Shoulder Surf Sentry: Secure and usable observation-resistant mobile computing in front of peering eyes

A Dissertation Prospectus
Michael Mitchell

1 Introduction

Shoulder surfing is becoming a concern for mobile computing. People are more frequently computing in public, at insecure places such as coffee shops, airports, and many other locations. Mobile devices are becoming more capable, enabling more advanced types of computing activity. Often, this can involve accessing data that includes private personal elements.

This behavior risks exposure by on-screen observation to a bystander. Personal information exposure can increase the risk of personal, fiscal, and criminal identity theft. Dependent upon the nature of the information, access of sensitive documents can lead to business losses, government espionage, and other forms of cyber terrorism [12, 13, 14].

This problem is getting worse every day. Surveillance is quickly becoming a part of everyday life in our society. Security cameras are everywhere and mobile devices are becoming increasingly capable in their ability to take pictures and record video. The increasing trend toward wearable computing devices will only serve to make this matter worse. Finally, criminals are getting more sophisticated visual analysis tools; partial screen capture, image reconstruction and OCR technology.

1.1 The threat is real

Exposure of sensitive information on-screen to bystanders is a real threat. In a recent visual data survey of IT professionals, 85% of those surveyed admitted there have been cases when they were able to see unauthorized sensitive on-screen data [1], 82% admitted that there have been cases where their own sensitive on-screen data could be viewed by unauthorized personnel [1], and 82% had little or no confidence that users in their organization would protect their screen from sensitive data exposure to unauthorized personnel [1]. These results are consistent with other surveys that show 76% being concerned about people observing their screens in public [2], and 80% admitting that they have attempted to shoulder surf the screen of a stranger in a public location [3].

Modern working and computing trends are serving to further increase this threat. Mobile devices are quickly replacing desktops with mobile device sales now accounting for over 73% of annual tech technical device purchases [4]. Employees more frequently take their work with them on the go; by 2015, the world's mobile worker population will reach 1.3 billion [5]. This is highest in the US, where more than 80% of the workforce continues working when they have left the office [6] and figures suggesting that 67% of employees regularly access sensitive data outside of sensitive environments [2]. Advances in screen technology are further increasing the risk of exