused to support web SSO for this reason.

"Single sign-on helps alleviate the usability pain point, but single-sign-on creates an access token that grants access broadly without re-verifying the user is authenticated."

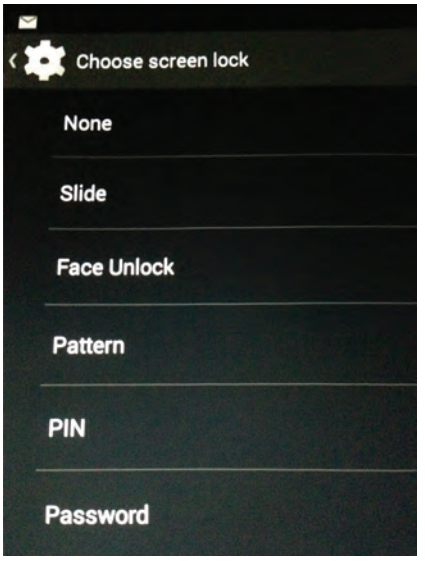


Figure 1: Android ScreenLock options (Source: Intel Corporation, 2014)

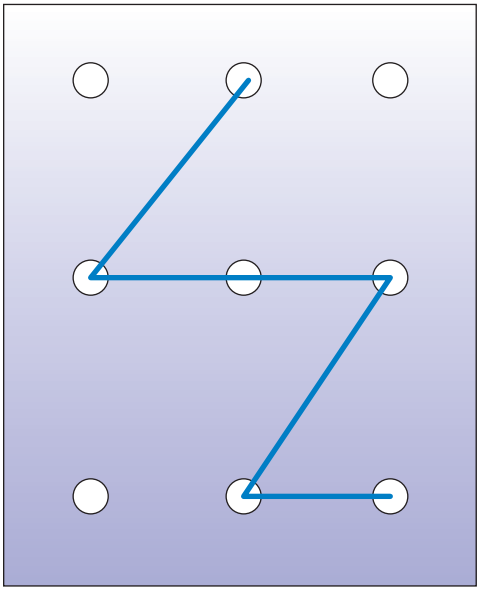


Figure 2: Android pattern swipe authentication factor (Source: Intel Corporation, 2014)

Many financial institutions have implemented two-factor authentication that requires use of a validation token, preregistered device or computer, or stronger passwords.

The collection of authentication and sign-on options and the arbitrarily defined reach of web single sign-on confuse users. Users are looking for consistency in authentication and access experiences given the line between what is local (on the device) and remote (on another device or service) is often blurred in the mind of the user.

Android Authentication beyond Passwords

The Android environment offers several authentication alternatives through open source repositories, Google Play store applications, and OEM-supplied extensions.

The Android Open Source Project (https://source.android.com) has native support for four authentication factors: password, PIN, pattern, and facial recognition. Users select an authentication method used to unlock the device using Android ScreenLock settings (select Settings, then Security, and then Screen lock), as shown in Figure 1.

The *Slide* unlock method doesn't require user authentication, but it does require a user action.

Our research shows there are usability issues with PINs and passwords. Users have trouble remembering multiple passwords so they reuse them and write them down. User subversion of proper password management makes us question whether passwords are as safe as conventional wisdom might suggest.

Pattern Swipe is a password alternative to PINs.

The pattern swipe authentication method is similar to a PIN except there are no printed numbers on the PIN pad—only dots. The user connects the dots following some pattern they select, as illustrated in Figure 2. The pattern can easily be mapped to a PIN by overlaying a numbered PIN pad, making it as easy to write down or to share as PINs. Many find patterns easier to remember than numbers so it is a popular alternative to PINs.

Android also supports facial recognition using the Viola-Jones[1] algorithm implemented in OpenCV.[2] The algorithm was among the first practical algorithms given optimal conditions, but differences in lighting and camera placement presents usability challenges for some users.

Android applications may control the screen lock behavior using Keyguard Manager:

```
KeyguardManager keyguardManager = (KeyguardManager)
getSystemService(Activity.KEYGUARD_SERVICE);
```

```
KeyguardLock lock = keyguardManager.newKeyguardLock
(KEYGUARD_SERVICE);
...
lock.reenableKeyguard();
...
lock.disableKeyguard();
```

KeyguardManager relies on the user's LockScreen settings to define the user action needed to restore the user's home screen from the lock screen state. If a new authentication factor is added, LockScreen settings likely are updated and the user may need to select the new factor. This approach doesn't work well when multiple factors are to be used.

"Users are looking for consistency in authentication and access experiences given the line between what is local (on the device) and remote (on another device or service) is often blurred in the mind of the user."

Authentication Using Nontraditional Sensors

This article distinguishes between more traditional *active* authentication factors and nontraditional *passive* factors. Active factors require users to perform an action such as swiping a finger, entering a password, or positioning a camera at a face. Active factors often require *liveness testing*, where the system tries to determine if a live human is interacting with the device. Liveness testing may further require user activity such as blinking eyes or verbalizing a randomly generated pass phrase. Active factors interrupt the user's train of thought. The distraction may result in several minutes of unproductive time waiting for the user to adjust to the context change.

Conversely, passive authentication may occur in the background, without the user having to take action or otherwise be distracted. Sensors that perform passive authentication typically monitor user behavior continually while evaluating confidence that current behavior aligns with expected behavior. Passive authentication is less intrusive, allowing users to experience more rewarding interactions with a computing device.

"Sensors that perform passive authentication typically monitor user behavior continually while evaluating confidence that current behavior aligns with expected behavior. Passive authentication is less intrusive, allowing users to experience more rewarding interactions with a computing device."

This article considers three passive factor types: *motion*, *location,* and *vicinity*. These and other passive factors may be referred to as *behaviometrics* because they combine behavior with authentication. We discuss our research in the context of the Android OS but acknowledge these concepts are broadly applicable.

Motion

Motion sensors are contained in virtually every mobile device shipped today. They almost always operational when the device is powered because other system services require motions sensing - such as the screen rotation feature. A motion sensor is a collection of several sensors integrated into a single package. They may have 6[3] or 9[4] axes of motion including acceleration in X, Y, and Z planes; gyro; and gravitational orientation.

Motion can be an effective passive authentication factor by creating a template image of motion data that is collected while a specific user is interacting with

the device. Most users are unaware of the subtle ways in which they move and behave differently from someone else. Most often these subtleties are filtered out by traditional motion sensing algorithms—for example a 90-degree rotation. As an authentication factor, we want to capture the anomalies in common motions and then look for repetitions that are specific to particular individuals.

In addition to behavior template matching, motion information can be used to establish common modes of context. Device state such as "in pocket," "laid on level surface," or "moving along commute path" can be used in modifying authentication confidence or changing authentication policy.

Location

Location often requires uses of multiple types of sensors. Classifications such as *outdoor* location, *indoor* location and *logical* location may be used as well. Outdoor location can be determined using GPS or GSM to triangulate cellular telephone towers. Indoor location can be found using Bluetooth* iBeacons or using WPS to triangulate Wi-Fi* access points. Logical location uses managed Ethernet ports on a network switch to roughly approximate an indoor location within the limitations of Ethernet cabling specifications.

Location may combine several location sensing techniques to achieve highly accurate location coordinates in three dimensions. They are often provided together by location frameworks.

Location can be used as a passive authentication factor by collecting training data using the location sensors over a period of time. Most people frequent the same places and follow well-known routes between them. If a user breaks with tradition and takes an unexpected excursion to someplace new, the training system may authenticate the user employing other factors while the location factor receives additional training.

Location alone may not be a strong authentication factor, but in connection with other factors, it builds a more complete picture of expected user behavior. Location is also particularly helpful in adjusting authentication policy, for example, between meaningful locations such as work, home, or at a public location such as a mall.

Vicinity

Vicinity refers to the spatial relationship between the user and a device. A refinement of this notion extends that relationship between two or more devices such that the presence of a collection of devices may be an indication of the presence (or absence) of one or more individuals. Bluetooth and Near Field Communication (NFC) are examples. They have limited radio ranges, implying a limited distance within which the devices may exist spatially.

Another class of vicinity sensor detects existence of physical objects. Although the resolution is often insufficient to identify a specific individual, they can detect movement, and the presence or absence of a variety of shapes.

Vicinity can be used as a passive authentication factor by collecting training data about the meaningful spaces the user frequents such as an office area at work, rooms within the user's home, or within the car. Different spaces have different vicinity signatures. Failure of an attacker to exactly replicate one of these spaces will diminish his or her ability to fake an identity.

As with the other passive factors, by itself vicinity doesn't promise reliable strong authentication, but when combined with other factors it contributes to a picture of the user that is increasingly challenging for attackers to replicate. Passive authentication frees the user from having to be constantly interrupted with authentication challenges.

Continuous Authentication

Traditional authentication factors are "active" because they require the user to halt application flow to perform the authentication. Applications assume the authenticated user is still present and in command of the device for several minutes or days following an authentication challenge. This presents a problem: the application assumes the authentication that took place moments ago is still relevant. Such assumptions are based on conventional wisdom that security circumstances won't change drastically following an authentication event. Yet people frequently get interrupted, walk away, or get distracted. Designers of banking and financial services web sites illustrate this phenomenon. They often monitor keyboard activity and if inactive for a few minutes will automatically log the user off. In reality, the security circumstances may have changed immediately following successful authentication, but monitoring keyboard activity is the only indication of user presence available. If the user happens to be reading (and not tapping the keyboard) or has switched to a different window but is still physically present, then the website timer expires and the user must reauthenticate when focus returns. Reauthentications come at a cost to user productivity and user experience. Studies[5][6] show users are inconvenienced by interruptions and that it takes significant work to reestablish the user's context. This includes reestablishing connections to services, reopening applications, finding the user's previous position within a data set, and allowing time for the user to regain mental context. In short, users are significantly inconvenienced by reauthentication prompts.

Continuous authentication is an alternative approach that is made feasible by using *passive* factors—like motion. Passive factors don't require user interruption to respond to a challenge. Instead, they monitor ambient indicators of the particular user's presence. Motion, vicinity of user, vicinity to other devices, location, ambient light, and ambient noise are examples of sensing that could be used for passive authentication.

A trustworthy active authentication factor's impact on authentication confidence over time is shown in Figure 3. Confidence is momentarily high and then decays quickly. The factors that influence confidence degradation are essentially unknown. Conversely, when passive authentication factors are applied continuously following a successful active authentication confidence degrades slowly based on measurable feedback from passive factors.

"Continuous authentication is an alternative approach that is made feasible by using passive factors—like motion. Passive factors don't require user interruption to respond to a challenge. Instead, they monitor ambient indicators of the particular user's presence."

"Conversely, when passive authentication factors are applied continuously following a successful active authentication confidence degrades slowly based on measurable feedback from passive factors."

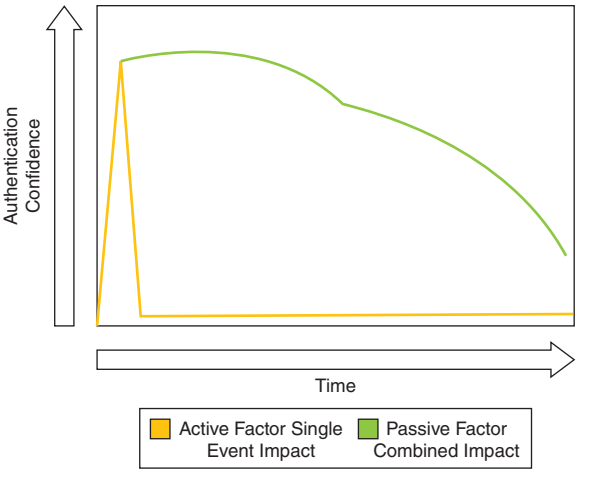


Figure 3: Graph showing authentication confidence impact for active and passive factors (Source: Intel Corporation, 2014)

The dip in the green confidence line suggests the loss of a passive factor, such as a Bluetooth vicinity device moving out of radio range. Confidence should trend lower over time to account for the possibility that a well-orchestrated attack could eventually compromise all passive factors.

Passive Multifactor Authentication

This section explains how multiple passive authentication factors can be combined to increase confidence that a subject is indeed authentic versus a singleton factor from the same set of factors.

Multifactor Authentication

"The quality of a biometric factor often relies on two metrics, False Accept Rate (FAR) and False Reject Rate (FRR)."

The quality of a biometric factor often relies on two metrics, False Accept Rate (FAR) and False Reject Rate (FRR). FAR computes the probability that the system identifies the user given it actually isn't the user.

$$FAR = P(System = User \mid Reality = !\,User)$$

FRR computes the probability that the system doesn't identify the user given it actually is the user.

$$FRR = P(System = !\,User \mid Reality = User)$$

When considering a single factor for verifying an identity, the System (S) relies on a matching function $m()$ that compares a current sensor sample to a template value known to describe the user.

$$S = m(T, x);\ \textit{where } x \textit{ is a sample and } T \textit{ is the template}$$

Calculating FAR and FRR involves a pair of relatively straightforward experiments: present the system with a representative sample of users and

record the responses. The samples and results form your data set D of ordered pairs (u,x), where u is the user sampled and x is the sensor sample. You then run each sample against the template for each user, categorizing each run as one of the following sets:

$$fp = false\ positives = \{(T,u,x) | T \neq T_u, m(T,x) = true\}$$
$$tp = true\ positives = \{(T,u,x) | T = T_u, m(T,x) = true\}$$
$$fn = false\ negatives = \{(T,u,x) | T = T_u, m(T,x) = false\}$$
$$tn = true\ negatives = \{(T,u,x) | T \neq T_u, m(T,x) = false\}$$

Then:

$$FAR = \frac{|fp|}{|fp| + |tn|}$$

$$FRR = \frac{|fn|}{|fn| + |tp|}$$

A system that processes multiple factors requires fusing function $f()$ that maps each matching function to the overall assertion of S.

$$S = f(m1(T1,x1), m2(T2,x2), \ldots, mn(Tn,xn))$$

In the majority of cases, the outputs of the matching functions $m1\ldots n()$ are n-ary, and if not, are often intended to be treated as binary according to some threshold chosen to match the use case. It should be noted that matching functions that output scalars are not typically outputting raw probabilities.

Given matching functions that provide n-ary outputs, our fusion function then takes n-ary inputs, which means that a given fusion function $f()$ can be expressed as a truth table.

The calculations for multiple factor *FAR* and *FRR* follow in much the same way as for single-factor, except that $f()$ is substituted for $m()$. Note, however, that if the sample data set D does not contain $x_{1\ldots n}$ for all samples, the calculation requires considerably more work. This article does not discuss strategies for estimating the *FAR* and *FRR* for multiple factor systems in these cases.

Authentication Confidence and MFA

Our use models anticipate calculation of authentication confidence that comprehends both multiple factors and continuous authentication. We define an authentication confidence as the probability that a given user is at the system, given that the system has identified the user:

$$CNF = P(Actual = User | System = User)$$

"Our use models anticipate calculation of authentication confidence that comprehends both multiple factors and continuous authentication."

Unfortunately, we cannot devise an appropriate experiment for $P(Actual = User | System = User)$ as we can for FAR/FRR. Instead, we must estimate CNF from FAR/FRR. FRR and CNF are related by Bayes' theorem:

R is Reality, *and R = u is the event that user u is at the system*

S is the System, *and S = u is the event that the system authenticates user u*

$$CNF = P(R = u \mid S = u) = P(S = u \mid R = u) * \frac{P(R = u)}{P(S = u)}$$

$$CNF = (1 - P(S \neq u \mid R = u)) * \frac{P(R = u)}{P(S = u)}$$

$$CNF = (1 - FRR) * \frac{P(R = u)}{P(S = u)}$$

We can eliminate the term $P(S = u)$ by using FAR in our calculation as well, ultimately reducing the equation to:

$$CNF = \frac{1 - FRR}{1 - FRR + \frac{FAR}{P(R = u)} - FAR}$$

"Unfortunately, we must still provide a value for P(R = u), which is the probability that a given user is at the system at the time of an authentication."

Unfortunately, we must still provide a value for $P(R = u)$, which is the probability that a given user is at the system at the time of an authentication. For a private system in a secure location, this could be close to 1. For a shared system in a public place, this could be much lower.

In practice, we plot CNF against $P(R = u)$ for our measured FAR/FRR. For example, given an FAR of 0.01 and an FRR of 0.02, we might generate the curve shown in Figure 4.

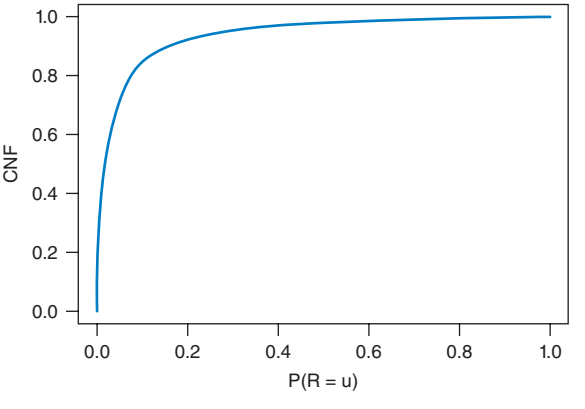


Figure 4: Graph showing a desirable confidence curve
(Source: Intel Corporation, 2014)

"The assumed P(R = u) should pass muster with the relevant domain experts, not too unlike gathering priors for other Bayesian methods."

From this curve, we determine whether the CNF value necessary for our given solution requires assuming an acceptable $P(R = u)$. The assumed $P(R = u)$ should pass muster with the relevant domain experts, not too unlike gathering priors for other Bayesian methods.

As it happens, plotting CNF against $P(R = u)$ for various pairs of FAR/FRR shows that importance of choosing a good $P(R = u)$ varies with the accuracy of the system. We consider the hypothetical FAR/FRR rates for hypothetical passive biometrics in Table 1 and Figure 5.

Biometric	FAR	FRR
Handy Co. palm vein recognition	0.00001	.001
Reliabull Tech motion recognition	0.1	0.2
Trusteaze Inc. device vicinity	0.2	0.2
Where 'R' Us Corp location	0.2	0.3

Table 1: FAR and FRR for hypothetical biometrics used in Figure 5
(Source: Intel Corporation, 2014)

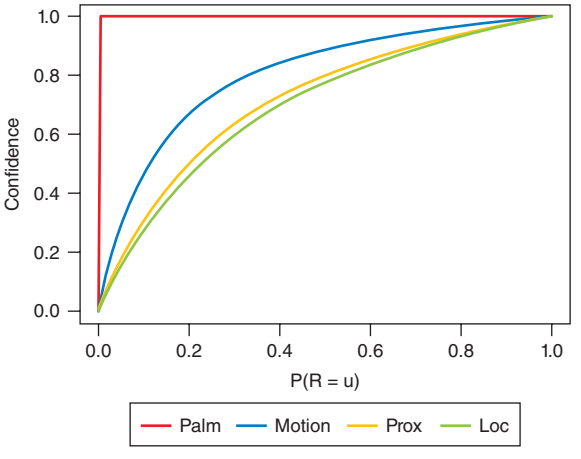


Figure 5: Graph showing confidence curves for four authentication factors: palm vein, motion, vicinity and location
(Source: Intel Corporation, 2014)

Less accurate biometrics, such as our hypothetical passive biometrics, require a much more accurate guess at $P(R = u)$ to give the proper confidence outputs than our hypothetical active, accurate palm biometric.

Depending on the probabilistic dependence of our factors, a multifactor solution could greatly improve our FAR/FRR. For example, if we used the fusion function $f()$, described by the truth table (essentially, a majority vote system) shown in Table 2.

Assuming close to independence between the biometrics, this $f()$ would yield an FAR of ~0.05 and a FRR of ~0.09, giving us the confidence curve (the red line) shown in Figure 6.

Tuning confidence properly requires the right combination of passive factors, fusion function, confidence target, and acceptable estimation of $P(R = u)$.

"Less accurate biometrics, such as our hypothetical passive biometrics, require a much more accurate guess at P(R = u) to give the proper confidence outputs than our hypothetical active, accurate palm biometric."

f()	Motion	Vicinity	Location
T	T	T	T
T	T	T	F
T	T	F	T
F	T	F	F
T	F	T	T
F	F	T	F
F	F	F	T
F	F	F	F

Table 2: Example truth table for a simple majority vote fusion function applied to our hypothetical biometrics, used in Figure 6.
(Source: Intel Corporation, 2014)

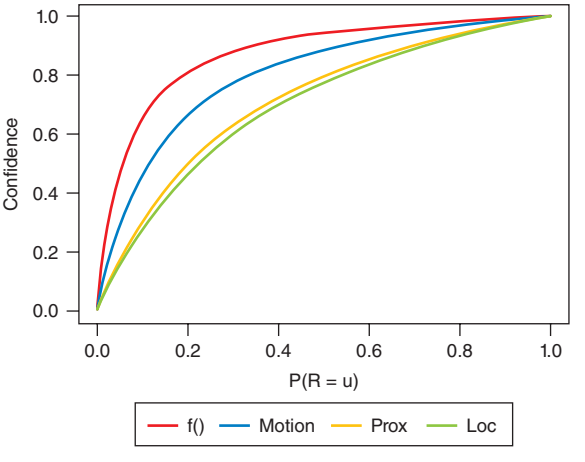


Figure 6: Graph showing confidence improvement for a fusion function *f*() from three confidence curves that are individually less desirable than *f*()
(Source: Intel Corporation, 2014)

Ground Truth

Ground truth establishes what sensor readings are in reality expected for a given user behavior or biometric. Ground truth can validate our methodology by comparing computed confidence values to our ground truth data set. Extant differentials may indicate the need to revisit our assumptions for how $P(R = u)$ is computed.

Ground-truth collection requires a statistically significant population of individuals who participate in data collection by annotating sensor data with characteristic behaviors. Characteristic behavior data helps during data analysis to recognize patterns indicative of the expected behavior. Pattern data assist in

creation of behavior classifiers that are used to train the passive authentication factor algorithms.

When the user trains the system for passive factors, we advocate training with all sensors active to avoid variance that may be observed if discretely trained sensors are later used jointly.

We can rely on user-settable controls to further fine-tune and personalize tolerances.

Collecting ground truth data is a necessary step that helps calibrate passive multifactor authentication classifiers.

Power Impact Avoidance

Different sensors have different impact on power consumption as continuously authenticating factors. Motion factors have very little power impact because motion sensors are often being used by other applications when the CPU is active. For example, the Android KitKat release supports sensor batching that improves power savings.

A vicinity factor based on Bluetooth could have very little if any impact by scheduling vicinity template collection tasks so they occur during the normal duty cycle.

Similarly, the power impact from location sensing can be negligible if location information is sampled while the user has enabled Wi-Fi, a cellular network, and GPS for other reasons.

Use of a passive factor outside of some other application use begins to cause noticeable effects to battery life. User settings are required to configure user's preferences based on the power vs. security tradeoff. Such settings only apply when no other application needs to use the sensor.

"Use of a passive factor outside of some other application use begins to cause noticeable effects to battery life. User settings are required to configure user's preferences based on the power vs. security tradeoff. Such settings only apply when no other application needs to use the sensor."

Both location and vicinity sensors use radio technology. Several strategies can be applied to conserve power:

- Batch sensor events so that they can be scheduled and processed efficiently by the Android OS.[7]
- Align with other applications by warming up the radios once for all subscribers.
- Compress data and use binary data formats instead of text.
- Optimize transfers for the protocol frame; generally fewer large transfers are better.
- Make transfers in both directions whenever possible.
- Cache and sample less often if the higher resolution isn't needed.

Android Integration

Our investigations reveal that Android can be enabled to use nontraditional authentication. A background Android service monitors authentication factors, makes policy decisions, and notifies clients of changes in passive authentication

state. The client interface uses Android interprocess communication architecture—in this case, an AIDL-described Binder interface—to access the authentication service, obtain authentication information, and at times provoke authentication behavior.

"The Android KeyguardManager controls lock screen behavior. By modifying the KeyguardManager to enable it for the passive authentication the number of times the user is disrupted by active authentication is reduced."

The Android KeyguardManager controls lock screen behavior. By modifying the KeyguardManager to enable it for the passive authentication the number of times the user is disrupted by active authentication is reduced.

Android Interprocess Communication Model

Android applications consist of a *caller* and a *callee* process that communicate through an interface (see Figure 7). The method for connecting the caller to the callee is called the *binder*. The binder is a kernel service that establishes a communications path between two otherwise isolated processes. The kernel constructs a caller *proxy* thread that can communicate with a *binder thread* and interface *stub*. The stub implements the interface syntax that the caller expects.

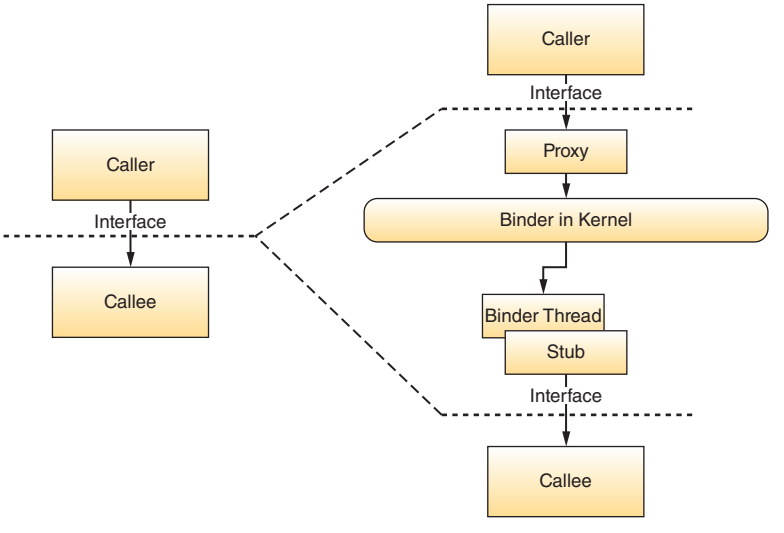


Figure 7: Android interprocess communication (Source: Intel Corporation, 2014)

Smart Locking Using Passive Authentication

Our authentication service uses multiple passive authentication factors to compute a composite authentication confidence value (see Figure 8). Confidence may decay naturally over time but it need not decay quickly if continuous monitoring of passive authentication sensors reveals conditions that existed at the time the user first authenticated using an active factor type (such as password or fingerprint) persist.

User actions that normally might result in screen lock may be avoided using the smart-lock authentication service. Many users put the device to sleep before slipping it into a pocket to go somewhere. Normally, the user will need to

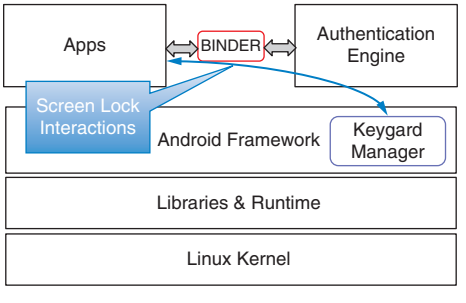


Figure 8: Smart locking using an authentication engine (Source: Intel Corporation, 2014)

reauthenticate. Reauthentication may not be needed with the authentication service or a less-intrusive reauthentication may be warranted instead.

Passive factors continue operating after the device is placed in a pocket or handbag—continually refreshing the authentication confidence value. When the device is used again, authentication confidence is still high; therefore, one of the less-intrusive ScreenLock options may be automatically selected by the authentication engine. For example, the *slide* method may be selected requiring no reauthentication. Or if confidence is even lower the *pattern* method may be selected. If authentication policies identify a context where pattern authentication is inappropriate—easily observed by over-the-shoulder observers, then the location factor may cause confidence to drop significantly resulting in selection of an active factor for reauthentication such as password.

Web Login from a Mobile Application

Android Interface Definition Language (AIDL) allows application developers to define a programming interface that both the client and service agree upon in order to communicate with each other using standard Android Interprocess Communication (IPC) mechanisms. AIDL makes it easy to write applications that have both the client and the server side components (see Figure 9).

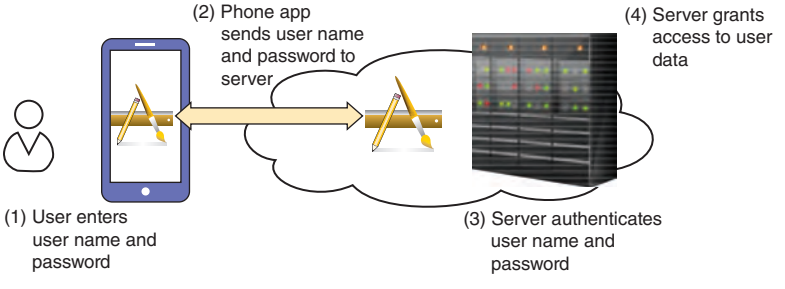


Figure 9: Mobile applications provide the front-end and a web service that authenticates the user
(Source: Intel Corporation, 2014)

A challenge facing passive factor authentication is integration with remote services. A typical Android services application assumes user names and passwords.

```
Example:
//UserInfoService
public iBinder onBind(Intent intent) {
     return new IUserInfoService.Stub() {
          @Override
          public String getInfo(String user, String pass)
throws remoteException {
               If(user.equals("john") return
                    "Hi John, you have 10 messages";
```

```
            return user+pass;
        }
    }
};
```

A ServiceConnection class gives a reference to the service interface:

```
//Service Interface
     service = IUserInfoService.Stub.asInterface
((IBinder) boundService);
```

To access the service:

```
     service.getUserInfo("john", "%#$%^#@#$");
```

The code example reveals a fundamental assumption about remote services: that passwords will be used. To move the Web toward passive factor authentication, remote procedure call frameworks such as AIDL need enhancing.

One reasonable enhancement mechanism uses SAML assertions with digital signatures. If at web registration, the user were to generate asymmetric keys instead of choosing a password, the user's account would contain a public key that could be used to verify the SAML assertion later. The client device needs to control access to the private key using locally enforced combination of active and passive authentication factors.

Client Hardening

As of the Android KitKat release, KeyguardManager doesn't have additional security hardening beyond kernel protections that apply to all framework services. It might be appropriate to consider ways to better protect KeyguardManager, sensors, and the Authentication Engine because these components enforce a security boundary.

"It might be appropriate to consider ways to better protect KeyguardManager, sensors, and the Authentication Engine because these components enforce a security boundary."

Intel platforms support trusted execution environments that can be utilized for security hardening of user authentication building blocks. Intel® Software Guard Extensions™ (SGX™) may be used to implement a trusted execution environment. The role that a TEE should play in authentication systems is described in more detail by Bhargav-Spantzel.[8]

User Convenience

Ethnographic research[9] reveals users desire security but not at the expense of convenience. A recent UX study conducted by Intel discovered users universally expect authentication methods will be convenient where convenience is measured in terms of response time, reliability, and availability. Intel's research characterized users according to three categories: *relaxed*, *active,* and *vigilant*. The study covered three countries in different continents including Asia, North America, and Europe. In total, 18 people were interviewed with over 72 hours of user interaction time.

Relaxed users embrace technology as a vehicle for enhancing the "flow of life." They often view authentication as a major disruption to that flow. *Active* users are motivated to take reasonable precautions and don't want to take unnecessary risks. For example one respondent said "I prefer to stay hidden," and another said, "I need to know how it works to be comfortable." *Vigilant* users take full responsibility for security and often view technology as the weak link. A vigilant user may freely admit having contemplated the possibility of a conspiracy theory, though unlikely or that they believe themselves as being slightly more paranoid than their counterparts. They feel safe when they are in control and may shy away from technology that is "too new."

Active and vigilant users want the convenience of passive authentication but expect friction points for making informed decisions. Active factors have served as these friction points historically, but with passive authentication, both active and vigilant users wanted feedback when crossing a security boundary or authorizing a transaction that a passive authentication system was working. But this desire may decrease as the user becomes more comfortable that passive authentication is working in the background.

"Active and vigilant users want the convenience of passive authentication but expect friction points for making informed decisions. Active factors have served as these friction points historically, but with passive authentication, both active and vigilant users wanted feedback when crossing a security boundary or authorizing a transaction that a passive authentication system was working."

Relaxed users, however, are quite happy to let the system make security decisions and only get involved after something goes wrong—fully expecting someone else will shoulder the risk.

What If Something Goes Wrong?

Relaxed users are concerned that FRR will prevent efficient access to important functionality of the device. Poor responsiveness or lockout is a major concern. In such cases an active factor may be an acceptable fallback authentication method, but it must work flawlessly. Nevertheless, use of active factors is the exception and not the rule.

Active and vigilant users want to prevent exceptional cases. An active user may become concerned if she was not notified when sensitive data was to be accessed or when a financial transaction is pending. Vigilant users may take it a step further by partitioning information and resources in ways that minimize risk if security was breeched. They might want to specify how confident the system must be before allowing access.

Step-up Authentication

Step-up authentication occurs when higher-level access is needed or when a financial transaction is performed. Using active authentication factors, the user is interrupted in order to perform the authentication challenge/response. When passive factors are used with step-up authentication, authentication confidence may be satisfied transparently, removing the need to interrupt the user. However, different users expect different behavior. Active and vigilant users may want friction points that afford them tighter control.

"Conventional wisdom suggests passwords are the simplest authentication mechanism. But this conclusion ignores the fact that users must manage many passwords."

More Sensors Increase Complexity

A legitimate concern related to our approach is complexity. A tenant of security is that complexity is an enemy of security. With this we agree. However, the added complexity is necessary to improve user experience while maintaining acceptable levels of security. Conventional wisdom suggests passwords are the simplest authentication mechanism. But this conclusion ignores the fact that users must manage many passwords. Each web site, device, and service may require a different password. Passwords may be simple to implement, but adds complexity to credential lifecycle management. For example, users often experience cognitive overload when trying to remember and manage multiple passwords, which leads to reuse, storage, and sharing. Credential lifecycle management adds to the overall complexity of the solution. We have not seen complexity analysis that also considers the user experience concerns raised in this article

Conclusion

The explosion of the Internet populated by mobile connected devices and the anticipated Internet of Things suggests new approaches to authentication are needed. Active authentications including still-emerging biometric factors are insufficient in addressing user need. The proliferation and ubiquity of passive sensors such as motion, vicinity, location, and ambient resonance may be effective nontraditional authentication factors. Android supports passive authentication factors using existing OS capabilities. Minor modifications let users have a consistent authentication user experience for both local and remote services. Power vs. security tradeoffs can be efficiently implemented while taking into account user preferences. The availability of a variety of new sensors sets the stage for compelling new authentication methods that allow the cognitive load currently associated with passwords to fade into the background, being replaced by "flow of life" experiences where users participate only when required at meaningful friction points.

References

[1] Viola, Paul and Michael Jones, "Robust Real-time Object Detection," *International Journal of Computer Vision*, 2001.

[2] OpenCV, http://opencv.org.

[3] InvenSense, http://www.invensense.com/mems/gyro/mpu6050.html

[4] InvenSense, http://www.invensense.com/mems/gyro/mpu9150.html

[5] Franke, Jerry L., Jody J. Daniels, and Daniel C. McFarlane, "Recovering Context After Interruption," in *24th Annual Meeting of the Cognitive Science Society*, 2002, pp 310–315.

[6] Ho, Joyce and Stephen S. Intille, "Using Context-aware Computing to Reduce the Perceived Burden of Interruption from Mobile Devices", *CHI* 2005, April 2–7 Portland OR, USA.

[7] https://developer.android.com/about/versions/kitkat.html#44-sensors.

[8] Bhargav-Spantzel, Abhilasha, "Trusted Execution Environment for Privacy Preserving Biometric Authentication," *Intel Technical Journal* 14.1, 2014.

[9] Scurfield, Hannah, (internal user study), Intel 2014.

Author Biographies

Ned M. Smith is a principal engineer at Intel, leading security pathfinding efforts in Intel's Software and Solutions Group. During his 19-year career at Intel he has contributed to Intel® vPro™ Technology, Intel® Identity Protection Technology, and Intel® Trusted Execution Technology. He has contributed to industry technology groups including The Open Group® Common Data Security Architecture (CDSA), Trusted Computing Group® Trusted Platform Module (TPM), and the Internet Engineering Task Force (IETF). Ned holds more than 50 patents. He can be reached at ned.smith@intel.com.

Micah Sheller works as a senior software engineer and research scientist in the Intel Labs Security and Privacy Research lab. Micah joined Intel's Digital Home Group in 2005, where he worked on various media server optimizations. He then joined the Mobile Platforms Group, focusing on the USB3 specification, which names him as a technical contributor, and SSD caching technologies. In 2009, Micah joined Intel Labs to contribute to the Intel SGX™ research effort, particularly in the spaces of media use models and the Intel SGX™ software runtime. Micah's research currently focuses on new methods for user authentication. Micah has patents pending in several security spaces such as management and sharing of digital information, user authentication techniques, and user authentication policy definition/management. He can be reached at micah.j.sheller@intel.com.

Nathan Heldt-Sheller is a senior software engineer working at Intel working in the Software and Solutions Group's Software Pathfinding team. Nathan has been at Intel for 14 years, joining in 1999 as part of the Intel Web Tablet group, focusing on embedded systems power and graphics. Nathan then moved to Intel Labs' Security and Privacy Lab, with a focus on media and trusted I/O. He was part of several Trusted Execution Environment efforts, which finally culminated in the Intel SGX™ architecture, before moving to his current team. Nathan holds patents in media processing, consumer electronics, premium content protection, embedded design and test, and several security-and privacy-related technologies. He can be reached at nathan.heldt-sheller@intel.com.

SECURITY ANALYSIS OF MOBILE TWO-FACTOR AUTHENTICATION SCHEMES

Contributors

Alexandra Dmitrienko
Fraunhofer SIT

Christopher Liebchen
Technische Universität Darmstadt

Christian Rossow
Vrije Universiteit Amsterdam

Ahmad-Reza Sadeghi
Intel Collaborative Research Institute for Secure Computing

Two-factor authentication (2FA) schemes aim at strengthening the security of login-password–based authentication by deploying secondary authentication tokens. In this context, mobile 2FA schemes require no additional hardware (such as a smartcard) to store and handle the secondary authentication token, and hence are considered as a reasonable tradeoff between security, usability, and cost. They are widely used in online banking and increasingly deployed by Internet service providers.

In this article, we investigate 2FA implementations of several well-known Internet service providers such as Google, Dropbox, Twitter, and Facebook. We identify various weaknesses that allow an attacker to easily bypass 2FA, even when the secondary authentication token is not under the attacker's control. We then go a step further and present a more general attack against mobile 2FA schemes. Our attack relies on a cross-platform infection that subverts control over both end points (PC and a mobile device) involved in the authentication protocol.

We apply this attack in practice and successfully circumvent diverse schemes: SMS-based TAN solutions of four large banks, one instance of a visual TAN scheme, 2FA login verification systems of Google, Dropbox, Twitter, and Facebook accounts, and the Google Authenticator app currently used by 32 third-party service providers. Finally, we cluster and analyze hundreds of real-world malicious Android apps that target mobile 2FA schemes and show that banking Trojans already deploy mobile counterparts that steal 2FA credentials like TANs.

Introduction

"…the traditional login/password authentication is considered insufficiently secure for many security-critical applications such as online banking or logins to personal accounts."

The security and privacy threats through malware are constantly growing both in quantity and quality. In this context the traditional login/password authentication is considered insufficiently secure for many security-critical applications such as online banking or logins to personal accounts. Two-factor authentication (2FA) schemes promise a higher protection level by extending the single authentication factor, that is, *what the user knows*, with other authentication factors such as *what the user has* (for example, a hardware token or a smartphone), or *what the user is* (for example, biometrics).[29]

Even if one device/factor (such as a PC) is compromised—a typical scenario nowadays—the chance of the malware to gain control over the second device/factor (such as a mobile device) simultaneously is considered to be very low.

While biometric-based authentication is relatively expensive and raises privacy concerns, one-time passwords (OTPs) offer a promising alternative for 2FA

systems. For instance, hardware-based tokens such as OTP generators[27] are less costly but still generate additional expenses for users and are inconvenient, particularly when the user needs to carry additional hardware tokens for different organizations (for example, for accounts at several banks). On the other hand, 2FA schemes that use mobile devices (such as smartphones) have become popular recently and have been adopted by many banks and large service providers. These *mobile 2FA* schemes are considered to provide an appropriate tradeoff between security, usability, and cost, and are the focus of this article.

"These mobile 2FA schemes are considered to provide an appropriate tradeoff between security, usability, and cost…"

A prominent example of mobile 2FAs are SMS-based TAN systems (known as mTAN, smsTAN, or mobileTAN). Their goal is to mitigate account abuse even if the banking login credentials have been compromised, for example, by a PC-based banking Trojan. Here, the service provider (the bank) generates a Transaction Authentication Number (TAN), which is a transaction-dependent OTP, and sends it over SMS to the customer's phone. The user/customer needs to confirm a banking transaction by entering this TAN into the other device (typically a PC). Alternatively, visual TAN schemes encrypt and encode the TAN into a 2D barcode (visual cryptogram), which is displayed on the customer's PC from where it is photographed and decrypted by the corresponding app on the smartphone. SMS-based TAN schemes are widely deployed worldwide, also by the world's biggest banks such as Bank of America, Deutsche Bank, Santander in UK, ING in the Netherlands, and ICBC in China. Further, some large European banks have adopted visually based TAN systems recently.[7][14][15] Moreover, mobile 2FA is increasingly used by the global service providers such as Google, Twitter, and Facebook to mitigate the massive abuse of their services. Users need their login credentials and an OTP to complete the login process. The OTPs are sent to the smartphone via SMS messages or over the Internet connection. In addition, some providers offer apps that can generate OTPs on the client side, a convenient setup without the need for out-of-band communication. For instance, such an approach is followed by Google Authenticator, the popular 2FA app currently used by 32 third-party service providers.

"…mobile 2FA is increasingly used by the global service providers such as Google, Twitter, and Facebook to mitigate the massive abuse of their services."

Goal, Contributions, and Outline

The main goal of our article is to investigate and evaluate the security of various mobile 2FA schemes that are currently deployed in practice and are used by millions of customers/users.

- *Single-infection attacks on mobile 2FA schemes.* We investigate the deployed mobile 2FA of Google, Twitter, Facebook, and Dropbox service providers (see the next section, "Single-Infection Attacks on Mobile 2FA"). We point out their conceptual and implementation-specific security weaknesses and show how malware can bypass them, even when a single device, a PC, is infected. For example, some providers allow the user to deactivate 2FA without the need to verify this transaction with 2FA—an easy way for PC malware to circumvent the scheme. Other providers offer master passwords,

which as we show, can be stolen and then be used to authenticate without using an OTP. We further show how to exploit Google Authenticator, a mobile 2FA login protection app used by dozens of service providers.

- *A more general 2FA attack based on dual infections.* Then we turn our attention to more sophisticated attacks of general nature, and show that even if one of the devices (involved in a 2FA) is infected by malware, it can infect the other device with a *cross-platform infection* in realistic adversary settings (see the section "Dual-Infection Attacks on Mobile 2FA"). We demonstrate the feasibility of such attacks by prototyping PC-to-mobile cross-platform attacks. Our concept significantly enhances the well-known banking Trojans ZeuS/ZitMo[23] or SpyEye/SpitMo.[6] In contrast to these attacks that need to lure users by phishing, our technique does not require any user interaction and is completely stealthy. Once both devices are infected, the adversary can bypass various instantiations of mobile 2FA schemes, which we show by prototyping attacks against SMS-based and visual transaction authentication solutions of banks and login verification schemes of various Internet providers.
- *2FA malware in the wild.* Finally, to underline the importance to redesign mobile 2FA systems, we cluster and reverse engineer hundreds of real-world malicious apps that target mobile 2FA schemes (see the section "Real-World 2FA Attacks"). Our analysis confirms, for example, that banking Trojans already deploy mobile counterparts that allow attackers to steal 2FA credentials like TANs.

"…if one of the devices (involved in a 2FA) is infected by malware, it can infect the other device with a cross-platform infection in realistic adversary settings…"

"…banking Trojans already deploy mobile counterparts that allow attackers to steal 2FA credentials like TANs."

Single-Infection Attacks on Mobile 2FA

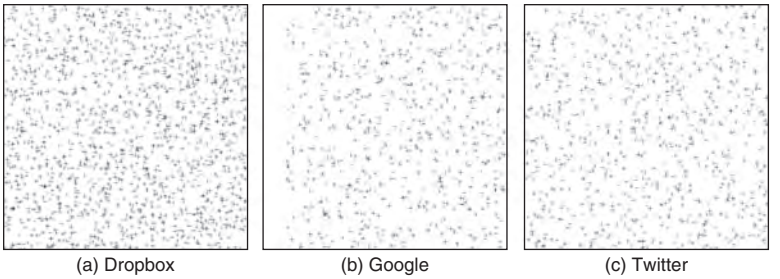
In this section, we analyze the security of mobile 2FA systems in face of compromised computers. We consider mobile 2FA schemes as secure if an adversary who compromised only a user's PC (but has no control over a mobile device) cannot authenticate in the name of the user. Such an attacker model is reasonable, as assuming a trustworthy PC would eliminate the need in utilizing a separate device to handle the secondary authentication credential.

Low-Entropy OTPs

Here we analyze the strength of OTPs generated by the four service providers under analysis. In general, low-entropy passwords are vulnerable to brute-force attacks. We thus seek to understand if the generated OTPs exhibit full basic randomness criteria. For this, we implemented a process to automatically collect OTPs from Twitter, Dropbox, and Google. We had to exclude the Facebook service from this particular test, because our test accounts were blocked after collecting only a few OTPs—presumably to keep SMS-related costs manageable.

To automate the collection process of OTPs, we implemented host software that initiates the login verification and submits the login credentials, while a mobile counterpart monitors incoming SMS messages on the mobile device and extracts OTPs into a database. The intercepted OTP is then used to

complete the authentication process at the PC. We repeat this procedure periodically. We used a collection time interval of 15 minutes for Dropbox and Twitter, but had to increase it to 30 minutes for Google to avoid our account from being blocked. In total, we collected 1564 (Dropbox), 659 (Google), and 775 (Twitter) OTPs. All investigated services create 6-digit OTPs represented in decimal format. We provide graphical representation of the collected OTPs in Figure 1.



(a) Dropbox (b) Google (c) Twitter

Figure 1: OTPs collected from three service providers. We plot a 6-digit OTP by plotting its two halves on the *x*- and *y*-axis (1000 dots wide). For example, the OTP "012763" is plotted at $x = 12$ and $y = 763$. Symbols "+" and "×" represent one and two occurrences of the same OTP, respectively.
(Source: Dmitrienko, Liebchen, Rossow, and Sadeghi, 2014)

While the OTPs generated by Dropbox and Twitter passed standard randomness tests, we observed that Google OTPs never start with a zero. Leaving out one tenth of all possible OTP values reduces the entropy of the generated passwords: the number of possible passwords is reduced by 10 percent from 10^6 to $10^6 - 10^5$.

"…we observed that Google OTPs never start with a zero."

Lack of OTP Invalidation

We made another important observation concerning invalidation of OTPs. We noticed that—if we do not complete the 2FA process—Google repeatedly created the same OTP for consecutive authentication trials. Google only invalidates OTPs (i) after an hour, or (ii) after a user successfully completed 2FA. We tested that the OTPs repeat even if the IP address, browser, and OS version of the user who wants to log in changes. An attacker could exploit this weakness to capture an OTP, while at the same time preventing the user from submitting the OTP to the service provider. This way, the captured OTP remains valid.

"We noticed that—if we do not complete the 2FA process—Google repeatedly created the same OTP for consecutive authentication trials."

The adversary can then reuse the OTP in a separate login session, because Google will still expect the same OTP—even for a different session.

Similar man-in-the-browser attacks are also possible if OTPs are invalidated, but they add a higher practical burden to the attacker.

2FA Deactivation

If 2FA is used for login verification, users can typically opt in for the 2FA feature. In the following section, we investigate how users (or attackers) can opt out from the 2FA feature. Ideally, disabling 2FA would require further security checks. Otherwise we risk that PC malware might hijack existing sessions in order to disable 2FA.

We therefore analyzed the deactivation process for the four service providers. We created one account per provider, logged in to these accounts, enabled 2FA and—to delete any session information—signed out and logged in again.

We observed that when logged in, users of Google and Facebook services can disable 2FA without any additional authentication. Twitter and Dropbox additionally require user name and password. None of the investigated service providers requested an OTP to authorize this action. Our observations imply that the 2FA schemes of the evaluated providers can be bypassed by PC malware without the need to compromise the mobile device. PC malware can wait until a user logs in, and then hijack the session and disable 2FA in the user's account settings. If additional login credentials are required to confirm this operation (as required by Twitter and Dropbox), the PC malware can reuse credentials that can be stolen, for example, by applying key logging or a man-in-the-browser attack.

"PC malware can wait until a user logs in, and then hijack the session and disable 2FA in the user's account settings."

2FA Recovery Mechanisms

While 2FA schemes promise improved security, they require users to have their mobile devices with them to authenticate. This issue may affect usability, because users may lose control over their accounts if control over their mobile device is lost (for example, if the device is lost, stolen, or temporarily unavailable due to discharged battery). To address this issue, service providers enable recovery mechanisms that allow users to retain control over their account in the absence of their mobile device. On the downside, attackers may misuse the recovery mechanism in order to gain control over user accounts without compromising the mobile device.

"…attackers may misuse the recovery mechanism in order to gain control over user accounts without compromising the mobile device."

Among the evaluated providers, Twitter does not provide any recovery mechanism. Dropbox uses a so-called recovery password, a 16-symbol-wide random string in a human-readable format, which appears in the account settings and is created when the user enables 2FA. Facebook and Google use another recovery mechanism. They offer users an option to generate a list of ten recovery OTPs, which can be used when they have no access to their mobile device. The list is stored in the account settings, similar to the recovery passwords of Dropbox. Dropbox and Google do not require any additional authentication before allowing access to this information, while Facebook additionally asks for the login credentials.

As the account settings are available to users after they have logged in, these recovery credentials (OTPs and passwords) can be accessed by malware that hijacks user sessions. For example, PC-residing malware can access this data by waiting until users sign in to their account. Hijacking the session, the malware

can then obtain the recovery passwords from the web page in the account settings—bypassing the additional check for login credentials (as in the case of Facebook).

OTP Generator Initialization Weaknesses

Schemes with client-side generated 2FA OTPs, such as Google Authenticator (GA), rely on pre-shared secrets. The distribution process of pre-shared secrets is a valuable attack vector. We analyzed the initialization process of the GA app, which is used by dozens of services including Google Mail, Facebook, and Outlook.com.

The GA initialization begins when the user enables GA-based authentication in the user's account settings. The service provider generates a QR code that is displayed to the user (on the PC) and should be scanned by the user's smartphone. The QR code contains all information necessary to initialize GA with user-specific account details and pre-shared secrets. We analyzed the QR code sent by Facebook and Google during the initialization process and identified the structure of the QR code. It includes such details as the type of the scheme (counter-based vs. time-based), service and account identifier, a counter (only for counter-based mode), the length of the generated OTP, and the shared secret. All this data is presented *in clear text*. To check if any alternative initialization scheme is supported by GA, we reverse engineered the app with the JEB Decompiler and analyzed the app internals. We did not identify any alternative initialization routines, which indicates that all 32 service providers using GA use this initialization procedure.

Unfortunately, PC-residing malware can intercept the initialization message (clear text encoded as a QR code). The attacker can then initialize the attacker's own version of the GA and can generate valid OTPs for the target account.

"…PC-residing malware can intercept the initialization message (clear text encoded as a QR code). The attacker can then initialize the attacker's own version of the GA and can generate valid OTPs…"

"…we apply cross-platform infection attacks (PC-to-mobile) in context of mobile 2FA schemes."

Dual-Infection Attacks on Mobile 2FA

In this section, we present a more general attack against mobile 2FA schemes. Particularly, we present the attack model that does not rely on implementation weaknesses (as, for example, weaknesses reported in the previous section), but rather conceptual. Particularly, we apply cross-platform infection attacks (PC-to-mobile) in context of mobile 2FA schemes. Our attack model undermines the security of a large class of 2FA schemes that are widely used in security-critical applications such as online banking and login verification.

System Model

Our system model is depicted in Figure 2. It includes the following actors: (i) a user *U*, (ii) a web server *S*, (iii) a computer *C*, (iv) a mobile device *M*, and (v) a remote malicious server *A*. The user *U* is a customer who is subscribed for the online service. The web server *S* is maintained by the service provider of the online service. The computer *C* is either a desktop PC or a laptop used by the user to access the web site hosted by *S*. The mobile device *M* is a handheld computer or a smartphone of *U,* which is involved in authentication of *U* against *S*.

The legitimate communication between the entities is illustrated with dashed arrows in Figure 2. To get access to the service, U has to prove to S possession of both authentication tokens T1 and T2. The first authentication token T1 is handled by C (typically represented by login credentials). The second authentication token T2 is handled by the mobile device M. T2 is an OTP which is either received from S via an out-of-band channel, or generated locally on M.

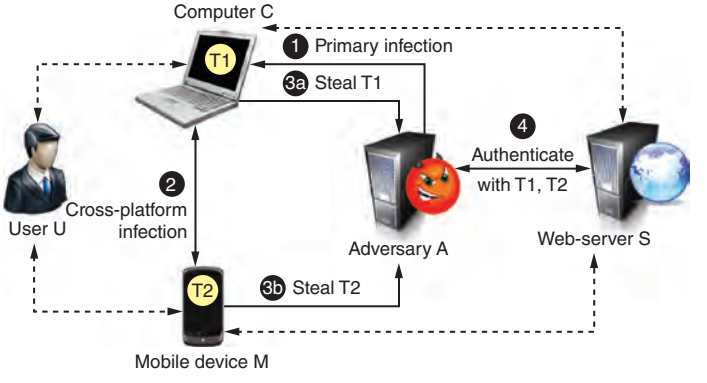


Figure 2: System model and attack steps
(Source: Dmitrienko, Liebchen, Rossow, and Sadeghi, 2014)

A remote malicious server A represents an adversary who aims to gain control over C and M and to steal authentication tokens T1 and T2 in order to be able to successfully authenticate against S in the name of U.

Assumptions

"…nowadays many PCs are infected."

We assume that *C*, the user's PC, is compromised. This assumption is reasonable, because nowadays many PCs are infected. We further assume that the second device, either M or C, suffers from a (memory-related) vulnerability that allows the attacker to subvert the control over the code execution. The probability for such vulnerabilities is quite high for both mobile and desktop operating systems. As a reference, the National Vulnerability Database[1] lists more than 55,000 discovered information security vulnerabilities and exposures for mainstream platforms. Despite decades of history, these vulnerabilities are a prevalent attack vector and pose a significant threat to modern systems.[31]

"Despite decades of history, these vulnerabilities are a prevalent attack vector and pose a significant threat to modern systems."

Attack Description

The general attack scenario has four phases, which are illustrated by solid lines in Figure 2: (i) primary infection; (ii) cross-platform infection; (iii) stealing authentication tokens, and (iv) authentication.

1. *Primary infection*. We do not specify the way the attacker achieves a primary infection. Instead, we assume that C is already infected (see the previous section, "Assumptions").

2. *Cross-platform infection.* The infected C attempts to compromise M by triggering a memory-related vulnerability. Exploitation is possible if, for example, both devices are connected to a single network, as described in the following section, "Cross-Platform Infection in LAN/WLAN Networks."
3. *Stealing authentication tokens.* As we will show, when controlling M and C, an attacker A can obtain both authentication tokens T1 and T2 (steps 3a and 3b respectively). Static authentication tokens that do not change from one to another authentication session (such as login credentials) are immediately transmitted to and persistently stored at A.
4. *Authentication.* Authentication is performed by A, who controls both authentication tokens. A has a local copy of static authentication tokens (such as login credentials), and can obtain OTPs by forwarding them from M to A. Note that A does not only hijack the session of U, but can even establish the attacker's own sessions at any time and independently from U.

Cross-Platform Infection in LAN/WLAN Networks

LAN/WLAN networks are often used at home, at work, or in public places, such as hotels, cafes, or airports. Users often connect both their PCs and mobile devices to the same network (for example, in home networks). To perform cross-platform infections in the LAN/WLAN network, the malicious PC becomes a man in the middle (MITM) between the mobile device and the Internet gateway in order to infect it via malicious payloads. To become an MITM, techniques such as ARP cache poisoning[5] or a rogue DHCP server[18] can be used. Next, the MITM supplies an exploit to the victim, which results in code injection and remote code execution.

"Users often connect both their PCs and mobile devices to the same network…"

For our implementation of cross-platform infection, we used a rogue DHCP server attack to become an MITM. In particular, C advertises itself as a network gateway and becomes an MITM when its malicious DHCP configuration is accepted by M. As the MITM, C can manipulate Internet traffic supplied to M. When M connects to the network and requests an IP address, this request is served by our malicious DHCP server, which assigns a valid configuration for this network, but substitutes the correct gateway IP address with its own. The malware loads a driver that implements network address translation (NAT) to dynamically forward any HTTP request to an external or local HTTP server. This server answers every HTTP request with a malicious web page.

When U opens the browser in M and navigates to any web page, the request is forwarded to C due to the network configuration of M specifying C as a gateway. The malicious C does not provide the requested page, but supplies a malicious page containing an exploit triggering the vulnerability in the web browser. In our prototype we used a use-after-free vulnerability CVE-2010-1759 in WebKit, the web engine of the Android browser. We further perform a privilege escalation to root by triggering the vulnerability CVE-2011-1823 in the privileged Android's volume manager daemon process.

"The malicious C does not provide the requested page, but supplies a malicious page containing an exploit triggering the vulnerability in the web browser."

"…we prototyped attacks against SMS-based TAN schemes of several banks, bypassed 2FA login verification systems of popular Internet service providers, defeated the visual TAN authentication scheme of Cronto, and circumvented Google Authenticator."

Bypassing Different Instantiations of Mobile 2FA Schemes

Next we present instantiations of dual-infection attacks against a wide range of mobile 2FA schemes. Particularly, we prototyped attacks against SMS-based TAN schemes of several banks, bypassed 2FA login verification systems of popular Internet service providers, defeated the visual TAN authentication scheme of Cronto, and circumvented Google Authenticator. Overall, our prototypes demonstrate successful attacks against mobile 2FA solutions of different classes.

Bypassing SMS-based TAN Schemes and 2FA Login Verification Schemes

To bypass SMS-based TAN schemes used by banks and 2FA login verification systems, we launched a man-in-the-browser attack on the PC to steal the login credentials (that is, PIN or password) from the computer before they are transferred to the web server of the bank or the service provider. Further, we implemented mobile malware that obtains the secondary credential, an OTP or TAN, by intercepting SMS messages on the mobile device. It acts as a man-in-the-middle between the GSM modem and the telephony stack of Android and intercepts all SMS messages of interest (so that the user does not receive them), while it forwards all other SMS messages for "normal" use.

"We successfully evaluated our prototype on online banking deployments of four large international banks…"

We successfully evaluated our prototype on online banking deployments of four large international banks (the names of the banks are kept undisclosed) and evaluated it against the 2FA login verification systems of Dropbox, Facebook, Google, and Twitter.

Bypassing Visual TAN Solutions

To demonstrate the effectiveness of dual-infection attacks against visual TAN solutions, we successfully crafted such an attack against the demo version of the Cronto visual transaction signing solution—the CrontoSign app (v. 5.0.3). We reused the man-in-the-browser attack to leak login credentials from the PC and used our mobile malware to steal key material stored by the CrontoSign app. We then copied stolen files with key material onto another (adversarial) phone with CrontoSign installed and then performed a login attempt with stolen login credentials and the adversarial phone. The app on the adversarial phone produced correct OTP, which was then used to successfully complete authentication.

Bypassing Google Authenticator (GA) App

We selected Google Authenticator (GA) as our attack target due to its wide deployment. As of October 2013, it was being used by 32 service providers, among them Google, Microsoft, Facebook, Amazon, and Dropbox. The GA app does not receive OTP from the server, but instead generates it on client side. The generation algorithm is seeded with a secret that is shared between the server and the mobile client and further requires a pseudo-random input like a nonce to randomize the output value of each run. GA supports the following nonce values: shared time (in a form of the time epoch) or a counter with a shared state. In either case, it stores all security-sensitive parameters

(such as the seed and a nonce) for the OTP generation in an application-specific database. Hence, to bypass the scheme, our PC-based malware steals login authentication credentials, while our mobile malware leaks the database file stored in the GA application directory. We copied the database on another mobile device with an installed GA app and were able to generate the same OTPs as the victim.

Real-World 2FA Attacks

Until now, we have drafted attacks that enable attackers to circumvent mobile 2FA systems in a completely automated way. In this section, we analyze real-world malware in order to shed light onto how attackers already bypass 2FA schemes in the wild.

"…we analyze real-world malware in order to shed light onto how attackers already bypass 2FA schemes in the wild."

Dataset

Our real-world malware analysis is based on a diverse set of Android malware samples obtained from different sources.

We analyzed malware from the Malgenome[33] and Contagiodump[34] projects. In addition, we obtained a random set of malicious Android files from VirusTotal. Note that we aimed to analyze malware that attacks 2FA schemes. We thus filtered on malware that steals SMS messages, that is, malware that has the permission to read incoming SMS messages. In addition, we only analyzed apps that were labeled as malicious by at least five antivirus vendors. Our resulting dataset consists of 207 unique malware samples.

Malware Analysis Process

We used a multistep analysis of Android malware samples, as depicted in Figure 3. First, we dynamically analyzed the malware in an emulated Android environment. Dynamic analysis helped us to focus on the malware's behavior when an SMS message is received. Second, to speed up manual static analysis, we clustered the analysis reports to group similar instances. Third, we manually reverse engineered malware samples from each cluster to identify malicious behavior.

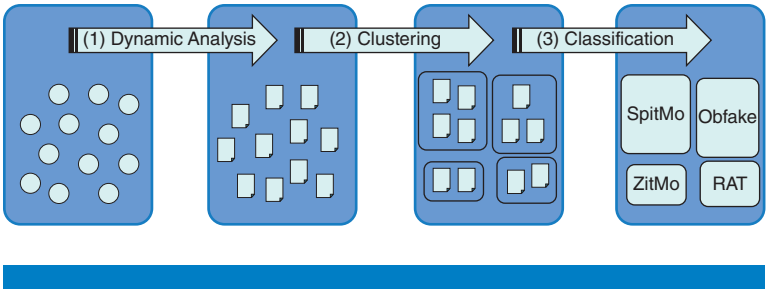


Figure 3: Multistep malware analysis procedure
(Source: Dmitrienko, Liebchen, Rossow, and Sadeghi, 2014)

Dynamic Malware Analysis

We dynamically analyzed the malware samples by running each APK file in an emulated Android environment. In particular, we modified the Dalvik Virtual Machine of an Android 2.3.4 system to log method calls (including parameters and return values) within an executed process.

We aimed to observe malicious behavior when SMS messages were received, that is, we were not interested in the overall behavior of an app. We therefore triggered this behavior by simulating incoming SMS messages while the malware was executed. To filter on the relevant behavior, the analysis reports contain only the method calls that followed the SMS injection. This way, we highlight code that is responsible for sniffing and stealing SMS messages, while we ignore irrelevant code parts (such as third-party libraries).

Likewise, in the case the malware bundles benign code (such as a repacked benign app), our analysis report does not contain potentially benign code parts. We stopped the dynamic analysis 60 seconds after we injected the SMS message.

The analysis reports consist of tuples with the format:

rline = <cls, method, (p[1], …, p[x]), rval>,

whereas *cls* represents the class name, *method* is the method name, *rval* is the return type/value tuple, and *p*[*i*] is a list of parameter type/value tuples; *rline* is one line in the report.

Report Clustering

We then used hierarchical clustering to group similar reports in order to speed up the manual reverse engineering process. Intuitively, we wanted to group samples into a cluster if they had a similar behavior when intercepting an SMS message.

We defined the similarity between to samples as the normalized Jaccard similarity between two reports A and B:

$sim(A, B) = |A \cap B| / |A \cup B|$,

whereas the reports A and B are interpreted as sets of (unordered) report lines. Two report lines are considered equal if the class name, method name, number and type of parameters and return types are equal.

We calculated the distances between all malware samples and grouped them to one cluster if the distance $d = 1 - sim(A, B)$ is lower than a cutoff threshold of 40 percent. In other words, two samples were clustered together if they shared at least 40 percent of the method calls when receiving an SMS message.

Classification

Given the lack of ground truth for malware labels, we chose to manually assign labels to the resulting clusters. We use off-the-shelf Java bytecode decompilers such as JD-GUI or Androguard to manually reverse engineer each three samples of the 10 largest clusters to classify the malware into families.

Analysis Results

This section shows the clustering results and gives a detailed analysis of one of the analyzed ZitMo samples.

Clustering Results

Clustering of the 207 samples finished in 3 seconds and revealed 21 malware clusters and 45 singletons.

"Clustering of the 207 samples finished in 3 seconds and revealed 21 malware clusters and 45 singletons."

We now describe the most prominent malware clusters. Table 1 details full clustering and classification results.

Family	Command & Control	Leaked TAN via	# Samples
AndroRAT	TCP	TCP	16
ZitMo.A	SMS	HTTP (GET)	13
SpitMo.A	SMS	SMS	13
Obfake.A	n/a	SMS	12
SpitMo.C	HTTP	HTTP (GET)	6
RusSteal	n/a	SMS	6
Koomer	n/a	SMS	5
Obfake.B	n/a	SMS	4
SpitMo.B	n/a	HTTP (POST)	3
CitMo.A	n/a	HTTP (GET)	3

Table 1: Real-world malware families targeting 2FA by stealing SMS messages
(Source: Dmitrienko, Liebchen, Rossow, and Sadeghi, 2014)

AndroRAT, a (formerly open-source) remote administration tool for Android devices, forms the largest cluster in our analysis with 16 unique malware samples. Attackers use the flexibility of AndroRAT to create custom SMS-stealing apps, for example, in order to adapt the command and control (C&C) network protocol or data leakage channels.

Next to AndroRAT, the app counterparts of the banking Trojans (ZitMo for ZeuS, SpitMo for SpyEye, CitMo for Citadel) are also present in our dataset. Except SpitMo.A, these samples leak the contents of SMS messages via HTTP to the botmaster of the banking Trojans. Two SpitMo variants have a C&C channel that allowed the configuration of the C&C server address or Dropzone phone number, respectively.

We further identified four malicious families that forward SMS messages to a hard-coded phone number. We labeled a cluster *RusSteal*, as the malware samples specifically intercept TAN messages with Russian contents. Except RusSteal, none of the families includes code that is related to specific banking Trojans. Instead, the apps blindly steal all SMS messages, usually without further filtering, and hide the messages from the smartphone user. The apps could thus be coupled interchangeably with any PC-based banking Trojan.

"…the apps blindly steal all SMS messages, usually without further filtering, and hide the messages from the smartphone user."

"…malware has already started to target mobile 2FA, especially in the case of SMS-based TAN schemes for banks."

Our analysis shows that malware has already started to target mobile 2FA, especially in the case of SMS-based TAN schemes for banks. We highlight that we face a global problem, and next to the Russian-specific Trojans that we found, incidents in many other countries worldwide have been reported.[11][12][19] The emergence of toolkits such as AndroRAT will ease the development of malware targeting specific 2FA schemes.

Until now, these families have largely relied on manual user installation, but as we have shown, automated cross-platform infections are possible. This motivates further research to foster more secure mobile 2FA schemes.

ZitMo Case Study

We now outline the reverse engineering results of one of the samples to show the inner workings of real-world malware in more detail. Here we provide a case study on the ZitMo malware samples (we analyzed the ZitMo sample with a SHA256 value of ceb54cba2561f62259204c39a31dc204105d358a1a10cee37de889332fe6aa27), which are the mobile counterparts of the ZeuS banking Trojan.

In order to install ZitMo, the ZeuS Trojan manipulates an online banking session such that ZeuS-infected users are asked to enter their mobile phone number. Once they do so, the attackers send an SMS message with a link to *security software*, which in fact is a camouflaged ZitMo Trojan. In contrast to the attack that we have described, the infection of the mobile device is a largely manual process and requires user interaction.

Once ZitMo is installed, it asks the user to enter a verification code, which the attackers use to establish a unique mapping between the infected PC and the mobile counterpart. From this point on, ZitMo operates in background. As ZitMo has registered as a broadcast receiver for SMS messages, it can intercept, manipulate, and read all incoming SMS messages.

Whenever an SMS message is received, ZitMo first checks if it contains a hidden command that can be used to reconfigure ZitMo. Such messages remain hidden to the user: they are not visible in the default messaging app. ZitMo embeds the content of all other messages as HTTP request parameters and sends the data (including the device ID) to the ZitMo dropzone server. Before the data can be forwarded, ZitMo needs to reverse its obfuscation of the dropzone URL. If the HTTP request fails, ZitMo stores the message in hidden data storage and retries submission at a later stage.

"…ZitMo or similar mobile malware have been observed to steal tens of millions of dollars from infected users."

Although using a simple scheme, ZitMo or similar mobile malware have been observed to steal tens of millions of dollars from infected users.[19]

Countermeasures and Tradeoffs

Possible defense strategies against attacks on mobile 2FA schemes can be divided into two classes: preventive and reactive countermeasures. Preventive countermeasures, such as exploitation mitigation, OS-level security extensions,

leveraging secure hardware, and using trusted VPN proxies, are applied in order to reduce the attack surface, while reactive countermeasures aim to detect ongoing attacks in order to mitigate further damage.

Exploitation Mitigation

Our cross-device infection attack relies on exploitation of memory-related vulnerability (see the earlier section, "Assumptions"), hence, mitigation techniques against runtime exploitations would be an effective countermeasure. However, despite more than two decades of research, such flaws still undermine security of modern computing platforms.[31] Particularly, while the Write-XOR-Execute (W^X)[37] security model prevents code injection (enforced on Android since 2.3 version), it can be bypassed by code reuse attacks, such as ret2libc[38] or return-oriented programming (ROP).[39] Code reuse attacks do not require code injection, but rather invoke execution of functions or sequences of instructions that already reside in the memory. Because code reuse attacks make use of memory addresses to locate instruction sequences to be executed during the attack, the mitigation techniques were developed that randomize program memory space, making it hard to predict exact addresses prior to program execution. For instance, address space layout randomization (ASLR)[40], which adds a random offset to loaded code at each program start, is available on iOS starting from version 4.3 and was also recently introduced for Android (in 4.0 version).

However, ASLR can be bypassed by brute-forcing the offset at runtime[41], which generated a new line of research on fine-grained address space randomization[42][43][44][45][46][47] (down to instruction level), which makes brute-force attacks infeasible. Unfortunately, fine-grained address space randomization techniques are ineffective in the presence of memory disclosure bugs. Particularly, these bugs can be utilized to disclose memory content and build a return-oriented programming (ROP) payload dynamically at runtime.[48,49]

Hence, while the deployed memory mitigation techniques raise the bar for the type of cross-device infection we demonstrated, such attacks are still possible, even if all protections are enforced.

"…while the deployed memory mitigation techniques raise the bar for the type of cross-device infection we demonstrated, such attacks are still possible…"

OS Level Security Extensions

OS security improves over time and can mitigate some attack classes. With respect to the threat of mobile malware targeting 2FA, the first significant changes appeared in version 4.2 of Android, where a new system API was introduced allowing users to verify and to selectively grant or deny permissions requested by applications during application installation. Ideally, the users can choose during the installation process what privileges a (potentially malicious) app should get, which could defeat some user-installed malware instances (see the earlier section "Real-World 2FA Attacks").

"OS security improves over time and can mitigate some attack classes."

Moreover, Android introduced SELinux[50] in version 4.3—a security framework that allows more fine-grained access control to system resources. This countermeasure makes it more difficult to perform privilege escalation

(also used in our exploits). Further, version 4.3 also introduced authentication for the Android Debug Bridge (adb), which can prevent cross-device infections via USB connections.

The most recent Android version 4.4 provides an enhanced message handling, which prevents third-party applications from silently receiving or sending any SMS. While malware like ZitMo/SpitMo is still able to relay received TAN messages, they will remain visible in the phone's default messaging application, giving the user the chance for an immediate reaction, such as, for example, to call the bank and cancel the transaction. However, this countermeasure will have no effect on our attacks, since we operate at a lower level of the software stack, meaning that the application framework itself will never receive a suppressed message. It is therefore likely that future attacks will follow our concept.

Leveraging Secure Hardware on Mobile Platforms

"A more flexible alternative to dedicated hardware tokens is utilizing general purpose secure hardware available on mobile devices for OTP protection."

A more flexible alternative to dedicated hardware tokens is utilizing general purpose secure hardware available on mobile devices for OTP protection. For instance, ARM processors feature the ARM TrustZone technology[51] and Texas Instruments processors have the M-Shield security Extensions.[52] Further, platforms may include embedded Secure Elements (SE) (available, for example, on NFC-enabled devices) or support removable SEs (such as secure memory cards[53] attached to a microSD slot). Finally, SIM cards available on most mobile platforms include a secure element. Such secure hardware allows establishment of a trusted execution environment (TEE) on the device, which can be used to run security-sensitive code to handle authentication secrets in isolation from the rest of the system. Developments in this direction are solutions for mobile payments like Google Wallet[54] and PayPass.[55]

With the release of version 4.3, Android started to support hardware-supported trusted key storage. This means that keys can now be saved in an SE or TEE. However, this is not sufficient to prevent attacks on 2FA schemes, because the keys can be retrieved from the trusted storage by the application that created them. Hence, the adversary could compromise the target application, which has the privileges to query the keys. Even if the OTP generation would take place within the TEE, an attacker could still impersonate the target application in one way or another.

"…the only way to build a secure 2FA on top of TEE is to shift the entire verification process into the TEE."

We believe the only way to build a secure 2FA on top of TEE is to shift the entire verification process into the TEE. We envision the following workflow, which was also described by Rijswijk-Deij[37]: An OTP/TAN application is securely deployed into the TEE. On the first start, this application would establish a secure connection to the service provider/bank (based on public key certificate of the service provider) and prove that it is executed in a legitimate TEE via remote attestation. Next, the application would generate a public/private key pair and send the public key to the service provider/bank. To begin a transaction, the user would start the application. It would then query the service provider/bank for any

transaction, which would need to be authorized. If such an action existed, it would be authenticated using the public key of the service provider/bank and displayed to the user, via a trusted user interface. The user would then either allow or deny this action via trusted input. The user's decision would be signed using the generated private key and could be verified by the service provider/bank.

A crucial requirement to underlying TEE in such a use case is trusted user in-/output, which allows the user to enter security sensitive data (such as transaction confirmation) directly into TEE. When such input is mediated by the OS, it can be manipulated by malware so that a program executed within TEE will confirm a transaction or login attempt without user consciousness. However, although some TEEs such as TrustZone can provide trusted user in-/output, in current implementations this feature is not supported. Hence, solutions built on top of existing TEEs still rely on trusted OS components to handle user input.

"…solutions built on top of existing TEEs still rely on trusted OS components to handle user input."

Moreover, most available TEEs are not open to third-party developers. For instance, secure elements available on SIM cards are controlled by network operators, while processor-based TEEs such as ARM TrustZone and M-Shield are controlled by phone manufacturers. Typically, only larger companies such as Google, Visa, and MasterCard can afford cooperation with phone manufacturers, while smaller service providers remain with an alternative to cooperate with network operators or use freely programmable TEEs such as secure memory cards. However, the solution utilizing SIM-based secure elements would be limited to customers of a particular network operator, while secure memory cards can be used only with devices featuring a microSD slot.

"…most available TEEs are not open to third-party developers."

Trusted VPN Proxy

Cross-platform infection attacks as discussed earlier can be defeated by deploying standard countermeasures against MITM attacks. For example, one could enforce HTTPS for every web page request or tunnel HTTP over a remote trusted virtual private network (VPN). However, the former solution would require changes on all Internet servers currently providing HTTP connections (which is infeasible), while the latter would impact performance (as in the case where a single VPN proxy serves several clients). Moreover, it is not clear which party is trustworthy to host such a proxy.

Detection of Suspicious Mobile Apps

SMS-stealing apps exhibit suspicious characteristics or behavior that can be detected by defenders. For example, using static analysis, it is possible to classify suspicious sets of permissions or to identify receivers for events of incoming SMS messages.[56] Similarly, taint tracking helps to detect information leakage.[57] However, tainting requires kernel modifications that are impractical on normal user smartphones and implicit flows can evade taint analysis.[58] An alternative are user space security apps that detect suspicious

behavior of the malicious CitMo/SpitMo/ZitMo apps. Such a security app could, for instance, identify SMS receivers that consume or forward TAN-related SMS by observing the receivers' behavior. Further, by knowing the command and control (C&C) channels of mobile malware, one could identify (and block) data leakage in network traffic.

"…our proposed attack cannot be detected in user space, as we show that we can steal OTPs before any app running in user space has noticed events such as an incoming SMS message."

However, these security measures require prior knowledge of the attacks and C&C obfuscation evades such defenses. Further, our proposed attack cannot be detected in user space, as we show that we can steal OTPs before any app running in user space has noticed events such as an incoming SMS message. Consequently, the aforementioned solutions are not suitable to counter our attack, and instead can only detect the existing SMS-stealing Trojans.

Attack Detection in the Network

Our cross-platform infection attack scenario can be detected or even prevented at the network layer.

"…regular Wi-Fi routers for private use remain unprotected."

Particularly, mitigation techniques exist against rogue DHCP attacks, such as DHCP snooping.[59] For example, the router could stop routing Internet traffic if it detects rogue DHCP servers. However, these mechanisms are available on advanced multilayer switches only and require configuration efforts by network administrators[60], while regular Wi-Fi routers for private use remain unprotected. We did not encounter any home router that uses such countermeasures. Further, these measures are specific to cross-platform infection attacks that rely on rogue DHCP, while ineffective against other scenarios, such as those, for example, based on tethering.

Related Work

In this section we survey previous research on mobile 2FA schemes, on attacks against SMS-based TAN systems, and on cross-platform infections.

Mobile 2FA Schemes

Balfanz et al.[10] aim to prevent misuse of the smartcard plugged into the computer by malware without user knowledge. They propose replacing the smartcard with a trusted handheld device that asks the user for permission before performing sensitive operations. Aloul et al.[8,9] utilize a trusted mobile device as an OTP generator or as a means to establish OOB communication channel to the bank (via SMS). Mannan et al.[20] propose an authentication scheme that is tolerant against session hijacking, keylogging, and phishing. Their scheme relies on a trusted mobile device to perform security-sensitive computations. Starnberger et al.[28] propose an authentication technique called QR-TAN that belongs to the class of visual TAN solutions. It requires the user to confirm transactions with the trusted mobile device using visual QR barcodes. Clarke et al.[13] propose to use a trusted mobile device with a camera and OCR as a communication channel to the mobile. The Phoolproof phishing prevention solution[24] utilizes a trusted user cell phone in order to generate an additional token for online banking authentication.

All these solutions assume that the user's personal mobile device is trustworthy. However, as we showed in this article, an attacker controlling the user's PC can also infiltrate that user's mobile device by mounting a cross-platform infection attack, which undermines the assumption on trustworthiness of the mobile phone.

"…an attacker controlling the user's PC can also infiltrate that user's mobile device by mounting a cross-platform infection attack,…"

Attacks on SMS-based TAN Authentication

Mulliner et al.[21] analyze attacks on OTPs sent via SMS and describe how smartphone Trojans can intercept SMS-based TANs. They also describe countermeasures against their attack, such as dedicated OTP channels that cannot be easily intercepted by normal apps. Their attack and countermeasure rely on the assumption that an attacker has no root privileges, which we argue is not sufficiently secure in the adversary setting nowadays.

Schartner et al.[26] present an attack against SMS-based TAN solutions for the case when a single device, the user's mobile phone, is used for online banking. The presented attack scenario is relatively straightforward as the assumption of using a single device eliminates challenges such as cross-platform infection or a mapping of devices to a single user. Many banks already acknowledge this vulnerability and disable TAN-based authentication for customers who use banking apps.

Cross-Platform Infection

The first malware spreading from smartphone to PC was discovered in 2005 and targeted Symbian OS.[2] Infection occurred as soon as the phone's memory card was plugged into the computer. Another example of cross-platform infection from PC to the mobile phone was proof-of-concept malware that had been anonymously sent to the Mobile Antivirus Research Association in 2006.[17][25] The virus affected the Windows desktop and Windows Mobile operating systems and spread as soon as it detected a connection using Microsoft's ActiveSync synchronization software. Another well-known cross-platform infection attack is a sophisticated worm Stuxnet[22], which spreads via USB keys and targets industrial software and equipment. Further, Wang et al.[32] investigated phone-to-computer and computer-to-phone attacks over USB targeting Android. They report that a sophisticated adversary is able to exploit the unprotected physical USB connection between devices in both directions. However, their attack relies on additional assumptions, such as modifications in the kernel to enable non-default USB drivers on the device, and non-default options to be set by the user.

Up to now, most cross-system attacks were observed in public networks, such as malicious Wi-Fi access points[4] or ad-hoc peers advertising free public Wi-Fi.[3] When a victim connects to such a network, it gets infected and may start advertising itself as a free public Wi-Fi to spread. In contrast to our scenario, this attack mostly affects Wi-Fi networks in public areas and targets devices of other users rather than a second device of the same user. Moreover, it requires user interaction to join the discovered Wi-Fi network. Finally, the infection does not spread across platforms (from PC to mobile or vice versa), but rather affects similar systems.

"…most cross-system attacks were observed in public networks, such as malicious Wi-Fi access points or ad-hoc peers advertising free public Wi-Fi."

Conclusion

"We thus see a need for research on more secure mobile 2FA schemes that can withstand today's sophisticated adversary models."

"As follow-up research, we propose to explore authentication mechanisms that use secure hardware on mobile platforms."

In this article, we studied the security of mobile two-factor authentication (2FA) schemes that have received much attention recently and are deployed in security-sensitive applications such as online banking and login verification.

Our results show that current mobile 2FA schemes have conceptual weaknesses, because adversaries can intercept OTPs or steal private key material for OTP generation. We thus see a need for research on more secure mobile 2FA schemes that can withstand today's sophisticated adversary models.

As follow-up research, we propose to explore authentication mechanisms that use secure hardware on mobile platforms. Although current secure hardware has its limitations (for example, no support for a secure user interface, or not being freely programmable), novel approaches based on secure hardware could eliminate the inherent weaknesses of existing authentication schemes.

References

[1] National vulnerability database version 2.2. http://nvd.nist.gov/.

[2] Kawamoto, Dawn, "Cell phone virus tries leaping to PCs," CNET, http://news.cnet.com/Cell-phone-virus-tries-leaping-to-PCs/2100-7349_3-5876664.html?tag=mncol;txt, 2005.

[3] Phifer, Lisa, "The security risks of 'Free Public WiFi,'" TechTarget, http://searchsecurity.techtarget.com.au/news/2240020802/The-security-risks-of-Free-Public-WiFi, 2009.

[4] Tobadmin, "KARMA demo on the CBS early show," http://blog.trailofbits.com/2010/07/21/karma-demo-on-the-cbs-early-show/, 2010.

[5] Nachreiner, Corey, "Anatomy of an ARP poisoning attack," WatchGuard, http://www.watchguard.com/infocenter/editorial/135324.asp, 2011.

[6] Liebowitz, Matt, "New Spitmo banking Trojan attacks Android users," TechNews Daily, http://www.securitynewsdaily.com/1048-spitmo-banking-trojan-attacks-android-users.html, 2011.

[7] Raiffeisen PhotoTAN. http://www.raiffeisen.ch/web/phototan, 2012.

[8] Aloul, F., S. Zahidi, and W. El-Hajj, "Two factor authentication using mobile phones," in *IEEE/ACS Computer Systems and Applications*, May 2009.

[9] Aloul, F., S. Zahidi, and W. ElHajj, "Multifactor authentication using mobile phones," *International Journal of Mathematics and Computer Science*, 4, 2009.

[10] Balfanz, D. and E. W. Felten, "Hand-held computers can be better smart cards," *USENIX Security Symposium* - Volume 8, USENIX Association, 1999.

[11] Castillo, Carlos, McAfee blog entry: "Android banking Trojans target Italy and Thailand," http://blogs.mcafee.com/mcafee-labs/android-banking-trojans-target-italy-and-thailand/, 2013.

[12] Castillo, Carlos, McAfee blog entry: "Phishing attack replaces Android banking apps with malware," http://blogs.mcafee.com/mcafee-labs/phishing-attack-replaces-android-banking-apps-with-malware, 2013.

[13] Clarke, D., B. Gassend, T. Kotwal, M. Burnside, M. v. Dijk, S. Devadas, and R. Rivest, "The untrusted computer problem and camera-based authentication," in *International Conference on Pervasive Computing*, Springer-Verlag, 2002.

[14] Cronto Limited, "Commerzbank and Cronto launch secure online banking with photoTAN—World's first deployment of visual transaction signing mobile solution," http://www.cronto.com/download/Cronto_Commerzbank_photoTAN.pdf, 2008.

[15] Cronto Limited, "CorpBanca and Cronto secure online banking transactions with CrontoSign, http://www.cronto.com/corpbanca-cronto-secure-online-banking-transactions-crontosign.htm, 2011.

[16] Malik, Amit, "DLL injection and hooking," SecurityXploded, http://securityxploded.com/dll-injection-and-hooking.php

[17] Evers, J., "Virus makes leap from PC to PDA," CNET, http://news.cnet.com/2100-1029_3-6044457.html, 2006.

[18] Jerschow, Y. I., C. Lochert, B. Scheuermann, and M. Mauve, „CLL: A cryptographic link layer for local area networks," in *International conference on Security and Cryptography for Networks*, Springer-Verlag, 2008.

[19] Kalige, E. and D. Burkey, "Eurograbber: How 36 million euros was stolen via malware," http://www.cs.stevens.edu/~spock/Eurograbber_White_Paper.pdf.

[20] Mannan, M. and P. C. Van Oorschot, "Using a personal device to strengthen password authentication from an untrusted computer," in *FC'07/USEC'07*, 2007.

[21] Mulliner, C., R. Borgaonkar, P. Stewin, and J.-P. Seifert, "SMS-based one-time passwords: Attacks and defense" (short paper), in *DIMVA*, July 2013.

[22] Falliere, N., "Exploring Stuxnet's PLC infection process," Symantec blog entry, http://www.symantec.com/connect/blogs/exploring-stuxnet-s-plc-infection-process, 2010.

[23] V. News, "Teamwork: How the ZitMo Trojan bypasses online banking security," Kaspersky Lab, http://www.kaspersky.com/about/news/virus/2011/Teamwork_How_the_ZitMo_Trojan_Bypasses_Online_Banking_Security, 2011.

[24] Parno, B., C. Kuo, and A. Perrig, "Phoolproof phishing prevention, in *Financial Cryptography and Data Security*, Springer-Verlag, 2006.

[25] Peikari, C., "Analyzing the crossover virus: The first PC to Windows handheld cross-infector," *InformIT,* http://www.informit.com/articles/article.aspx?p=458169, 2006.

[26] Schartner, P. and S. Bürger, "Attacking mTAN-applications like e-banking and mobile signatures," Technical report, University of Klagenfurt, 2011.

[27] Sparkasse Pfullendorf-Meßkirch, "Online banking mit chipTAN," https://www.sparkasse-pm.de/privatkunden/banking/chiptan/vorteile/index.php?n=/privatkunden/banking/chiptan/vorteile/.

[28] Starnberger, G., L. Froihofer, and K. Goeschka. "QR-TAN: Secure mobile transaction authentication," in *International Conference on Availability, Reliability and Security*, IEEE, 2009.

[29] Tanenbaum, A., "*Modern Operating Systems*", 3rd edition, Prentice Hall Press, Upper Saddle River, NJ, USA, 2007.

[30] TrendLabs, "3Q 2012 security roundup. Android under siege: Popularity comes at a price," http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/reports/rpt-3q-2012-security-roundup-android-under-siege-popularity-comes-at-a-price.pdf, 2012.

[31] van der Veen, V., N. dutt Sharma, L. Cavallaro, and H. Bos, "Memory errors: The past, the present, and the future," in *Recent Advances in Intrusion Detection Symposium*, 2012.

[32] Wang, Z. and A. Stavrou, "Exploiting smart-phone USB connectivity for fun and profit," in *26th Annual Computer Security Applications Conference*, ACM, 2010.

[33] Zhou, Y. and X. Jiang, "Dissecting Android malware: Characterization and evolution, in *IEEE Symposium on Security and Privacy*, 2012.

[34] "Mobile Malware Mini Dump," Contagio Mobile, http://contagiominidump.blogspot.de/

[35] Lee, Byoungyoung, Long Lu, Tielei Wang, Taesoo Kim, and Wenke Lee, "From Zygote to Morula: Fortifying Weakened ASLR on Android," in *IEEE Symposium on Security and Privacy*, 2014.

[36] Roland van Rijswijk-Deij, Roland and Erik Poll, "Using trusted execution environments in two-factor authentication: comparing approaches," in Open Identity Summit 2013 (OID-2013), volume 223 of Lecture Notes in Informatics.

[37] PaX Team, http://pax.grsecurity.net/

[38] Shacham, Hovav, "The geometry of innocent flesh on the bone: return-into-libc without function calls (on the x86)," in *14th ACM conference on Computer and Communications Security*, CCS '07. ACM, 2007

[39] Buchanan, Erik, Ryan Roemer, Hovav Shacham, and Stefan Savage, "When good instructions go bad: Generalizing return-oriented programming to RISC," in *CCS' 08: Proceedings of the 15th ACM Conference on Computer and Communications Security*, ACM, 2008

[40] PaX Team, PaX address space layout randomization (ASLR), http://pax.grsecurity.net/docs/aslr.txt

[41] Shacham, Hovav, Eu Jin Goh, Nagendra Modadugu, Ben Pfaff, and Dan Boneh, "On the effectiveness of address-space randomization," in *CCS' 04: Proceedings of the 11th ACM Conference on Computer and Communications Security*, ACM Press, 2004.

[42] Bhatkar, Sandeep, R. Sekar, and Daniel C. DuVarney, "Efficient techniques for comprehensive protection from memory error exploits," in *USENIX Security Symposium*, 2005.

[43] Kil, Chongkyung, Jinsuk Jun, Christopher Bookholt, Jun Xu, and Peng Ning, "Address space layout permutation (ASLP): Towards fine-grained randomization of commodity software," in *Annual Computer Security Applications Conference*, 2006.

[44] Pappas, Vasilis, Michalis Polychronakis, and Angelos D. Keromytis, "Smashing the gadgets: Hindering return-oriented programming using in-place code randomization," in *IEEE Symposium on Security and Privacy*, 2012.

[45] Hiser, Jason D., Anh Nguyen-Tuong, Michele Co, Matthew Hall, and Jack W. Davidson, "ILR: Where'd my gadgets go?" in *IEEE Symposium on Security and Privacy*, 2012.

[46] Wartell, Richard, Vishwath Mohan, Kevin W. Hamlen, and Zhiqiang Lin, "Binary stirring: Self-randomizing instruction addresses of legacy x86 binary code," in *ACM Conference on Computer and Communications Security*, 2012.

[47] Giuffrida, Cristiano, Anton Kuijsten, and Andrew S. Tanenbaum, "Enhanced operating system security through efficient and fine-grained address space randomization," in *USENIX Security Symposium*, 2012.

[48] Snow, Kevin Z., Lucas Davi, Alexandra Dmitrienko, Christopher Liebchen, Fabian Monrose, and Ahmad-Reza Sadeghi, "Just-in-time code reuse: On the effectiveness of fine-grained address space layout randomization," in *34th IEEE Symposium on Security and Privacy*, 2013.

[49] Bittau, Andrea, Adam Belay, Ali Mashtizadeh, David Mazieres, and Dan Boneh, "Hacking blind," in *35th IEEE Symposium on Security and Privacy*, 2014.

[50] National Security Agency, Security-Enhanced Linux, http://www.nsa.gov/research/selinux.

[51] Alves, Tiago and Don Felton, "TrustZone: Integrated hardware and software security," *Information Quarterly*, 3(4), 2004.

[52] Azema, Jerome and Gilles Fayad, "M-Shield mobile security technology: Making wireless secure," Texas Instruments white paper, 2008.

[53] Giesecke & Devrient Press Release, "G&D Makes Mobile Terminal Devices Even More Secure with New Version of Smart Card in MicroSD Format," http://www.gi-de.com/en/about_g_d/press/press_releases/G%26D-Makes-Mobile-Terminal-Devices-Secure-with-New-MicroSD%E2%84%A2-Card-g3592.jsp.

[54] Google Wallet, http://www.google.com/wallet/how-it-works/index.html.

[55] MasterCard Contactless, "Tap to pay," http://www.mastercard.us/paypass.html#/home/,2012.

[56] Zhou, Yajin, Zhi Wang, Wu Zhou, and Xuxian Jiang, "Hey, you, get off of my market: Detecting malicious apps in official and alternative Android markets," in *19th Annual Network and Distributed System Security Symposium*, 2012.

[57] Enck, William, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth, "TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones," in *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2010.

[58] King, Dave, Boniface Hicks, Michael Hicks, and Trent Jaeger, "Implicit flows: Can't live with Em, can't live without Em," on *Information Systems Security*, Springer, 2008.

[59] Bhaiji, Yusuf, "Understanding, preventing, and defending against layer 2 attacks," Cisco, http://www.nanog.org/meetings/nanog42/presentations/Bhaiji_Layer_2_Attacks.pdf.

[60] Cisco, "Configuring DHCP Snooping," http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/4_1/nx-os/security/configuration/guide/sec_nx-os-cfg/sec_dhcpsnoop.pdf.

Author Biographies

Alexandra Dmitrienko is a research assistant at Fraunhofer Institute for Secure Information Technology in Darmstadt (Germany). She obtained her MSc in IT-Security from the Saint Petersburg State Polytechnical University in Russia in 2007. Her academic achievements were honored by the Intel Doctoral Student Honor Award in 2013. Her research is focused on mobile operating system security, runtime attacks, and secure mobile applications (online banking, access control, mobile payments and ticketing). She can be contacted at alexandra.dmitrienko@sit.fraunhofer.de.

Christopher Liebchen is a student at the Technical University of Darmstadt and currently working on his master's degree in IT-Security. His research focuses on system/mobile security, reverse engineering, and runtime attacks. Results of his work have been presented at scientific as well as at industrial conferences. He can be contacted at christopher.liebchen@cased.de.

Christian Rossow is a postdoctoral researcher at the Vrije Universiteit Amsterdam (The Netherlands) and at the Ruhr University Bochum (Germany). Christian completed his PhD studies at the VU Amsterdam in April 2013, and his PhD dissertation was awarded with the prestigious Symantec Research Labs Fellowship award in 2013. His research interests are malware analysis, mobile security, botnet tracking, and denial-of-service attacks. He can be reached at c.rossow@vu.nl.

Ahmad-Reza Sadeghi is a professor of computer science at Technische Universität Darmstadt, Germany and the head of the System Security Lab at the Center for Advanced Security Research Darmstadt (CASED). Since January 2012 he has been the Director of the Intel Collaborative Research Institute for Secure Computing (ICRI-SC) at TU-Darmstadt. He holds a PhD in computer science from the University of Saarland in Saarbrücken, Germany. Prior to academia, he worked in research and development of telecommunications enterprises, such as Ericsson Telecommunications. He is on the Editorial Board of the ACM Transactions on Information and System Security. He can be reached at ahmad.sadeghi@trust.cased.de.

TRUSTED EXECUTION ENVIRONMENT FOR PRIVACY PRESERVING BIOMETRIC AUTHENTICATION

Contributor

Abhilasha Bhargav-Spantzel
Intel Corporation

This article describes how a platform's trusted execution environment (TEE) can be leveraged effectively to provide trustworthy client- and server-side biometric verification. To the service providers the TEE can be attested to provide high assurance about the correctness of biometric verification at the client or server. To the user it provides high confidence regarding the privacy and user control of the biometric data under consideration. Additionally this article shows how portability of biometric authentication can be supported cross platform, thus providing better usability and scale to the identity verification system. Finally we show how this would fit with other forms of identity verification for multifactor authentication, such as cryptographic tokens, location, and group membership, to allow for novel usages for secure privacy preserving messaging. I describe how the richer identity context provides higher security assurance without jeopardizing the privacy of the user, which is protected by the TEE and user policies. With the burden of complexity taken over by the TEE-based biometric engine, I describe how such systems can greatly enhance user experience and enable novel applications.

Introduction

To support online activities, such as commerce, healthcare, entertainment, collaboration, and enterprise usages, it is crucial to be able to verify and protect the digital identity of the individuals involved. Misuse of identity information can result in compromise of the security of the overall system and continued problems to the user such as in the case of identity theft. An approach to achieve high assurance identity verification is the use of multifactor authentication (MFA). Such a process requires an individual to prove his/her identity by demonstrating the knowledge of secrets (such as a password), or what the user possesses (such as hardware tokens or a phone) or who they are (such as a fingerprint).

Biometric data in particular represents an important class of identity attributes. To fully realize their potential, identity verification protocols should be able to support the use of biometric data in combination with other digital information such as passwords, cryptographic tokens, and the richer user context, like user location, other user devices, and so on. In addition, the privacy of the biometric information is very important, especially because of the lack of revocability of such identifiers.

"…the privacy of the biometric information is very important, especially because of the lack of revocability of such identifiers."

As such, the use of biometric data in the context of identity attribute verification poses several challenges because of the inherent features of the biometric data. In general, two subsequent readings of a given biometric do not

result in exactly the same biometric template. Therefore matching against the stored template is probabilistic. Further, there are requirements to protect the biometric data during transmission and storage. Addressing these challenges is crucial not only for the use of biometric data in protocols for identity verification but also for the large-scale adoption of biometric authentication and its integration with other authentication techniques for MFA.

Storing biometric templates in repositories along with other personally identifiable information introduces security and privacy risks. Those databases can be vulnerable to attacks by insiders or external adversaries and may be searched or used for purposes other than the intended one. If the stored biometric templates of an individual are compromised, there could be severe consequences for the individual because of the lack of revocation mechanisms for biometric templates.

"Storing biometric templates in repositories along with other personally identifiable information introduces security and privacy risks."

To overcome the issues related to server-side storage and matching, several efforts in biometric verification technology have been devoted to the development of techniques based on client-side matching. Such an approach is convenient, because it is relatively simple and cheap to build biometric verification systems supporting biometric storage at the client end able to support local matching. Nevertheless, most systems of this type are not secure if the client device is compromised; therefore additional security mechanisms are needed.

"A Trusted Execution Environment (TEE) provides and excellent place for biometric verification and policy enforcement."

- Trusted Execution Environment (TEE) such as Intel® Software Guard Extensions (Intel® SGX) capabilities on platforms, both clients and/or servers, provide an excellent place for biometric verification and policy enforcement. They can be trusted and used effectively by relying parties to authenticate users.
- TEEs for biometric storage and verification ensure better user control and privacy for the user. The biometric data is always protected by the hardware.
- Great user experience is ensured using the portability of the biometric verification if carried out in a trustworthy TEE environment. Users would not need to re-enroll every time they are connecting to new cloud services or new devices.
- Numerous existing usages can leverage TEE-based biometric verification and novel usages are possible using this model. We show a few use cases as examples in the areas of finance, social networking, healthcare, and government security, where biometric security, privacy, and usability are important. We also elaborate on a novel use case employing the same underlying TEE approach—namely face-based messaging, where just having your photograph is sufficient to send you secure messages (no need to exchange cryptographic secrets or passwords).

The rest of this article is organized as follows. In the section "Where Can TEEs Be used in a Biometric System?" I show how TEEs fit into a traditional biometric verification model. This is followed by the section on "Security and Privacy Requirements for TEE-Based Biometric Verification". The section

"TEE-Based Biometric Verification Solution" provides an overview of the basic TEE concepts and many use cases where such a solution contributes to improved security and privacy. I also show how this model leads way for several innovative uses of biometric data and authentication system. This section provides a high level architecture and protocol flow based on one illustrative example. In the sections "Portable Biometric Authentication Using TEE" and "Usability Considerations" I talk about the portability and usability advantages of such a solution. The section "Conclusion" provides a comparison of the TEE-based model to the existing client- and server-side models to motivate innovation and development of biometric solutions using the TEE capabilities of platforms.

Where Can TEEs Be Used in a Biometric System?

Short answer is—everywhere! In this section we investigate the existing biometric system components step by step and show how TEE can be leveraged.

"…the TEE helps mitigate against the attack vectors that exist in every component of the biometric verification system."

A detailed reference model for a biometric system has been developed by ISO/IEC JTC1 SC37[1], which aides in describing the sub-processes of a biometric system. Typically there are four main subsystems in the biometric model, namely the Data Capture, Signal Processing, Data Storage, Matching and Decision subsystems. These subsystems are illustrated in Figure 1. Generally speaking the TEE helps mitigate against the attack vectors that exist in every component of the biometric verification system.

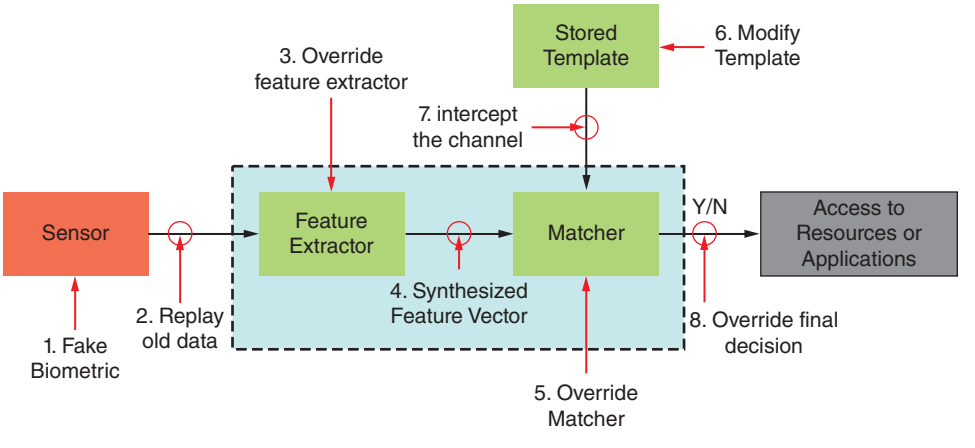


Figure 1: Biometric subsystem and points of attack (Source: Ratha, Connell, and Bolle, 2001[2])

- *Data capture subsystem (sensor):* It collects the subject's biometric data in the form of a sample that the subject has presented to the biometric sensor. *In a TEE-based biometric verification system a trusted channel would need to be created between the sensor and the TEE environment to prevent against spoofing and replay attacks (Attacks 1 and 2).*

- *Signal processing subsystem (feature extractor):* It extracts the distinguishing features from a biometric sample to then either be stored as the reference template during registration or be matched during verification. A template is data, which represents the biometric measurement of an individual, used by a biometric system directly or indirectly for comparison against other biometric samples. *This processing would be done within the boundaries of TEE preventing against any malicious software trying to override this extraction (Attacks 3 and 4).*
- *Data storage subsystem:* Reference templates are stored either at the server or at the client depending on the chosen architecture. *If we use the TEE to store and manage these templates, then it secures them from theft as well as ensuring privacy and rightful use of the stored biometric information (Attacks 6 and 7).*
- *Matching subsystem:* It compares the features extracted from the captured biometric sample against one or more enrollment reference templates. The obtained similarity scores are then passed to the decision subsystem. *In a TEE-based matching subsystem not only do we get high assurance that the template has not been modified but also that the right policy-based configurations are being used for the matching criteria such as the threshold (Attack 5).*
- *Decision subsystem:* It uses the similarity scores generated from one or more matching comparisons to make a decision about a verification transaction. The features are considered to match a compared template when the similarity score exceeds a specified threshold. *When the TEE is responsible for not only the decision making but also the access control or policy enforcement for the resource access, then there is no possibility for any malware to override the final decision (Attack 8).*

Thus we see how a TEE protects all subsystems to ensure an end-to-end trustworthy biometric verification system.

"…a TEE protects all subsystems to ensure an end-to-end trustworthy biometric verification system."

To further elaborate on the end-to-end (E2E) use of a biometric system, let us analyze the two key sub-processes of the biometric system, namely registration (also called enrollment) and verification.

Enrollment

Enrollment is the process of capturing the features from a biometric sample provided by an individual and converting it into a template. The effectiveness of enrollment strictly depends on the quality of the data submitted along with the biometric. Thus, the enrollment process has also to ensure that the verification documents (such as passports and driver's licenses) are trustworthy so that a fake or false identity is not linked to a biometric. Additionally, no duplicate records have to be stored in the database for the same identity. This enrollment mechanism is a key aspect of making biometric verification reliable. Enrollment is the first interaction of the user with the biometric system, and misuses of such operations can affect the quality of the sample being provided by the user, which in turn affects the overall performance of the system. Once

"Enrollment is the first interaction of the user with the biometric system, and misuses of such operations can affect the quality of the sample being provided by the user, which in turn affects the overall performance of the system."

the process of registration is successfully completed, the individual can use the biometric system for verification.

Using a TEE during enrollment would ensure the data could not be compromised or incorrectly linked to the wrong individual, thus ensuring enrollment correctness. More specifically, the stored template would not be replaced, or spoofed, thus ensuring privacy and confidentiality. Additionally for multifactor authentication it is important that the identity record of the user is strongly linked to the right biometric such that there is no identity theft or misuse as mentioned earlier. Relying on a TEE on a local system also allows for online registration of the biometric that can be relied upon by service providers. The lack of TEE would require only an in-person enrollment, which limits the use for several online cloud service scenarios.

"…it is important that the identity record of the user is strongly linked to the right biometric such that there is no identity theft or misuse…"

Verification

The verification is performed when the individual presents his/her biometric sample along with some other identifier that uniquely ties a template with that individual. The matching process is performed against only that template.

For example in traditional fingerprint-based biometric verification systems[3], verification is based on matching of fingerprints. One way to do the matching is to extract the minutiae points of the fingerprint and compare it against the second fingerprint template's minutiae points. The effectiveness of such systems is based on evaluating error rates such as False Accept Rate (FAR), False Reject Rate (FRR), and Equal Error Rate (EER). The processing time for the matching has shown to be efficient (0.2–0.4 seconds) for practical purposes. A relying party or IT administrator should be able to define the EER and corresponding matching threshold.

"Using a TEE during client-side verification allows not only the protection of all biometric information during verification but also the correctness of the biometric protocol used…"

Using a TEE during client-side verification allows not only the protection of all biometric information during verification but also the correctness of the biometric protocol used and the configuration (for example, matching threshold) set by the administrator. It also enhances the final biometric verification by leveraging richer user context (multifactor authentication) and presence detection, which can be attested to the backend systems. This contributes towards liveness detection, which is a key requirement for a biometric verification.

For remote users, biometric verification normally requires server-side biometric template matching. This is because there is no guarantee that the client-side biometric subsystem has not been compromised and a positive match was sent even if the actual matching failed. As mentioned before, the server-side matching has privacy concerns because the biometric data can be stolen or used for other purposes. If a TEE is used at the server end, then users can gain assurance that their biometric information is protected and used as intended by the authorized entities. If a TEE is used at the client end, the service provider can attest the TEE and rely on client-side verification providing better user control on his/her biometric.

Security and Privacy Requirements for TEE-Based Biometric Verification

The biometric authentication system must protect against each of the attack types that can be carried out in a biometric system by malware or malicious users. From a security and privacy requirement perspective there are several key desired properties of the biometric system and processes. These are elaborated in Table 1. A TEE-based biometric system helps achieve each of these requirements.

Requirement	Description
Confidentiality	Confidentiality deals with the protection of the personally identifiable information (PII) from unauthorized disclosure. This property applies to biometric information and transactions in the system. Identity information should only be accessible by the intended recipients.
Integrity	Integrity requires data not to be altered in an unauthorized way. Revocation of trusted parties or any related identity information is required to maintain the validity of the verification system. It should ensure that once invalid information is recognized, it is not used for identity verification purposes.
Unlinkability	Unlinkability of two or more users or transactions means that the attacker, after having observed the transactions, should not gain additional information on linking those transactions. Unlinkability prevents (illegitimate) merging of user profiles by linking them.
User choice and selective release	User choice means that the individual can choose which identity information to release and to whom. Selective release of biometric information means that the biometric information can be released at a fine-grained level based on user choice.
Verifiability	Verifiability means that the individual can verify that the correct protocol was followed and the intended identity data is being used for the defined purpose.
Nonreplay	Nonreplay of the biometric data provided in transactions prevents unauthorized parties from successfully using an individual's biometric data to conduct new transactions. This is also related to the liveness analysis in a biometric system. Nonreplay is one prerequisite for obtaining the nonrepudiation property.
Nonrepudiation	Nonrepudiation of transactions and identity data itself means that the sending of a nonrepudiable identity data cannot be denied by its sender: (1) the ownership of the identity data cannot be denied; (2) the presence or liveness of the biometric cannot be denied. If the TEE can leverage other platform sensors and advanced platform capabilities such as user presence in the MFA engine, then the important property of nonrepudiation can be better achieved.
Stealing protection	Stealing protection applied to identity data is concerning the issue of protecting against unauthorized entities illegitimately retrieving an individual's data items. Stealing protection is required to achieve properties such as nonrepudiation.

Table 1: Security and Privacy Requirements for a TEE-Based Biometric Verification System
(Source: Intel Corporation, 2014)

TEE-Based Biometric Verification Solution

This section describes the basic concepts behind trusted execution environments (TEEs) and several interesting use cases that can be enabled based on the TEE authentication model. An example architecture is also provided to help illustrate how one of those use cases can be implemented using existing and upcoming technology components.

TEE Basic Concepts

There are several reasons why a standard computing environment is vulnerable to attacks. Some primary ones include human behavior (such as clicking on a link pointing to a malicious site), unpatched software vulnerable to malware attacks, keyloggers, rootkits, advanced persistent threats, and so on. Generally speaking, the trusted execution environment (TEE) is a secure area that resides in the main processor of a device and ensures that sensitive data is stored, processed, and protected in a trusted environment. The TEE's ability to offer safe execution of authorized security software, known as "trusted applications," enables it to provide end-to-end security by enforcing protection, confidentiality, integrity, and data access rights.[4] TEE is isolated from the "normal" processing environment, sometimes called the rich execution environment (REE), where the device operating system and applications run.

There are several TEE specifications and solutions available in the market today. Some examples include ARM Trustzone[5] implementing the Global Platform TEE specifications, embedded secure elements using JavaCard, TEE based in Dynamic Root of Trust (DRTM), and TEE based on a virtual machine monitor (VMM).[6]

Intel Management Engine has been used as a TEE for several security capabilities of the platform such as those available for Intel® vPro™ systems. Systems capable of employing Intel® Software Guard Extensions (Intel® SGX) are available to create TEEs that we can leverage for biometric verification as described in this article. Using Intel SGX would allow biometric solution applications to protect sensitive biometric and user identity data from unauthorized access or modification by rogue software running at higher privilege levels. It also enables biometric verification applications to preserve the confidentiality and integrity of the sensitive (biometric verification) code and data without disrupting the ability of legitimate system software to schedule and manage the use of platform sensors and resources. Intel SGX can also connect with other platform sensors and capabilities such as Intel location-based services, Intel® Identity Protection Technology (Intel® IPT) with MFA solutions, and so on. Leveraging the rich security and identity infrastructure on the client would further enable the biometric system to check for biometric liveness, user presence, and service usage policy specification and enforcement. Overall the Intel SGX and related platform capabilities allow trustworthy use of local and remote biometric verification both by users as well as relying parties or service providers.

"Leveraging the rich security and identity infrastructure on the client would further enable the biometric system to check for biometric liveness, user presence, and service usage policy specification and enforcement."

Use Cases

The following are some interesting use cases that are possible only because of the TEE-based identity verification model. Note the advantages related to the security, privacy, and usability using such a model.

Use Case 1: Financial Services

A trustworthy biometric engine on the platform can help secure and ease online or in-person financial transactions in several ways. Imagine using a

handprint or fingerprint enrolled in person with your bank to withdraw money from ATMs without having to carry a card or remember the PIN. Taking it further, one can envision paying at a restaurant or a shop with the biometric verification linked to an account or the user's mobile device. The trustworthiness of a local or remote TEEs on the verification devices greatly enhances how the authentication can be relied upon and assurances of privacy. As mentioned earlier in the section "TEE Basic Concepts," if the TEE leverages other platform sensors such as user presence detection and MFA engines, then the biometric liveness detection can be improved significantly.

This solution helps overcome one downside of a lot of biometric-based authentication, which is the need for in-person enrollment. In the example below we show how a TEE-based messaging and biometric verification system enables remote discovery of the user with a given biometric and uses it to enroll and establish trust relationship with that user. This capability is patented. The mechanics of how the problem in this specific example can be solved is described in the section "Approach and Logical Architecture."

"…a TEE-based messaging and biometric verification system enables remote discovery of the user with a given biometric and uses it to enroll and establish trust relationship with that user."

Example 1: Secure Bank Account Trust Establishment based on TEE—Consider a bank called SecureBank that has customers who may be residing in all parts of the world. Alice is one such customer. Alice is required to enroll in person where several identity details are verified and a picture is taken for reference. She lives in a different country and uses her email for basic communication. For secure communication SecureBank requires that the messages be encrypted. However Alice does not have the key and given the long waiting times and time zone difference it is very inconvenient for her to set the key or password over the phone.

It would be very beneficial if SecureBank had a way to use Alice's photograph and depend on a trustworthy face recognition client subsystem at Alice's end to send this key. The burden of complexity for authentication is taken away from the user to the trustworthy client subsystem running in a TEE.

Additionally, if there were an ability to transparently comply with policy conditions that some legal documents may have such as user presence and confirmation of viewing, then it would greatly enhance the user experience and secure usage of the services.

Use Case 2: Social Networking

In today's world a lot of the social networking sites use the names and demographic information to find friends or people a user might like to network with. Often the interaction can be compromised due to attacks such as identity theft. Additionally a given user cannot control the conditions under which his or her message is delivered or used. Discovering people with the help of biometric data and using biometric verification and presence information can greatly enhance social networking use cases and interaction. The example below elaborates on this idea with a specific scenario.

Example 2: Privacy Preserving P2P Social Networking—Consider Bob who knows Cathy at an Alcoholics Anonymous meeting, and they had photographs of each other and a nonidentifying email address. If Bob wanted to send Cathy a private message that should only be viewed by Cathy, there is no way of doing that based on the information that Bob has (based on photograph and email). Bob and Cathy may also talk over a secure chat session, which requires that the chat message appears on a trusted output display and that the message appear only when the recipient is in front of the screen.

"…P2P private messaging could be easily possible with no complex user interaction and key exchange."

If Bob could use the photograph and depend on a trustworthy client system at Cathy's end to verify the face and deliver the message securely, then this P2P private messaging could be easily possible with no complex user interaction and key exchange. The client system would be responsible to first verify Cathy's face using face recognition systems, compare it against the photograph, and then enforce policies for the usage of the message (for example, decrypt only when the user is viewing the screen).

Such mechanisms can be made possible with a TEE-based biometric verification system as described in the section "Approach and Logical Architecure."

Use Case 3: Government and National Security

National security and transportation security (such as the TSA in the United States) often rely on identifying the user with a given biometric (face or fingerprint) to identify legitimate and malicious users. Critical checkpoints not only depend on government-issued credentials but also biometric verification to identify on trusted databases. Biometric subsystems running in a TEE can be attested to ensure correctness of the procedure and protection against malware.

"…such a system can potentially be used as a distributed threat identification system in smart cities."

In the future, such a system can potentially be used as a distributed threat identification system in smart cities. The term *smart cities* refers to cities enhanced with rich IT infrastructure with high availability and quality of knowledge, communication, and social infrastructure. There are current efforts to extensively deploy wireless sensor networks, which can also be used to identify users and potential threats. To use such an infrastructure relying on the capture and use of PII and biometrics, we can see how a TEE-based biometric verification system can help protect the distributed sensors and evaluation engines deployed while at the same time preserving privacy by preventing data exfiltration and unauthorized use of the data.

Use Case 4: Healthcare and Telemedicine

Similar to the above use cases, the TEE-based biometric verification system can provide secure and user-friendly mechanisms to protect health IT usages including medical records management, EMR access control, telemedicine, and health insurance infrastructure. It also helps in enabling healthcare innovation to scale up healthcare solutions to reduce cost with better security and accountability.

For example, consider the area of telemedicine that is being actively investigated by organizations like the Veteran's Administration to provide personalized and effective care to the veterans. Current solutions don't scale and lack accountability because of the lack of caretakers and the necessary IT infrastructure. TEE-based biometric verification systems can help in ensuring that the biometric information is used to identify the right user accessing the right medical record in the cloud and to provide secured biometric readings that themselves can be used for healthcare assessment (such as ECG). In addition, it does not require the user to remember complex passwords, providing an opportunity to employ self-service telemedicine more effectively.

"…biometric information is used to identify the right user accessing the right medical record in the cloud…"

Approach and Logical Architecture

The key components of our proposed TEE-based biometric system are :

1. Sender Client System
2. Receiver Client System
3. Attestation Trust Broker

These are illustrated later in this article. I describe how this model would work with a novel use case described in the earlier examples 1 and 2. They involve a privacy-preserving message exchange with no user-managed passwords or keys.

This model ensures three key features:

1. *TEE attestation*—Providing the trustworthiness of the TEE environment where the biometric verification is taking place.
2. *Policy enforcement*—Ensuring that the biometric verification policies are enforced by the TEE.
3. *Use of multifactor authentication and user context*—Linking the TEE to the rich context provided by the platform, including ability to do multifactor authentication, location sensing, and security profile detection.

The rest of this section provides details of the system applied to a face-based secure messaging usage.

Face-Based Messaging Example

The underlying idea is based on the fact that in today's world, pictures of people's faces are universal. At any given second, studies suggest, about 500 billion to 1 trillion photos are available online. Photos are taken in numerous consumer lifestyle settings and also for many businesses and services. However they are not useful as a biometric authentication mechanism for a remote sender to securely transmit confidential messages to an individual whose photograph is available to the sender.

"At any given second, studies suggest, about 500 billion to 1 trillion photos are available online."

This is because there are no trustworthy mechanisms that

1. *Enforce the identity verification* ensuring the correctness of the biometric verification and liveness tests; and
2. *Enforces secure message delivery policies* such as that the message should be decrypted only if the user is in front of the screen.

"Current mechanisms depend on complicated passwords or public key infrastructures that are not usable."

Current mechanisms depend on complicated passwords or public key infrastructures that are not usable. Alternate mechanisms to send messages based on photographs could be by using cloud services such as Facebook but they do not provide the necessary security and privacy controls.

The following protocol is one solution to solve this problem using a TEE.

The message components involved in the Face-based Secure Messaging (FSM) protocol are as follows:

- *Enc_M:* Encrypted message (text, media, and so on)
- *Dec_M:* Decrypted message
- *Pol_M:* Sender-defined policy for message access
- *Receiver Client System Attestation requirements:* For example, using an Intel TEE such as Intel SGX
- *Private Message Use Obligations:* For example, using a specified face recognition application with a given threshold match
- *Private Message Use Restrictions:* For example, do not allow a decrypted message to be visible for more than 10 minutes
- *Quote_Rec:* Receiver systems Attestation Quote—a signed blob that has the result of the receiver client attestation
- *ZKProof_Rec:* Proof of compliance of the Pol_M created by the TEE on the receiver side and sent to the sender. This proof is similar to a zero knowledge proof where compliance with the policy is verified but no additional information about the receiver is leaked.
- *Key_M:* The secret key required to decrypt the message. The understanding is that this key is communicated securely from the sender system to the TEE of the receiver system.
- *Key_Com:* A communication key to securely communicate with the end point. This could be an RSA public key.

The protocol flow is as follows (this flow is illustrated in Figure 2):

1. The sender creates a request package denoted as an FSM Request, which contains four key components:
    a. Receiver Picture
    b. Pol_M (policy to access the message)
    c. Enc_M (message encrypted with a symmetric key Key_M)
    d. Key_Com1: public key of the sender

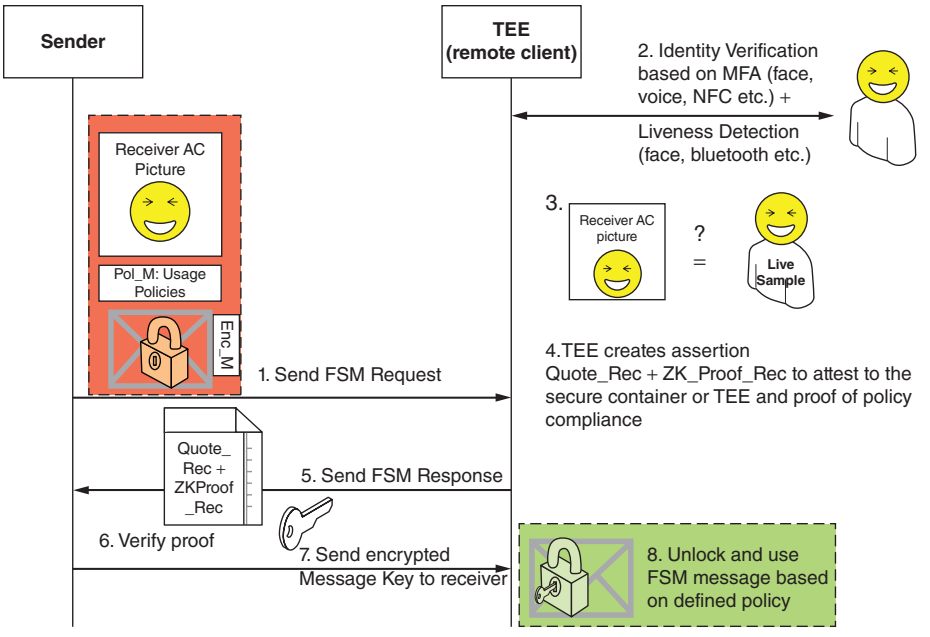


Figure 2: Example End to End Flow
(Source: Intel Corporation, 2014)

2. Once the receiver FSM app (running in TEE) gets this message, it starts by verifying the user identity and presence. The requirements for user authentication and presence detection may be specified in Pol_M
3. Then the FSM collects a live sample of the user in front of the camera. The FSM application is considered trustworthy to do the collection and appropriate liveness detection (potentially using other platform sensors). Matching is based on threshold policy defined in Pol_M. The TEE should establish a trusted-path connection to the camera as defined by Pol_M.
4. The TEE then creates a Quote_Rec (similar to TPM_Quote in TXT[7] attestation or EPID[8] based or SGX attestation) and also the ZKProof_Rec, which contains the result of the calculations from steps 2 and 3 signed by the client key. The Receiver AC picture is post-processed to extract a facial recognition template that may be matched against a template collected in step 3.
5. At this point the TEE encrypts the FSM Response with Key_Com1 and provides Key_Com2 that is the public key of the receiver TEE.
6. The sender then verifies the attestation quote and the zero knowledge proof.
7. If the verification is successful it encrypts the Key_M with Key_Com2 and sends it to the receiver FSM app.
8. As the final step, the FSM app in TEE uses the Key_M to decrypt the message and enforce the message use policy defined in Pol_M. The policy can be enforced using TEE services such as secure display, data sealing for later use, or trusted time for time-based policies.

"…the FSM collects a live sample of the user in front of the camera."

Note that steps 4 through 8 could be implemented in part using Intel's Sigma technology.[9] Additionally steps 4 through 8 may be implemented within an SMTP and IMAP protocol where Quote_Req, ZKProf_Rec, Key_m, and Key_comm objects are packaged as MIME extensions to an automated email processing system.

"…the sender can establish trust once and use the same encryption key until it decides to change the key."

In the above flow the receiver must attest itself to the sender and get the key each time. However, depending on the sender policy, the sender can establish trust once and use the same encryption key until it decides to change the key. This would avoid the need for the sender system to be up and running and reach the Attestation server when the receiver wants to consume the message. This would also avoid the need for a cloud service to set up the communication, that is, standard messaging or mail servers can be used.

Another extension of the above flow would be the ability to use this protocol in a bidirectional fashion to verify the authenticity of the sender as well. In this case the sender would have to prove that he or she is the same one whose picture is available to the receiver.

A logical architecture diagram of the proposed solution is provided in Figure 3. The trust attestation service illustrated in Figure 3 corresponds to a service that can be used by the sender to verify the attestation quote provided by the receiver system to check to see whether the TEE and client system is considered trustworthy.

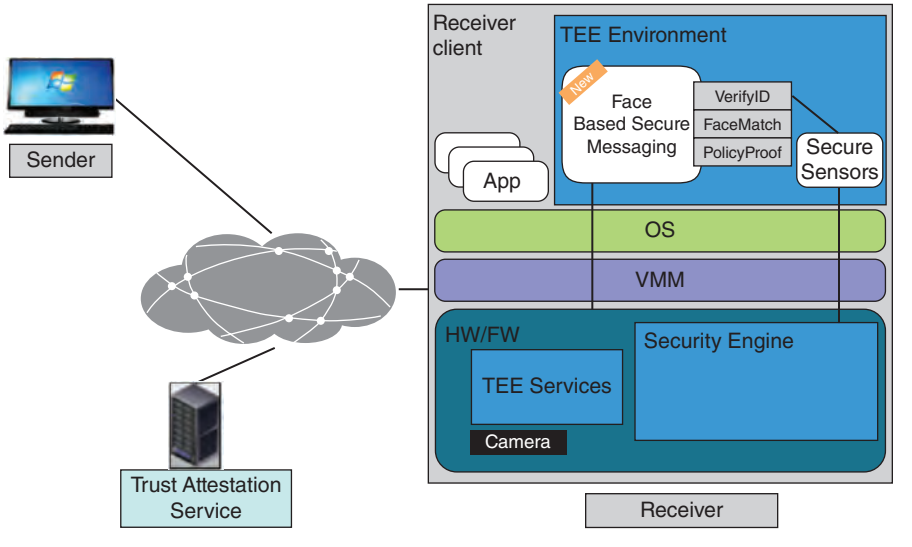


Figure 3: Logical architecture of the E2E TEE-based secure messaging solution (Source: Intel Corporation, 2014)

The above approach and logical architecture can be tweaked based on the type of use case and solution requirements. The example shows how the TEE component is a fundamental building block for high assurance and trustworthiness of the overall solution depending on biometric authentication.

Portable Biometric Authentication Using TEE

The TEE can potentially be used as a portable biometric authentication store. This can be done by a user's local biometric system with TEE capabilities or even a remote cloud biometric system with a TEE.

Portable biometric with local device TEE: Consider a mobile device with a biometric verification system that runs in a TEE. Since users carry this system with them at all times, it can be used effectively as the users' credential manager. Access to multiple systems can be controlled based on credentials released by the mobile device. The TEE ensures that the credentials are released securely only when a valid policy-based authentication occurs. For example, a user can authenticate to his or her phone with a fingerprint and the phone can create a secure channel with the user's laptop to communicate the user's password or signed certificate for user logon.

"A mobile device … can be used effectively as the users' credential manager."

Portable biometric with cloud TEE: Cloud-based TEE has been explored for various security usages such as data anonymization[10] and trusted compute pools.[11] Biometric verification in a cloud-based TEE can provide the privacy and trustworthiness guarantees of the biometric verification while ensuring portability and cross-platform usage. There might be challenges based on the different types of sensors used to collect that given biometric information, but with advanced sensor capabilities of client platforms such as the next generation Intel client systems containing dual microphone array, high resolution 3D camera, and so on, we can see how this potential challenge can be resolved.

A hybrid model containing an E2E trust with a client-based TEE and the TEE in the cloud can also provide a flexible and scalable portable biometric verification model.

Usability Considerations

The TEE-based biometric verification system takes the burden of computation and complexity away from the user to the TEE environment itself. As we saw in the face-based messaging protocol in the section "Approach and Logical Architecture," the complex protocol used to (1) securely negotiate trust, (2) provide a zero knowledge proof of identity and (3) verify liveness, followed by the (4) policy usage enforcement, and so on—all was transparent to the user. With the right network connectivity and computation power the entire interaction would be seamless and the user would not be required to take special steps to establish trust and authenticate. This is related to the passive MFA solutions in "Adding Nontraditional Authentication to Android," an article in this issue of the *Intel Technical Journal* that describes nontraditional authentication mechanisms and the advantages to user convenience. Additionally, the various use cases described earlier can be successful because of the underlying requirement for transparency and minimal user involvement and because they depend on the TEE to ensure security of the biometric authentication process.

"The TEE-based biometric verification system takes the burden of computation and complexity away from the user…"

Conclusion

There are several advantages of the TEE-based biometric verification system as illustrated in Table 2. It also opens exciting new possibilities for the use of biometric information ensuring security, privacy, usability, and scale. With advanced TEE capabilities present in Intel server and client platforms (such as Intel SGX) we see how they can be effectively used to provide an E2E trustworthy client- and server-side biometric verification system. Thus the models discussed in this article motivate innovation and development of biometric solutions using the TEE capabilities of platforms.

#	Requirement	Client-Side Matching	Server-Side Matching	TEE-Based Matching
1	Confidentiality	Yes	No	Yes
2	Integrity	No	Yes	Yes
3	Unlinkability	No	No	Yes
4	User choice and Selective release	Yes	No	Yes
5	Verifiability	No	No	Yes
6	Nonreplay	No	Maybe	Yes
7	Nonrepudiation	No	Maybe	Yes
8	Stealing protection	No	No	Yes
9	Portability	No	Yes	Yes

Table 2: Comparison of Traditional and TEE-Based Biometric Verification System
(Source: Intel Corporation, 2014)

Complete References

[1] http://www.iso.org/iso/iso_technical_committee.html?commid=313770

[2] Ratha, N., J. Connell, and R. M. Bolle, "An analysis of minutiae matching strength," In *Audio and Video-Based Biometric Person Authentication* (Springer: Berlin, 2001), J. Bigun, and F. Smeraldi (editors), pp. 223–228, 978-3-54042-216-7.

[3] www.validityinc.com/

[4] http://www.globalplatform.org/mediaguidetee.asp

[5] http://www.arm.com/products/processors/technologies/trustzone/index.php

[6] https://wiki.helsinki.fi/download/attachments/117218151/SP-2013-06-0097.R1_Kostiainen.pdf

[7] http://www.intel.com/content/www/us/en/architecture-and-technology/trusted-execution-technology/malware-reduction-general-technology.html

[8] https://eprint.iacr.org/2007/194.pdf

[9] eprint.iacr.org/2010/454.pdf

[10] http://www.intel.com/content/dam/www/public/us/en/documents/best-practices/enhancing-cloud-security-using-data-anonymization.pdf

[11] http://www.intel.com/content/www/us/en/architecture-and-technology/trusted-execution-technology/malware-reduction-general-technology.html

Author Biography

Abhilasha Bhargav-Spantzel is an information security professional with many years of research and industry experience with a focus on security, privacy, and identity and access management. For the last seven years at Intel, she has held the role of security solutions architect within multiple product and architecture groups spanning the end-to-end software and platform lifecycle phases. She has had extensive exposure to customer facing and eco-system partner engagements. She is a recognized expert in the area of identity management and has given numerous talks at conferences and universities as part of distinguished lecture series and workshops. She has written five book chapters, over 30 ACM and IEEE articles, and has over 15 patents. She has worked with several standards organizations, including the Open Data Center Alliance, CSA, Liberty Alliance, and OASIS. She has co-chaired the ACM Workshop on Digital Identity Management and continues to serve in the program committee for various security conferences. Previously she worked as a resident fellow at Symantec and conducted research at IBM. Abhilasha holds a doctorate degree specializing in Security and Identity Management and bachelor's degree in Computer Science and Mathematics with minors in Physics and Psychology from Purdue University, Indiana. She can be reached at abhilasha.bhargav-spantzel@intel.com.

PROTECTING SENSOR DATA FROM MALWARE ATTACKS

Contributors

Jonathan Edwards
McAfee

Ken Grewal
Intel Labs

Michael LeMay
Intel Labs

Scott H. Robinson
Intel Labs

Ravi Sahita
Intel Labs

Carl Woodward
McAfee

"…sensor-driven usages and underlying sensor capabilities attract attacks that threaten the connected world's privacy and security."

"To help mitigate these threats, we describe an architecture using Intel® Virtualization Technology (Intel® VT-x, Intel® VT-d) to provide access controls for sensor data and software…"

A connected, intelligent world is being forged where data from sensors is used to make decisions, take actions, and deliver compelling user experiences. The effects are seen across industry, enterprises, and consumers: keyboards, microphones, and video cameras are firmly entrenched in many applications. Natural human-computer interfaces (such as speech and gesture), authentication, and context-aware computing (such as digital personal assistants) are emerging usages that involve always-on, always-sensing platforms that observe both users and their environments. Sensors include touch, fingerprint, location, audio, imaging, and accelerometers, which are either directly connected to devices or available over wireless interfaces, such as Bluetooth.*

These sensor-driven usages and underlying sensor capabilities attract attacks that threaten the connected world's privacy and security. Keyloggers capture financial data and passwords, other malware activates microphones and cameras to spy on unwitting users, and cloud services may leak uploaded private data due to vulnerabilities or use and distribute it in undesirable ways. Unmitigated, these threats negatively affect the reliability and trustworthiness of new services and devices relying on the sensor data. Some environments even require a baseline level of trustworthiness that may be unattainable without added protections, such as laptops or phones with integrated cameras used in secure or restricted areas.

To help mitigate these threats, we describe an architecture using Intel® Virtualization Technology (Intel® VT-x, Intel® VT-d) to provide access controls for sensor data and software that operates within different operating systems. New instructions extending Intel® VT (VMFUNC and #VE) provide introspection capabilities beyond the classical virtualization models. In this article, we describe how we can use these Intel® 64 and IA-32 architecture primitives to protect sensor data flow as the data enters host memory and as it traverses through a partially trusted OS stack to various authorized local or cloud applications. The architecture can attach platform integrity information to the sensor data, while minimizing the trusted computing base (TCB) for software components involved in the data touch points. These capabilities can be extended from the platform to the cloud infrastructure to provide end-to-end access control for sensor data, providing the automatic security controls over the data, hence preserving the user experience for application interactions.

Introduction

Great benefits across many aspects of modern society can be reaped by building a connected, intelligent world where data from sensors are used to drive high-value decisions, actions, and user experiences. Sensors can enable consumer

and industry usages such as biometric-based authentication (such as facial and voice recognition), natural user interactions and interfaces, contextual computing, command and control, data entry (keyboard, touch, speech), and data sharing and analytics. The world is made intelligent because sensors can, for example, provide information about material composition, pressures, temperatures, orientation, acceleration, location, gender, emotional state, who you are with, what objects are nearby, where you are, what you are doing, what you are saying—there are many more.

"New instructions extending Intel® VT (VMFUNC and #VE) provide introspection capabilities beyond the classical virtualization models."

Risks to the Connected, Intelligent World

But realization of such a world of intelligent services, conveniences and efficiencies is threatened because: (1) valuable data attract a variety of abuses and threats; and (2) people and institutions care about their privacy and security and will shun, prohibit, or prevent uses that expose them to such risks.[3][4][5][6][7][8] Recent events show that attacks on sensor data are rising and can be used to modify or replay data, or expose direct and indirect information leaks about the user and their environment. Examples include theft of authentication credentials for banking (keylogging), falsifying traffic information, sextortion, and nation-state and industrial espionage.[9][10][11][12][13][14][15][16][17][18][19][20]

The response by classic security practices is to prevent or remove capability and to add complex policy decisions to the user's responsibilities. But, this is precisely what must be avoided; the connected world won't be intelligent if users are burdened with vetting each piece of data or decision! Automatic policies and controls are needed to protect the data, while preserving the user experience and allowing the user to focus on the tasks at hand.

Sensor Data Protection Objectives

Our vision is to deliver a trustworthy computing platform that provides a consistent and secure sensor solution optimized for privacy, cost, power, performance, and user experience. The sensor data protection goals are as follows:

"…vision is to deliver a trustworthy computing platform that provides a consistent and secure sensor solution optimized for privacy, cost, power, performance, and user experience."

- Input sensor and human-input device data is protected at the source and remains access-controlled during processing from within different layers of the operating system stack from any software threats.
- The user is provided with an easy-to-use environment, with automatic controls on sensor data processing, while additional controls are provided (such as, for example, to an IT department in the corporate landscape), to fine-tune the policy engines based on corporate requirements.
- An approach is used that is scalable across different platforms and sensor types, has a consistent software architecture, and can provide evidence of data authenticity based on the originating platform's hardware and software configurations.

This article describes one promising methodology for delivering sensor data protection—the use of Intel® Virtualization Technology (Intel® VT), Intel® Virtualization Technology for Directed I/O (Intel® VT-d) and related technology.[1][2] We show that hardware-based virtualization acceleration

coupled with a minimal memory-virtualizing hypervisor using Intel® VT with Extended Page Tables (EPTs) provides isolation of sensor processing code and data in an efficient manner, meeting the desired security goals. The result is a platform with an authorized and attestable configuration that produces trusted sensor data flows that have controlled data forwarding and use.

The use of virtualization technology here differs from traditional virtualization discussions where hypervisors (also known as virtual machine managers or VMMs) are typically used to host separate virtual machines (VMs), each running a guest operating system. Indeed, VMs can provide powerful separation and isolation properties between different guest VMs. Our focus is on using virtualization to support introspection for protecting OS and user application components within a guest virtual machine. Thus, these techniques may be used to augment existing hypervisor (VMM) systems managing multiple guest VMs.

Threats and Security Requirements

"A sensor protection architecture protects sensor data by establishing a secure path from the sensor source through hardware and firmware to software processing."

A sensor protection architecture protects sensor data by establishing a secure path from the sensor source through hardware and firmware to software processing. With proper policy controls in place, this allows the user or environment to establish trusted sensor data flows that have authentic data and controlled data forwarding and use. This includes considerations for protecting data at rest and when the data is released off platform.

This article focuses on the predominant adversaries for consumer and enterprise markets: software-based attacks, such as application space and operating system kernel malware. Such attacks can be perpetrated without physical access to the device or environment and represent the preponderance of today's threats.

The biohazard icons in Figure 1 mark the potential software threat areas posed by these software (and firmware) adversaries. As shown, sensor data enters via the hardware domain I/O controller and may be additionally processed by one

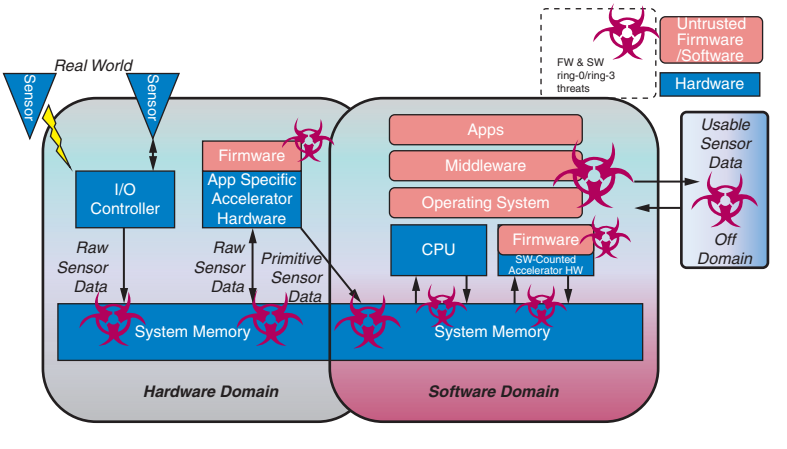


Figure 1: Ring-3/Ring-0 software attacks (Source: Intel Corporation, 2013)

or more firmware-controlled (FW) accelerators that require memory buffering often reserved from system memory. Eventually, data from the hardware domain passes into the software domain via buffers, again, often carved from system memory. From there, software and accelerators process the data. Thus, adversaries penetrating host software, accelerator firmware, or accelerator software can gain read/write access to plaintext sensor data and code (drivers, middleware, applications) directing the processing and distribution of such data, even while data is being processed by the hardware domain because buffering takes place in system memory.

Table 1 enumerates assets that need protection for a given sensor stream, as well as the ownership of these assets and the associated threats. In Table 1, Ring-0 is kernel-mode software and Ring-3 is user-mode software.

Asset	Access control or Ownership	Threats
Sensor Data	Operating System (OS) or Virtual Machine Monitor (VMM)	Ring 0/3 malware can access/modify/replay the data
VMM	OS or Boot loader	Ring 0 malware can access/modify VMM code/data Unauthorized VMM
PCIe Configuration space (Device Base Address Ranges [BAR])	OS or VMM	Ring 0 malware can modify device BAR
Device Registers (MMIO [operational, runtime, etc.])	Device Driver	Ring 0 malware can access/modify/remap MMIO space
Device driver memory (TRs or TRBs, DMA buffers, etc.)	Device Driver	Ring 0 malware can access/modify connection-specific data structures
Kernel and Driver code or data segments	OS or Device Driver	Ring 0 malware add hooks for driver functions and access/modify driver code/data
Paging (IA-PT/EPT) data structures	OS or VMM	Ring 0 malware can change Paging structures
Interrupts	OS or VMM	Ring 0 malware can remap interrupts
Accelerator firmware and software	VMM, OS, or Drivers	Malware can obtain read/write access to data
BIOS and System Management Mode	BIOS services and SMM code	Malware can obtain read/write access to system-wide code/data.

Table 1: Assets, Ownership, and Threats
(Source: Intel Corporation, 2014)

Our objective is to mitigate threats to the sensor data streams shown in Table 1, including software stacks and other platform elements such as accelerators. The ideal solution identifies the smallest set of components that must have access to the sensor data or have a role in protecting those assets—these are collectively termed the Trusted Computing Base or TCB.[24] Access by all other components outside of that set must be prevented. Intuitively, a minimal TCB is critical because flaws (bugs or vulnerabilities) in the TCB may compromise asset security, but flaws or bugs outside the TCB should not. The smaller the TCB, the easier it is to provide assurances on the trustworthy nature of the code within the TCB and hence assurances on the data being managed. Access to the sensor data is only provided to

"…a minimal TCB is critical because flaws (bugs or vulnerabilities) in the TCB may compromise asset security, but flaws or bugs outside the TCB should not."

"Access to the sensor data is only provided to components within this minimal TCB and everything else is denied access."

components within this minimal TCB and everything else is denied access. In this case, the (TCB) includes:

- Sensor hardware and firmware for sensor subsystems
- Sensor drivers and accelerators (such as GPUs)
- Sensor data buffer memory
- Middleware/applications that may process sensor data
- Associated sensor data policy control infrastructure (such as whitelisted applications)
- Assets used to establish these protections (such as a secure boot policy for the VMM)

Our main focus is on the software TCB, because the hardware components are assumed to be immutable, while firmware components are typically measured before loading.

For firmware such as that loaded by drivers (for example, audio DSP firmware), we assume well-known validated software loading methodologies are used.[22] And, of course, the solution must protect against attacks made by leveraging hardware, firmware, and software outside of this TCB (such as device DMA attacks). While denial-of-service attacks are not generally mitigated, we show that many software-initiated read/write attacks can be blocked and operation can continue with data confidentiality and integrity intact. Some portions of the TCB must be protected in other ways, such as by monitoring. In those cases, detection of an attack would be handled according to policy. Such policy could, for instance, securely terminate a sensor stream, log the event, and notify the user.

Because this proposed architecture employs a memory-virtualizing hypervisor (VMM), derived assets for the operating system (OS) and VMM must also be considered as points for attack. In particular, certain portions of the OS can be used to mount attacks such as MMIO or PCIe base address remapping or confused deputy attacks such as using OS-level guest-linear-to-guest-physical page table remapping attacks. Several of these vulnerabilities are discussed by Diskin[21] and mitigations are described in later sections of this article.

"The solution must move sensor data from the sensor hardware device (such as a camera) to the authorized destination without leaking information outside to unintended components…"

Security Properties and Caveats

The solution must move sensor data from the sensor hardware device (such as a camera) to the authorized destination without leaking information outside to unintended components; this privacy property is known as data confidentiality. For some uses we also seek to protect the data from unintended modification or to ensure that the data was recently created and not a replayed version of previously transferred data; such integrity and freshness properties are needed for uses such as biometric authentication. Properly constructed software, with additional mitigations, can provide sensor data from an identified sensor with these properties. TCB attestation is discussed later in the context of a root-of-trust secure boot. The attestation provides evidence that the data was produced

on a known platform running a valid hardware and software configuration. It should also be noted that such a solution does not guarantee goodness or integrity of the data coming from the hardware or the TCB software because, for instance, these may have failures or bugs. Similarly sensor data ordering, data delay, or delay estimation and data availability may not be guaranteed by the TCB software if the sensor hardware or firmware has been compromised.

Background

The architecture described herein leverages the use of memory-protection–based isolation techniques to protect platform sensor data handling against software-based threats. Protection is achieved through the use of hypervisor-controlled "memory views" constructed using existing hardware-accelerated virtualization technology, Intel VT.

"Protection is achieved through the use of hypervisor-controlled "memory views" constructed using existing hardware-accelerated virtualization technology, Intel VT."

In the Intel 64 and IA-32 architectures the operating system maintains page tables that translate linear addresses into physical addresses. Page faults (#PF) are then delivered through the Interrupt Descriptor Table (IDT) to the appropriate OS handler. When a VMM hypervisor turns on Extended Page Tables (EPTs), an additional layer of address translation is introduced by the hypervisor, called the host. The VM containing the OS is said to be hosted as a guest, so the OS-maintained linear and physical addresses are now called guest linear addresses (GLA) and guest physical addresses (GPA), respectively. EPTs translate from guest physical addresses (GPA) to host physical addresses (HPA) and EPT violations are delivered to the security hypervisor (VMM) via a VM-exit control transfer and notification facility. Following the VM-exit, the VMM could either handle the event or pass it to the guest for handling. The advantage of handling the event in the guest is that the event handler has much more context to work with. The disadvantage is a potential loss of performance. Intel VT was originally optimized for handling virtual machines and not introspection within a guest, new instructions (VMFUNC) drastically reduce this performance overhead, by handling specific functions (such as EPTP switching) in hardware. Additionally, exception events may be passed to the guest via the VMM in a soft manner or directly reported by the CPU using a Virtualization Exception (#VE).

In this article, a *memory view* is a guest physical memory address space with an associated set of memory protections. The protected memory view is used to hold code or data as part of the minimal TCB for a given sensor flow. Most often memory view address spaces are managed at the page level. While memory isolation may also be accomplished by partitioning memory across multiple VMs, this architecture does not use VMs to create separate protection domains.

"The protected memory view is used to hold code or data as part of the minimal TCB for a given sensor flow."

As shown in Figure 2, a memory view, represented by a colored rectangle, is a set of access control permissions applied to memory regions within the sensor flow TCB. Memory views may or may not share overlapping regions of memory. Multiple layers of permissions can be applied to further restrict access.

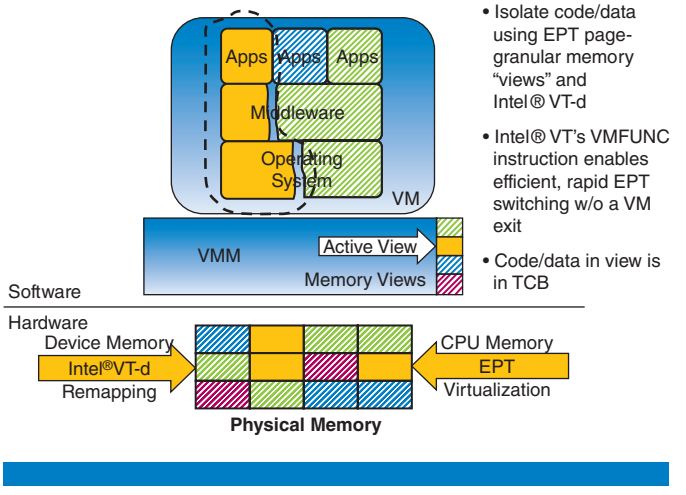


Figure 2: Intel® VT-x2 EPT and Intel® VT-d memory view protections
(Source: Intel Corporation, 2014)

These permissions ensure that only code within the TCB has access to the code's associated data. At the device level, only drivers for a given device have access to that device. All other resources outside the TCB are denied access to these resources.

A given EPT (memory view) is activated when the hardware is directed to use that page table hierarchy for address translation and access permissions. Operating systems use similar techniques (multiple page tables, one per process) to provide each process with their own virtual address space while also admitting the OS address space into each process' page table.

"Memory views permit different partitions on memory access to be rapidly switched and imposed."

In Figure 2, the yellow page table is active and so the code/data captured by those pages is considered to be in the protection domain of the memory view. Memory views permit different partitions on memory access to be rapidly switched and imposed.

Intel VT-d can be used to provide protections from devices in a similar manner by associating page tables with specific devices to regulate access attempts from those devices.[2]

Each memory view defines a distinct protection domain and the *VMFUNC* instruction invocations in non-root operation (the VM guest) can be used to rapidly switch between predefined EPT-based memory views without incurring the overhead of a VM exit. To switch from one memory view to another without incurring a VM exit, the *VMFUNC*-calling code page in the active, source memory view must share that executable page with the destination memory view—the hypervisor is involved in this explicit sharing during memory view creation.

Another Intel VT extension adds the ability for EPT violations to be mapped into a guest VM Virtualization Exception (#VE, vector 20) instead of causing a

VM exit to the host or root operation. This allows OS-level view management logic to handle such exceptions in a manner semantically similar to a page fault (#PF). This provides the view management agent the ability to detect and respond to EPT permission violations with the latency of exception handling as opposed to accruing VM-exit latencies before reaching the view management logic. This makes it possible, for example, to vet access to specific pages in an efficient manner.

For further details, please consult Volume 3 virtualization material of the Intel® 64 and IA-32 Architectures Software Developer Manuals, Intel® Virtualization Technology for Directed I/O, Architecture Specification, and other related application notes and literature for additional details.

Architecture

This section describes the software architecture for our sensor protection methodology.

Solution Approach

The approach protects sensor data by applying memory access-control properties on that data. The access-control properties also enforce restrictions on software code that is allowed to read or write sensor data in memory. The sensor data is directed to an isolated area within the host memory so that it is protected. The memory protection is enforced using the processor capabilities (Intel VT-x and VT-d) and context-specific policies. Further protections can be enforced as the data traverses through the operating system stack as it is delivered to an application or application library that is trusted to operate on the sensor data. The protection model (TCB definition) for the sensor data as it is transferred through the OS stack is dependent on the type of sensor data, the software expected to consume that data, as well as potential touch points to the data en route to the authorized consumer software. Furthermore, sensor data may be protected by transforming the sensor data via trusted consumer software to output an abstracted or limited logical stream. For example, raw video of a person may be converted to abstract gestures by the software trusted to process the raw video; only the user's gestures are output, not the raw video .

"…protections can be enforced as the data traverses through the operating system stack as it is delivered to an application or application library that is trusted to operate on the sensor data."

Architecture Overview and Components

Figure 3 describes the key components in the architecture.

A sensor data processing software stack can be logically viewed as a hierarchical set of services provided by different components in the operating system.

Input/output to a device at the lowest level in a device stack is typically controlled through a set of bus-interface-specific device drivers (for example, USB), with the lowest driver (that is, port driver) maintaining device state synchronization, managing control and data flow to the device and providing basic I/O services to the upper layers in the stack. Higher layer stack components are responsible for managing class-specific sensor data (such as camera or audio stream over USB) and exposing this through APIs to OS

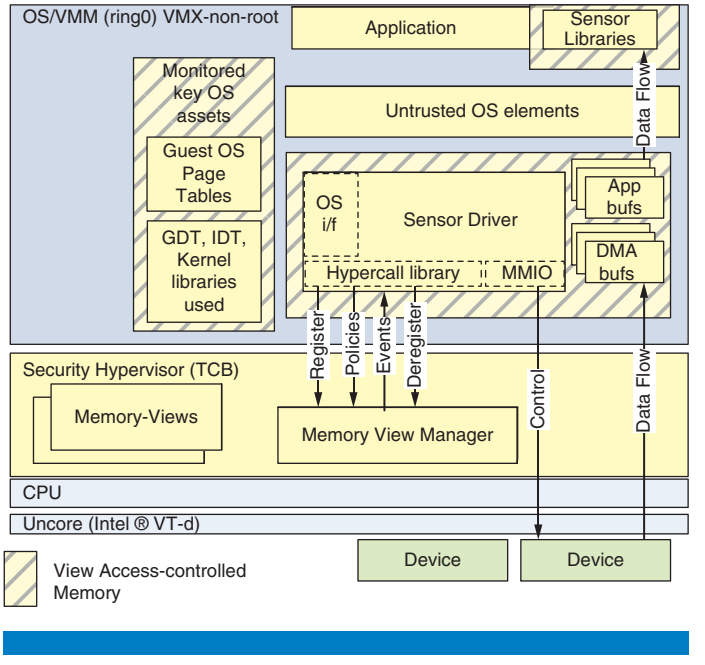


Figure 3: Sensor protection stack (Source: Intel Corporation, 2014)

"In our protection model we add a security hypervisor to provide management and access control over the memory regions of interest, which yields isolation of code and data for critical sensor services."

"Trusted (whitelisted, TCB) software components are mapped into EPT memory views such that code and data for these whitelisted components are protected from tampering by untrusted (non-TCB) OS/application components on the system."

services, which are consumed by applications or other software libraries. The OS provides core memory and interrupt management services. In our protection model we add a security hypervisor to provide management and access control over the memory regions of interest, which yields isolation of code and data for critical sensor services. We describe the function of each of these components below.

Security Hypervisor (VMM)

A security hypervisor is used to enforce runtime integrity checks for the CPU registers and protected memory. CPU register protection is enforced via Intel VT-x controls to track key CPU state changes such as control registers, Model-Specific Registers (MSRs), and Descriptor Tables (such as interrupts). Memory protection is enforced via the processor EPTs, (described in the "Background" section). Trusted (whitelisted, TCB) software components are mapped into EPT memory views such that code and data for these whitelisted components are protected from tampering by untrusted (non-TCB) OS/application components on the system. The use of EPTs for memory views has the following key properties:

- It enables the enforcement of safe memory monitoring policies by restricting memory accesses in an OS-compatible and scalable manner.
- The processor reports memory policy violations to the security hypervisor or to a trusted in-guest component, thereby mitigating circumvention attacks.
- It enables the enforcement of separate memory permission domains for a single OS, which in turn enables isolation of sensor drivers for efficient memory access control.

The hypervisor exposes a Memory Monitoring API for use by *Sensor Drivers* as well as *Sensor Libraries*. This API exposes operations allowing trusted components to:

- Register/unregister memory that contains the drivers code and static data sections. The hypervisor performs its own guest page table walks to verify that the driver image is whitelisted. Removal of memory protection is restricted to trusted memory view code.
- Register/unregister dynamic memory allocated by the Sensor Driver for protected DMA buffers.

Because the security hypervisor is a key element of the Trusted Computing Base (TCB) for this architecture, it is implemented with minimal code (for example, our research hypervisor is just 0.2 percent the size of a typical operating system). This can be accomplished with a minimal hypervisor, focusing on memory virtualization. Effectively using memory views allows inserting trusted programs in the device stack without increasing the complexity of the hypervisor.

"…the security hypervisor is a key element of the Trusted Computing Base (TCB) for this architecture…"

The hypervisor virtualizes relevant guest OS resources to ensure that memory view interaction does not add a large performance overhead. To ensure that code execution across memory view boundaries do not cause VM exits, the sensor driver code uses VMFUNC-based trampoline code pages to avoid VM exits on legal code-entry points into itself. Such a scenario might occur, for example, on a synchronous kernel callback when new sensor data is available in the protected DMA buffers. For asynchronous exits from a driver due to other device interrupts, an overloaded Interrupt Descriptor Table (IDT) is used to ensure the trusted driver code can save and restore state safely without VM exits. Effectively with such para-virtualized sensor components, no VM exits occur for "normal" data processing unless a memory violation occurs due to an attack.

"…no VM exits occur for 'normal' data processing unless a memory violation occurs due to an attack."

The security hypervisor is launched via a hardware root of trust for measurement to ensure that a trusted hypervisor is loaded on the system. Existing technologies such as Intel® TXT, Trusted Platform Module (TPM) support such static or dynamic (late-launch) measurement of hypervisors and OS software. Past effort in this space has been described in detail in an earlier article.[25] Additionally, the hardware root of trust can be used to provide evidence, or attest, that a specific measured hypervisor has been launched on the platform to support usages that need remote attestation of trust for the sensor data protected via the security hypervisor. Measurements providing evidence of the correct components being loaded at operating system initialization time and hardened using memory views provide assurances that the data originated from a given device on a given platform and has not been compromised by an unauthorized entity on the platform.

"…the hardware root of trust can be used to provide evidence, or attest, that a specific measured hypervisor has been launched on the platform…"

Sensor Drivers

The sensor drivers can be considered as a logical entity that interfaces with a sensor device to retrieve data into host memory, typically through a DMA operation. This data is then packaged and sent to upper layers in the stack in

a well-defined, interface-specific manner. The sensor driver may be separated into multiple discrete physical drivers, if the interface to the sensor device supports transporting different data types. One example of hierarchical sensor drivers is the USB interface and related protocols to support different sensors including audio, camera, keyboard, touch, and storage. In the case of USB, the port driver deals with DMA buffers and with mapping these to appropriate device descriptors for managing specific data flows for a logical device, while additional upper layer drivers manage the actual data flow pertaining to a specific sensor class (for example, video vs. audio vs. Human Interface Device [HID]).

"…ensures that unauthorized software is unable to modify the code/data within the driver that manages critical memory regions…"

In order to secure the sensor data flow, the lower sensor driver interfaces with the security hypervisor to protect the transfer buffers used to DMA the data from the device. This memory-view protection is applied in conjunction with self-protection memory views for the driver code and data segments. This serves two purposes: (1) it ensures that unauthorized software is unable to modify the code/data within the driver that manages critical memory regions for the sensor data; (2) additionally, it define boundaries for code regions, within the sensor protection TCB, which is allowed to access the memory-view protected sensor data. As well as protecting the DMA buffers for sensor data transfer, the sensor driver also protects the data structure hierarchy that points to the transfer buffer from the device perspective. This is illustrated in Figure 4.

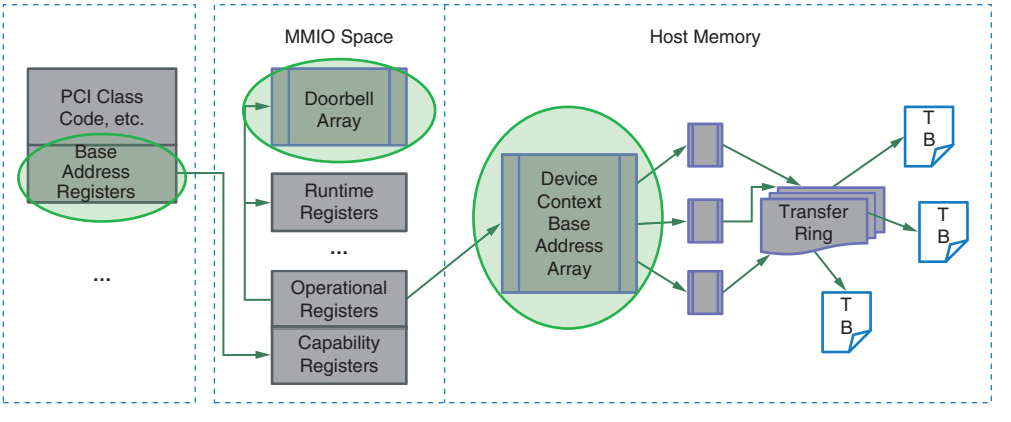


Figure 4: USB descriptors
(Source: Intel Corporation, 2014)

Figure 4 provides an example for USB descriptor mappings from the PCI configuration space, where the device lands, through the USB memory-managed I/O (MMIO) space and associated registers to the device context pointers in host memory and the transfer rings that will point to the transfer buffers used for DMA. As the sensor driver is defining and configuring the device for these descriptors, it can identity and protect the individual array entry for a logical device when the device is first enumerated and discovered through plug-and-play or bootstrapping the machine. Protecting the device

descriptors, transfer rings, and DMA buffers in this manner ensures that only the sensor driver in the TCB has access to the sensor data.

Once the data is received into the transfer buffer, the sensor driver can either collaborate with the upper layer sensor libraries to exchange the data using additional protected memory views, or alternatively, build a cryptographic channel with the recipient using a simple key management scheme to protect (encrypt or protect the integrity of) the data. The latter allows the protected sensor data to be released to an untrusted kernel component outside the TCB, and any tampering of the data can be detected by the recipient by validating the cryptographic data integrity. As one example, this cryptographic channel can be established by using a protected memory view shared between the sensor driver and sensor library to share a large random number, which can be used in conjunction with the platform root-of-trust information, to compute additional ephemeral keys (using well-known cryptographic primitives and algorithms (such as HMAC-SHA-256), for encrypting or protecting the integrity of the sensor data. Once keys are established, the sensor driver can encapsulate the data in a cryptographic channel, before releasing it (outside the TCB) to an untrusted component up in the stack. The cryptographic keys can be refreshed periodically, based on usage or time, in accordance with good cryptography practices defined by standards bodies such as NIST. In one instance, the data transfer may be simply copying the encrypted data to a memory buffer of an upper driver in the stack. The specific tradeoffs and approach taken to define a TCB that protects the sensor data between the sensor driver and sensor library may depend on the OS architecture, the ability to make OS source-code modifications, TCB considerations of the overall solution, and the type of sensor data and the number of touch points on the data by different intermediate services within the OS.

"…the sensor driver can either collaborate with the upper layer sensor libraries to exchange the data using additional protected memory views, or alternatively, build a cryptographic channel…"

Sensor Libraries and Data Consumers

The sensor libraries are another memory-view protected element within the TCB, which processes sensor data. Sensor library may refer to a sensor fusion component (such as geo-fencing of certain sensor data), a logical sensor abstraction (such as taking a raw camera feed and outputting gestures) or a native sensor consumption component (such as securing keystrokes to a browser for an ecommerce application). In all cases, the sensor library has some logical binding to the sensor driver, as it needs to receive the sensor data in a secure manner. If the data is protected using a cryptographic channel, the sensor library may need to collaborate via the hypervisor to generate cryptographic key material that is shared with the sensor driver. In one instance, a shared memory-mapped buffer may be constructed and protected, ensuring that the keys are only accessible to the sensor driver and library. Just like the sensor driver, the sensor library needs to perform self protection of code/data by calls back into the security hypervisor API to ensure untrusted software is not able to modify the sensor library in any way, as well as ensuring that the shared, protected memory buffers are only accessible from within the TCB code boundaries of the sensor library. Once encrypted sensor data is received by a normal untrusted channel into an unprotected buffer of the

"…the sensor library needs to perform self protection of code/data by calls back into the security hypervisor API to ensure untrusted software is not able to modify the sensor library in any way…"

sensor library, the sensor library can copy this data into a protected memory region before using the cryptographic keys to decrypt the data in place. Once the data is decrypted, the data can be passed to an authorized (whitelisted) application; such whitelisting and other polices can be delivered by a manifest method or, in our case, via existing McAfee products such as Site Advisor or e-Policy Orchestrator (ePO). Whitelisting is a method of separating known (trusted, whitelisted) applications from malware (known untrusted) applications, as well as unknown (uncategorized) applications. Measurement, signing, and reputation techniques may be employed to differentiate applications and their trust metrics. The sensor library can provide further wrappers to export simple but secure APIs, which may be natively consumed by an application. In this case, the application can be agnostic of the actual sensor data, instead relying on an off-device or cross-platform service (such as a secure cloud service) to authenticate the data. A study of this approach is described in a prior paper.[26] In another instance, the sensor library may be constructed as a dynamically linked library, which can be loaded directly into the application space to process secure data, as well as exposing a library of interfaces to secure the appropriate application memory.

"The sensor library can provide further wrappers to export simple but secure APIs, which may be natively consumed by an application."

Theory of Operation

In this section, we describe a theory of operation for a single sensor stream and how the data can be protected for this sensor.

On platform initialization, if a static root of trust for measurement is used, the platform firmware is measured before initialization by a hardware root of trust for measurement on the platform. The hypervisor may be measured by the firmware and initialized before the operating system is initialized, or it may be measured by the measured OS boot loader at an early stage during OS initialization before untrusted operating system components are initialized. The measurements captured at the different stages are recorded in the platform hardware root of trust such as a trusted platform module[23] or an alternate custom root of trust. The hardware root of trust may, during initialization, also enforce a whitelist of the firmware, hypervisor, and operating system loader allowed to initialize the platform resources during this trusted boot phase.

"The hypervisor may be measured by the firmware and initialized before the operating system is initialized, or it may be measured by the measured OS boot loader at an early stage during OS initialization before untrusted operating system components are initialized."

If the hypervisor is initialized before the guest operating system, it may choose to revoke access to certain devices before the appropriate whitelisted device drivers are loaded, initialized, and measured per the hypervisor policies to create the appropriate memory view to protect the device drivers assets in memory. During this phase the hypervisor may allow accesses to a given device's PCI configuration space but disallow accesses to the device-specific MMIO regions in physical memory so that the device cannot be maliciously initialized or configured by any unknown guest kernel code that may load as part of the untrusted device drivers loading into the kernel. Any failures should be logged and the device owner/administrator should be notified.

If the platform is initialized as expected (per the allowed measurements), untrusted operating system drivers can begin initialization and register with the

hypervisor using the control interface shown in Figure 3. These device drivers may be signed per operating system policies and may be additionally verified by the hypervisor per the hypervisor policies to verify the driver code (text) and data (read-only) sections in memory. The hypervisor can protect the physical memory assets for a verified driver via a memory view; it can also initiate monitoring for the virtual address mappings for the driver address mappings in the kernel address space.

"The hypervisor can protect the physical memory assets for a verified driver via a memory view…"

A measured driver may perform initialization of dynamic data (such as heap-allocated buffers) and allocate additional memory that can be assigned exclusive access to the driver via the assigned memory view using the hypervisor interfaces. Memory for these critical resources is typically pinned, so that the OS does not page out the memory in preference for another resource. The driver at this point can map the MMIO space and request the hypervisor, via a view-restricted control interface, to map the blocked MMIO space for the device into the drivers memory view with read-write permissions. At this point the device driver can continue with the initialization of the device per the required logic in the device driver. Note that the MMIO space for the device is still effectively unavailable (not present) to other kernel components.

The device driver also allocates and registers DMA targeted memory with the device so that the device can perform the required I/O to transfer data buffers specific to the device. Device drivers typically also allocate metadata and aggregation data structures such as ring-buffer pools and descriptor structures, which are protected in a similar manner. Any data received by the device driver can now be transmitted via the DMA controller directly into the memory view for the device driver where the driver code has exclusive read-write access to the data buffers. These data buffers may now be copied by the driver into other memory views protecting application code that the driver wants to share this data with, or be encrypted using ephemeral keys derived from the platform root of trust (described in the section "Architecture Overview and Components"), so that the data can be transferred to a remote platform or safely through the untrusted operating system stack to an application component that uses a similar approach to decrypt the data for further processing using memory isolated on behalf of the application component.

"Any data received by the device driver can now be transmitted via the DMA controller directly into the memory view for the device driver where the driver code has exclusive read-write access to the data buffers."

Applications that need access to the protected data must load sensor libraries that initialize in a similar manner as the device driver initialization described above. Also the application must receive access either to shared, protected buffers or the ephemeral keys (managed by the hypervisor) for the data exchange session with the protected device driver. Application code that decodes or decrypts the data in its protected memory view must handle the data carefully to avoid exposing it to untrusted, non-TCB elements in the application address space (or untrusted kernel components). Ideally, the application will process the data as needed and will use cryptographically protected sessions (such as via SSL) to send any sensor data to peer services it interacts with to continue to protect the data. Products and tools already exist in the marketplace that perform whitelisting of software components based

on integrity/signatures over the software, managed through centralized policy services. One example of this is the McAfee whitelisting tools and the ePolicy Orchestrator (ePO) for policy management. These tools are already available and widely deployed in the marketplace.

Conclusion and Discussion

"The approach outlined in this article is capable of hardening endpoint client software against both known and unknown software-based threats."

The approach outlined in this article is capable of hardening endpoint client software against both known and unknown software-based threats. The effectiveness of the solution is not tied to signature and heuristic-based detection mechanisms and does not require that threats be cleaned or otherwise made impotent by security software. Therefore, this approach can be effective against zero-day threats where signatures, heuristics, and remediation approaches are not yet available.

Existing approaches to system protection against attacks targeting sensors include system policy restrictions (such as whitelisting), file reputation, process protection, and behavioral monitoring. These fall short by either failing to proactively identify new malicious attacks, or applying heavy-handed access prevention to wanted applications, such as denying execution on all unknown programs. These lead to a poor user experience, but worse, leave the user's data at risk.

"The reduced attack surface will force malware to focus on specific attacks that are potentially easier to detect and categorize…"

We have shown that the use of secured alternative communication channels for transferring sensor data between the data producers and the data consumers eliminates traditional inspection and interception points for malware, and significantly reduces the available attack surface for the traditional human-input/sensor I/O stacks. The reduced attack surface will force malware to focus on specific attacks that are potentially easier to detect and categorize; it is possible that our approach could improve the effectiveness of signature- and heuristic-based threat prevention. Further, we assert that the secured delivery and protected consumption of sensor data has several advantages over signature and heuristic based remediation of threats:

- Reduced risk of false positives
- Improved performance over on access AV scanners
- Reduced footprint since AV engine components and signature can be eliminated or trimmed (for example, removal of keyboard logger signatures)

However, we do not propose that the solution is a replacement for traditional antimalware software. Instead, such an approach should be used in conjunction with antimalware, adding defense in depth. The principal goal of antimalware will be to help protect the approach outlined in this article, as well as to categorize, remove, and report on threats to the user, or via central management tools like McAfee's Enterprise Policy Orchestrator suite EPO.

Initial implementations of the approach have shown that the core approaches of restricting access to buffers, preventing changes to code sections and critical

data regions, and intercepting execution of critical APIs (hypervisor-mediated hooking) scale and perform well compared to equivalent protection in a traditional antimalware solutions. While these approaches can be implemented on other platforms, Intel platforms are optimized to reduce the impact of hypervisor-based protection mechanisms, and as a result, will perform significantly better on future Intel platforms.

The protection model that we have chosen successfully meets our high level goals:

- Input data is protected as early as possible in Ring 0 and transferred to Ring 3 via a secure alternative I/O channel. In so doing, traditional attacks against the sensor stack are completely defeated. In addition, the same hypervisor techniques that are used to collect and protect sensor data can also be applied to protect the alternative communication channel, preventing attacks against the sensor to consumer channel itself.
- Protection can be retroactively applied to existing software without code modification. Existing software can be further strengthened by preventing the addition of information-stealing hooks and other methods of attack by preventing memory changes—all of this without cooperation with or consent of the original software authors and thereby achieving the original goal of ease of implementation and policy enforcement.
- Performance and user experience is preserved using introspection capabilities provided by the new Intel instructions (VMFUNC, #VE).

In summary, protecting sensor data input, delivery, and consumption via hypervisor-mediated protection techniques provides a less-invasive, scalable, and effective approach, preserving the user experience, without adding additional dials for security configuration.

"…protecting sensor data input, delivery, and consumption via hypervisor-mediated protection techniques provides a less-invasive, scalable, and effective approach, preserving the user experience, without adding additional dials for security configuration."

Acknowledgements

We would like to recognize others who helped develop these concepts: Ray Askew, Sanjay Bakshi, Ernie Brickell, Prashant Dewan, David Durham, Ghayathri Garudapuram, Nathan Heldt-Sheller, Howard Herbert, Barry Huntley, Reshma Lal, Jason Martin, Mark Scott-Nash, Pradeep Pappachan, Richard Reese, Carlos Rozas, Manoj Sastry, Uday Savagaonkar, Paul Schmitz, Geoffrey Strongin, Salmin Sultana, Ulhas Warrier, and many others.

References

[1] Intel® 64 and IA-32 Architectures Software Developer's Manual Combined Volumes 1, 2A, 2B, 2C, 3A, 3B and 3C. Intel Corporation, 2014, http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html. Intel® Virtualization Technology requires a computer system with a processor, chipset, BIOS, virtual machine monitor (VMM) and for

some uses, certain platform software, enabled for it. Functionality, performance or other benefit will vary depending on hardware and software configurations. Intel Virtualization Technology-enabled VMM applications are currently in development.

[2] Intel® Virtualization Technology for Directed I/O; Architecture Specification; September, 2013, http://www.intel.com/content/dam/www/public/us/en/documents/product-specifications/vt-directed-io-spec.pdf.

[3] Choe, Eun Kyoung, Sunny Consolvo, Jaeyeon Jung, Beverly Harrison, and Julie A. Kientz, "Living in a Glass House: A Survey of Private Moments in the Home," Proceedings of UbiComp'11, Beijing, China, September, 2011.

[4] Boyd, Danah and Alice Marwick, "Social Privacy in Networked Publics: Teens' Attitudes, Practices, and Strategies," Oxford Internet Institute's "A Decade in Internet Time: Symposium on the Dynamics of the Internet and Society", September, 2011, http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1925128.

[5] Boyles, Jan Lauren, Aaron Smith and Mary Madden, "Privacy and Data Management on Mobile Devices," Pew Research Center, September, 2012, http://pewinternet.org/~/media/Files/Reports/2012/PIP_MobilePrivacyManagement.pdf.

[6] Bryant, Chris, "Biscuit tin keeps lid on Evonik's secrets," *Financial Times*, June, 2011, http://www.ft.com/intl/cms/s/0/010c3e80-a020-11e0-a115-00144feabdc0.html.

[7] Holehouse, Matthew, "iPads banned from Cabinet meetings over surveillance fears," *The Telegraph*, November, 2013, http://www.telegraph.co.uk/news/politics/10423514/iPads-banned-from-Cabinet-meetings-over-surveillance-fears.html.

[8] Nerney, Chris, "For shareholder meeting, Apple bans laptops, cell phones," *ITworld*, February, 2012, http://www.itworld.com/mobile-wireless/252716/shareholder-meeting-apple-bans-laptops-cell-phones.

[9] Xu, Zhi, Kun Bai, Sencun Zhu, "TapLogger: Inferring User Inputs on Smartphone Touchscreens Using On-board Motion Sensors," WiSec'12, April, 2012, http://www.cse.psu.edu/~szhu/papers/taplogger.pdf.

[10] Templeman, Robert, Zahid Rahman, David Crandall, and Apu Kapadia, "PlaceRaider: Virtual Theft in Physical Spaces with Smartphones," arXiv:1209.5982v1, September, 2012, http://arxiv.org/pdf/1209.5982v1.pdf.

[11] Kirk, Jeremy, "SoundMiner Android Malware Listens, then Steals, Phone Data," *PCWorld*, January, 2011, http://www.pcworld.com/article/217133/article.html.

[12] McAfee, "Attack: "Flame"," May, 2013, http://www.mcafee.com/us/about/skywiper.aspx.

[13] Spohn, Michael, "Know Your Digital Enemy: Anatomy of a Gh0stRAT," McAfee White Paper, 2012, http://www.mcafee.com/us/resources/white-papers/foundstone/wp-know-your-digital-enemy.pdf.

[14] Tarakanovk, Dmitry, "Big Brother," Securelist, May, 2012, http://www.securelist.com/en/blog/208193513/Big_Brother.

[15] Whitney, Lance, "Android malware uses your PC's own mic to record you," CNET, February 2013, http://news.cnet.com/8301-1035_3-57567430-94/android-malware-uses-your-pcs-own-mic-to-record-you/.

[16] Wilson, Charles, "Online 'Sextortion' Of Teens On The Rise: Feds," Huffington Post, August, 2010, http://www.huffingtonpost.com/2010/08/14/online-sextortion-of-teen_n_682246.html.

[17] Anderson, Nate, "Meet the men who spy on women through their webcams," ars technical, March, 2013, http://arstechnica.com/tech-policy/2013/03/rat-breeders-meet-the-men-who-spy-on-women-through-their-webcams/.

[18] Anderson, Nate, "Webcam spying goes mainstream as Miss Teen USA describes hack," ars technical, August, 2013, http://arstechnica.com/tech-policy/2013/08/webcam-spying-goes-mainstream-as-miss-teen-usa-describes-hack/.

[19] BBC News, "US Army: Geotagged Facebook posts put soldiers' lives at risk," BBC, March, 2012, http://www.bbc.co.uk/news/technology-17311702.

[20] Essers, Loek, "Researcher: Hackers can cause traffic jams by manipulating real-time traffic data," *PC World Magazine*, March, 2013, http://www.pcworld.com/article/2030991/researcher-hackers-can-cause-traffic-jams-by-manipulating-real-time-traffic-data.html.

[21] Diskin, Gal, "Virtually Impossible: The Reality of Virtualization Security," 30th Chaos Communication Congress, Hamburg, Germany; December, 2013, http://www.youtube.com/watch?v=GoipioWrzAg.

[22] Sailer, Reiner, Xiaolan Zhang, Trent Jaeger, Leendert van Doorn, "Design and Implementation of a TCG-based Integrity Measurement Architecture," Usenix Security Symposium, 2004, https://www.usenix.org/legacy/event/sec04/tech/full_papers/sailer/sailer_html/.

[23] Trusted Platform Module WG, "TPM Main Specification Level 2 Version 1.2, Revision 116," Trusted Computing Group; March, 2011, http://www.trustedcomputinggroup.org/resources/tpm_main_specification.

[24] Brand, Sheila L. "DoD 5200.28-STD Department of Defense Trusted Computer System Evaluation Criteria (Orange Book)," *National Computer Security Center* (1985): 1–94.

[25] Sahita, Ravi, Ulhas Warrier, and Prashant Dewan, "Intel Corporation—Protecting Critical Applications on Mobile Platforms," *Intel Technology Journal*, Advances in Internet Security Vol. 13 Issue 2, June 2009, p. 21.

[26] Sahita, Ravi and Uday Savagaonkar, "Towards a Virtualization-enabled Framework for Information Traceability (VFIT)," Insider Attack and Cyber Security 2008, pp. 113–132.

Author Biographies

Jonathan Edwards is a solutions architect at McAfee Labs in Beaverton, Oregon. He entered the security business in 1995 working for Dr. Solomon's Software in Great Britain. Jonathan worked mainly in the Windows desktop and server antivirus field, helping develop the first on-access scanner for Windows* NT at Dr. Solomon's and then led the VirusScan NT team at Network Associates/McAfee. Jonathan is the lead architect for the McAfee DeepSAFE* technology and a member of the McAfee Principal Architects Forum. Jonathan received an MS in Scientific Computing from University of Sunderland, and BS in Chemistry with Mathematics, from the University of Southampton. He can be reached at jedwards@mcafee.com.

Ken Grewal is a senior research engineer at Intel Labs in Hillsboro, Oregon. Ken received a BSc in Applied Physics from City University in London, UK and has worked in the computer science field for over 25 years, with the last 15 years at Intel. Ken's current research is focused on hardening the platform from malware attacks and providing Sensor and Data privacy. He can be reached at ken.grewal@intel.com.

Dr. Michael D. LeMay is a research scientist at Intel Labs in Hillsboro, Oregon. He received the BS degree in Computer Science from the University of Wisconsin-Eau Claire and the MS and PhD degrees in computer science from the University of Illinois at Urbana-Champaign. Michael has research

interests in antimalware techniques based on virtualization and control-flow monitoring. He can be reached at michael.lemay@intel.com.

Dr. Scott H. Robinson, principal engineer at Intel, received a BS degree in Electrical Engineering and Computer Science from Duke University and MS and PhD degrees in Electrical and Computer Engineering from Carnegie Mellon University. Besides a research interest in sensor data security and privacy, he has worked in the areas of control-flow integrity monitoring, ubiquitous computing, extensible processor architecture, and processor performance and tracing. He can be reached at scott.robinson@intel.com.

Ravi Sahita is a security architect and principal engineer at Intel Labs. His primary research interests are processor/platform approaches to mitigate computer malware and enable software (runtime) integrity. Ravi has designed new CPU intrinsics and collaborated with McAfee to develop McAfee DeepSAFE. Ravi has previously defined security capabilities for Intel® AMT (System Defense, Agent Presence). Ravi has authored several Trusted Computing Group (TCG) and IETF standards specifications. Ravi received his BE in Computer Engineering from the University of Bombay, and an MS in Computer Science from Iowa State University. Ravi holds over 50 issued patents and is the recipient of an Intel Achievement Award. He can be reached at ravi.sahita@intel.com.

Carl Woodward is a software architect at McAfee in Santa Clara, California. He received his MSc in Information Technology from the University of Nottingham, UK. Carl is a member of McAfee's Principal Architects Forum, one of Intel's patent committees and his primary research interests are Trusted Computing and Closed Operating Systems. He can be reached at carl_woodward@mcafee.com.

THE NEW PARADIGM: WHEN AUTHENTICATION BECOMES INVISIBLE TO THE USER

Contributors

Ramune Nagisetty
Intel Labs

Cory Booth
Intel Labs

Today's model for user authentication is based on an outdated computing paradigm where multiple users accessed a stationary system and the same applications on that system. Unfortunately this model has not changed while the devices, applications, and services have been through decades of iteration and revolution, with computing becoming more mobile, personal, and even wearable. New devices cater to multiple user needs and desires with hundreds of thousands of applications available at the swipe of a finger. The outdated model of user ID and password was not intended for this future and is a fundamentally flawed process for accessing multiple applications and services across many devices throughout the day as users currently do. This article describes a vision of the future where smart devices, wearables, and integrated computing will drive new authentication modes and schemes, with authentication ultimately becoming a background process invisible to the user.

Introduction

"Today's model for user authentication to gain access to computing devices and services originated from a decades-old scenario where many individuals shared a single stationary computing device."

Today's model for user authentication to gain access to computing devices and services originated from a decades-old scenario where many individuals shared a single stationary computing device. In this "many-users-to-one-computer" model where users primarily used the same application, it was necessary for users to identify themselves through a user ID and password. Amazingly, over the last fifty years this model has remained largely unchanged even while computing devices and services have endured multiple revolutions, with each revolution making computing more mobile and more personal:

- The mainframe evolved into the desktop computer
- The desktop evolved into the laptop computer
- The laptop evolved into the smartphone
- Standalone computing evolved to connected computing, that is, the Internet
- Connected computing evolved into social computing
- Now, smartphones are evolving into wearable computing, and embedded computing is finding its way into our homes (Nest*) and businesses (iBeacon*)

With each of these revolutions came new technologies, services, and ways in which users interacted with them, and yet the move to more intensely personal, connected, and mobile computing has not significantly changed the base user identification and authentication model. The method for

how users are identified and authenticated must adapt to the world of today while anticipating a more complex future where users access a variety of applications, content, and services across many computing platforms. Furthermore, the level of authentication required for different types of transactions must comprehend usage patterns and the necessity of security therein. A few examples:

1. Today users access innocuous applications like maps as well as higher security applications like banking apps on the same system. Clearly those activities should require differing levels of confidence in user identification.
2. As individuals we are asked for a user name and password multiple times each day, even when accessing the same system throughout the day.
3. Multiple users employ shared devices and services like Zipcar*, Redbox*, treadmills, and ATMs. How can identification schemes accommodate for users taking "ownership" of devices and services in a public space?
4. Wearable devices are changing the landscape that we live in. Imagine wearing Google Glass* and being prompted for a user ID and password. Clearly the fifty-year-old model is completely broken when we go to the next step of wearable devices where a keyboard is no longer the input device of choice.

More subtlety and intelligence is needed in user identification. It's not a 1:1 mapping, nor should it be, and new technologies should drive changes in the industry for how identification and authentication is architected and experienced by users. This article describes a vision of the future where devices and device/service ensembles will drive completely new user experience design, traditional authentication schemes are replaced by new ones, and authentication becomes a background process invisible to the user.

Current State of the Art

The life of today's user is a series of starts and stops; technologically mediated interruptions to daily routines versus technology and systems enabling users' lives. Sometimes the stops are a result of poor execution, but many more times it is a lack of integration of new technologies and system-wide integration of authentication solutions.

Let's consider a scenario of a user on his way to the airport for a relaxing vacation. In order to access the information on his laptop he may need to first enter a hard drive password, and then an operating system password (Figure 1), and then a password for the airline website (Figure 2).

In this case, none of the systems share authentication information with each other, and the user is prompted multiple times as he makes his way to the check-in screen. At the check-in screen he needs to pay for his checked baggage and is prompted for his credit card information (Figure 3).

Figure 1: User enters hard drive password and OS password, the first passwords of many that will be needed. (Source: Intel Corporation, 2014)

Figure 2: Finally, to check in on the airline's website, he enters yet another password.
(Source: Intel Corporation, 2014)

Figure 3: To pay for his baggage, he adds additional credit card information that needs to be verified.
(Source: Intel Corporation, 2014)

Figure 4: The User is reminded to pay his bill.
(Source: Intel Corporation, 2014)

Before he leaves his home he realizes that he needs to pay a bill that will come due before he returns from his vacation (Figure 4). He is prompted for a password to log in and pay a utility bill (Figure 5).

On the way to the airport, the user stops at an ATM and once again is prompted for a bank card and PIN (Figure 6).

He gets back in the car, heads to the airport, presents identification in order to check his luggage (Figure 7), and presents his identification one more time to pass through the security gate (Figure 8).

By the time the user has made it to the airport, he has run through a gauntlet of authentication processes (Figure 9 and Figure 10). This is a device, service, and ecosystem conundrum with various systems failing to relay information to each other. While extremely inefficient, it's also a source

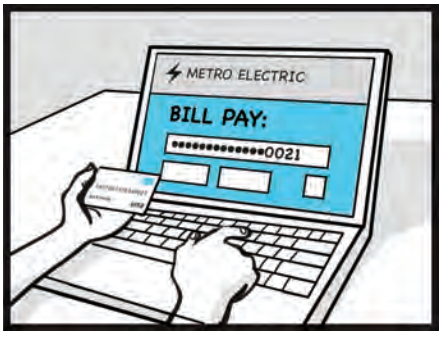


Figure 5: In order to pay his bill, the user once again enters his credit card information.
(Source: Intel Corporation, 2014)

Figure 6: The user is once again prompted to authenticate at the ATM.
(Source: Intel Corporation, 2014)

Figure 7: To check his baggage, the user needs to authenticate yet again.
(Source: Intel Corporation, 2014)

Figure 8: This time the user needs his confirmation number in order to begin the authentication process to check his bags.
(Source: Intel Corporation, 2014)

Figure 9: Finally the user presents his ID and boarding pass to enter the gate area.
(Source: Intel Corporation, 2014)

Figure 10: In the course of his journey to begin his vacation, he has authenticated himself into various systems no less than eight times.
(Source: Intel Corporation, 2014)

of frustration for users to have to repeatedly authenticate themselves. There has to be a better way.

While system integration challenges are at the root of the scenario described above, point solutions for replacing passwords with biometric information are gaining adoption. Most of these solutions address password pain points through the use of biometrics but none address the broader issues of overall systems integration. Some examples of current solutions follow.

"…point solutions for replacing passwords with biometric information are gaining adoption."

The Samsung Galaxy S III* was launched in 2012 with voice and face recognition. Widespread adoption of face recognition has been limited by latency and usability. For example, when using a phone, it's much more convenient to swipe a design as Android allows or quickly type in a PIN. Also, while driving or in social situations, holding the phone up to your face is not only awkward but potentially illegal and dangerous.

The Apple iPhone 5S*, released in the second half of 2013, was the first smartphone to integrate a fingerprint sensor as an alternative to typing in a password. However, it was hacked within a couple days of release.

Google released an update to Android late last year that integrated voice recognition for quick searches and general access to phone capabilities. The voice recognition can only be used for Google services and applications on Android. Additionally, in situations with background noise, voice recognition has a high failure rate, making for a frustrating user experience. Furthermore, nascent face and voice recognition technologies are easily spoofed, and require more refined techniques to be used reliably.

A more mature technology, palm vein authentication, has been used successfully in Japanese ATMs for several years. Palm veins are difficult to spoof because palm veins are imaged subdermally, and people rarely share images of their palm veins.

ECG (electrocardiograph), traditionally collected in a hospital environment with 12 leads and wet electrodes, has found its way into new form factors, such as an iPhone case. It is entirely possible that some form of ECG will be integrated into a wrist-worn device in the near future. The accuracy in nascent portable ECG technology is likely to be low but will certainly improve with sensor advancements.

While biometric sensors are becoming increasingly available and sensing analytics more prevalent, biometric sensing solutions are still suffering from lackluster adoption because of false rejects, latency, lack of overall system integration, spoofability, and in some cases corporate or government policies against collecting biometric data. We're a far cry from the future experience that users would desire where identification and authentication happens automatically or invisibly. The question is how to integrate a broad array of sensing technologies to support users throughout their daily lives, lives that entail a variety of low and high security identification needs. Solutions must accommodate base-level applications like games and infotainment as well as higher value services like

"The question is how to integrate a broad array of sensing technologies to support users throughout their daily lives, lives that entail a variety of low and high security identification needs."

banking, media, and even enterprise applications and security. Going forward, we'll outline some of the future usage scenarios that provide direction for sensing, identification, authentication, and system integration.

Future Scenarios

Sensor advancements have made it possible to incorporate individual and multiple sensors into devices we carry with us, interact with, and wear throughout the day. Accelerometers and GPS are already included in smartphones and some fitness trackers, and a collection of body-worn accelerometers can be used to identify gait. Wearable devices are incorporating sensors for heart rate, galvanic skin response, skin temperature and others. Analytics of sensor data can provide meaningful information based on things like location: is the person in a familiar or expected location such as home or office, versus an unfamiliar location, such as a hotel in a foreign country? Furthermore, "soft sensor" data such as the information available through a calendar app can be used to verify expected location and behavior, thereby increasing confidence in a person's identity. In conjunction with other soft sensor information, a combination of lower accuracy biometrics may prove sufficient for user authentication. For example, certain types of applications such as maps may not require the highest level of accuracy. Tradeoffs between accuracy and convenience can be made.

"In conjunction with other soft sensor information, a combination of lower accuracy biometrics may prove sufficient for user authentication."

Now we come to the "what if?" What if all of these innovations could be integrated into the multiple computing touch points of each user's life? If that happened, the previous user's trip to the airport would look much different.

Repeating the previous scenario of departing for a flight, once again the user starts in his home. He begins his day by getting dressed, putting on his shoes and smart watch, and checking his phone (Figure 11 and Figure 12). These devices provide a collection of both hard and soft sensor data that can be used within an authentication system. First, devices incorporating cellular,

Figure 11: Repeating the previous scenario of departing for a flight, once again the user starts in his home getting dressed. (Source: Intel Corporation, 2014)

Figure 12: Aggregation of sensed information enables the user to easily authenticate and check in to the flight. (Source: Intel Corporation, 2014)

Wi-Fi*, or GPS radios are able to determine location. In this case, the system is aware that the user is within his home based on past history. His smart watch, phone, and shoes also have the ability to collect bits of biometric data. His smart watch is able to detect his heart rate and ECG signal. His shoes, watch, and smartphone are equipped with accelerometers and gyros that are capable of gait recognition. Microphones incorporated into the phone and watch verify his voice. Individually these various biometrics are not strong enough to authenticate him. However, collectively, with the context of his location within his home and calendar information indicating he should be leaving for the airport, the available information is sufficient for checking the user and his luggage onto the flight.

The user realizes he needs to quickly pay a utility bill before departing for the airport (Figure 13). He pulls up his bank website and is able to authorize bill payment by using his voice (Figure 14).

Figure 13: The user realizes he needs to take care of an unpaid bill before he leaves for the airport
(Source: Intel Corporation, 2014)

Figure 14: He accesses his bank website and based on all available information including context and location, the user is able to authorize bill payment by using his voice.
(Source: Intel Corporation, 2014)

As the user departs his home, information continues to be collected and verified against what is expected. The user's location, including his route to the airport, the car he is driving (which is equipped with voice recognition), and soft data like calendar information, is available to his smartphone and his smart watch. The constellation of personal devices continuously gather information, verify against expected behaviors, and provide varying levels of authentication into systems based on the level of security required.

On the way to the airport the user stops at an ATM, which uses information available from the user's personal devices and augments that with real time 3D face recognition before dispensing cash (Figure 15). If the cash withdrawal is significant, an additional factor such as palm vein imaging or voice is requested.

Figure 15: User withdraws cash from the ATM and is only asked for additional identification information if a request to withdraw a significant sum is made.
(Source: Intel Corporation, 2014)

The user then proceeds to the airport, where his smart devices provide the authentication needed for him to drop off his luggage and proceed to the security gate (Figure 16 and Figure 17).

Figure 16: Information from his smart devices provides authentication needed to check in and drop off luggage. (Source: Intel Corporation, 2014)

Figure 17: The airport kiosk deposits a secure token into the smart watch. (Source: Intel Corporation, 2014)

At the security gate, the officer's screen displays a photo of the traveler (Figure 18), and along with the authentication information collected by the smart devices, there is enough confidence in the available data to allow the traveler to proceed to the gate (Figure 19).

Figure 18: A security guard reviews transferred identification information and only requests more if necessary. (Source: Intel Corporation, 2014)

Figure 19: The user's smartphone sends credentials to automatically identify him to the gate agent. (Source: Intel Corporation, 2014)

Next we explore a second scenario of renting public services and how emergency response systems may change in the future. A user is checking out a rental car for a short trip. She approaches the car and her smart devices provide information to initiate the authentication process (Figure 20). The checkout system uses her voice to augment the data available from her smart devices. She

Figure 20: The user checks out a rental car. Her smart devices provide the information to initiate the authentication process. The checkout system augments the available credentials with voice recognition. Based on the user's credentials, the user's paid services are loaded into the vehicle's console.
(Source: Intel Corporation, 2014)

gets inside the vehicle, which authenticates against her smart devices and pulls up her preferences profile along with her paid music services.

En route to her destination she loses control of the vehicle and crashes. As the paramedic arrives on site, he and his smart EMT system authenticate with the crashed vehicle and with the user's devices so he has any available information about the user and analytics of the crash sequence (Figure 21). The vehicle relays information regarding its trajectory and speed, and her smart devices relay vital signs and critical information like allergies.

Figure 21: The analytics of the crash sequence, as well as the user's identity, are recorded by the vehicle. The vehicle relays information regarding its trajectory and speed, and the user's smart devices relay vital signs and critical information.
(Source: Intel Corporation, 2014)

The EMT quickly sees relevant health information and is able to determine the cause of the crash was a seizure (Figure 22).

In a third scenario, we explore a casual social situation. A group of friends are having dinner together in a restaurant. Their smart devices communicate relevant information with the smart wine list, which makes recommendations based on their profiles and sharing of limited personal information (Figure 23).

Figure 22: The EMT, who is authenticated into emergency response systems, is able to make a diagnosis based on available information.
(Source: Intel Corporation, 2014)

Figure 23: A group of friends are having dinner. Their smart devices communicate relevant information with the smart wine list. The smart wine list makes recommendations based on their profiles and sharing of limited personal information.
(Source: Intel Corporation, 2014)

A smart bill of the future might automatically begin the authentication process as soon as a patron picks it up (Figure 24). When that user passes the bill to someone else at the table, the smart bill automatically authenticates the new user. Likely additional authentication will be needed because of the monetary transaction, so the bill prompts the user to say the tip amount verbally. The system uses voice as the additional authentication parameter while also allowing the user to complete the transaction in a more natural way, without getting out of the flow of the conversation with friends.

Figure 24: A smart bill automatically begins authentication as soon as a patron picks it up. A user passes the bill to another user for payment. The smart bill recognizes that another person has picked up the bill. Based on available credentials and the amount of the bill, additional authentication will be needed. The bill prompts the user to say the tip amount verbally.
(Source: Intel Corporation, 2014)

What Are the Barriers to Making This Future a Reality?

There are many barriers to making the future scenarios described above into reality. These barriers can be categorized as technology, business, and user experience barriers.

Technology Barriers

From a technology perspective, low power and small form-factor biometric sensors are needed. Some of the required sensors are already implemented in smartphones and smart watches today. Consumer-grade inertial sensors such as accelerometers, gyros, and magnetometers used in smartphones have already made rapid advancements in size, power, and cost. Real-time integration of sensor data to identify the user is also making strides (for example through things like gait recognition). High accuracy biometric sensors for vein imaging or ECG measurement are already available, but they need to be miniaturized to inexpensive, small form factors that can be integrated into power-constrained smartphones and smart watches.

A bigger challenge lies in combining all available information from hard and soft sensors. Entirely new algorithms must be created to intelligently combine all available information to identify and authenticate the user. Models for variable confidence in user identity must be created as well as models for acceptable risk and false positives and negatives. In the old "user id + password" model, it was mathematically straightforward to calculate the likelihood and risk of an imposter hacking a password through trial and error. In the future, new mathematical models to calculate the risk of false accepts/rejects from fused low accuracy sensor and context data will need to be developed and adopted.

"Models for variable confidence in user identity must be created as well as models for acceptable risk and false positives and negatives."

System integration is likely the most significant barrier to realizing this future. In the first scenario, today's travel and airport experience, the user had to authenticate multiple times because authentication information is not shared between devices and services. This is a problem that could be solved today through improved system integration.

"System integration is likely the most significant barrier to realizing this future."

One organization that is trying to address the system integration aspect is the FIDO (Fast IDentity Online) Alliance, which was formed to address the lack of interoperability among strong authentication devices as well as the problems users face with creating and remembering multiple user names and passwords. FIDO's mission is to change the nature of authentication by developing specifications that define an open, scalable, and interoperable set of mechanisms that supplant reliance on passwords to securely authenticate users of online services. However, while notable industry players such as Google, Mastercard, and Paypal have joined FIDO, companies such as Apple are conspicuously absent.

Business Barriers

Companies in many segments have woken up to the value of user data over the past two decades. From advertising models to big data analytics for optimization and innovation, user data is driving changes within organizations and is seen as the "new oil" fueling future revenues and new service creation. While vertical solutions providers such as Apple, Google, Microsoft, and Samsung define value within their product lines, the value to a broad spectrum of users will be in providing identification and authentication schemes across vertical ecosystems.

There are several difficulties with this from a business perspective:

1. Companies perceive lost value in opening up identification and authentication to competitors.
2. There is also concern that users will view that company's devices and services as less valuable as compared to competitors'.
3. Integration within a company's product lines is already difficult and costly. When considering, then, integration across multiple industry partner's solutions, the difficulties may be seen as insurmountable and the financial outlay unacceptable. Companies would surely like to pass along the expense to consumers with consumers loath to pay it.

User Experience Barriers

There are many different ways to create a future where authentication becomes invisible to the user, and reducing the barriers of using multiple platforms and services is a very clear and easily articulated user experience value proposition. Implementation of such a future will greatly advance or limit user acceptance as seen in the scenarios above. One implementation possibility seen in parts of Europe, China, and other locales is constant surveillance, where cameras monitor important public places and transportation thoroughfares. Advanced video technologies could be employed as a constant source of user identification. This type of Orwellian future is coming closer to reality, but is unacceptable in some societies. In addition to societal and governmental desires and constraints, recent events like the NSA leaks are resulting in governments and people rethinking how personal information is used and how it is disseminated. In the future scenarios described here we deliberately focused on the user's personal devices, devices that the user presumably has control of, as a multipronged source of identification and authentication. In that multipronged model, external sources such as cameras could be employed as well, but they are not the only source or barrier to bringing this usage model to market. The authors of this article believe the most effective way to enable this future is to ensure:

"…we deliberately focused on the user's personal devices, devices that the user presumably has control of, as a multipronged source of identification and authentication."

"The user has visibility and control of personal data, how it is used, and whether it is shared or not. This will enable users to trust the system; a foundation to the future we advocate."

- Security models adapt to a multi-input, variable model of user identity.
- The user has visibility and control of personal data, how it is used, and whether it is shared or not. This will enable users to trust the system; a foundation to the future we advocate.
- Businesses adapt to enabling authentication across ecosystem boundaries

Conclusion

The current state of user authentication, largely based on a decades-old model, has not evolved to today's world of personal, connected computing. Several scenarios were used to describe a vision of the future, where hard and soft sensor data are combined with context to seamlessly authenticate users into devices and services. Different services inherently require varying levels of confidence in the user's identity, therefore the overall system allows for different levels of authentication.

System integration poses the largest barrier to adoption. While the user is a clear beneficiary, both real and perceived issues related to collection of personal data and privacy need to be addressed. Furthermore, an improved authentication scheme will rely on many companies working together to create and adopt standards. The return on investment from a business perspective is difficult to quantify, especially with so many forces at play.

Author Biographies

Ramune Nagisetty is a principal engineer in Intel Labs. She currently leads research in wearable computing. Ramune earned a BSEE from Northwestern University in 1991 and an MSEE specializing in solid state physics from the University of California, Berkeley in 1995. She joined Intel in 1995, where she spent ten years working in device physics and process technology for Intel's Logic Technology Development group. She was the lead engineer for Intel's world class 65-nm technology, taking it from pathfinding through manufacturing transfer. From 2006 through 2009 she was the Director of Intel's Strategic Technology Programs, where she was responsible for several technology programs that report directly to executive management. She has eight technical publications and four issued and three pending patents related to device physics, high performance process technology, and wearable usage models. In 2008 she received the Emerging Leader Award from the Society of Women Engineers. She can be contacted at Ramune.Nagisetty@intel.com.

Cory J Booth is a principal engineer in Intel Labs, User Experience Research. Cory leads a cross-disciplinary team focused on defining future wearable user experience strategy and building prototypes to prove out those futures through iterative user experience design and research. Cory leads UXR initiatives and teams and works closely with ethnographers, design researchers, market researchers, technologists, and strategic planners to guide direction of future Intel platforms and services, investment strategies, and development of novel intellectual property. He has one patent and 16 undergoing the patent filing process. Cory holds a Master's of Science in Experimental Psychology and Human Factors and Bachelors in Psychology/Computer Science. Prior to joining Intel, he worked as a usability and user experience consultant for Usability Architects, Inc. After joining Intel in 2000, he worked in various internal design and user experience research groups, which enabled him to develop a multidimensional background including experience in: B2B solutions, web portals, factory automation systems, digital home and "smart TV" platforms, wearables, strategic planning, and managing teams of UX practitioners. Cory has led Intel in creating usage roadmaps to help define Intel strategic direction and product plans; now an Intel-wide adopted planning practice. He can be reached at Cory.J.Booth@intel.com.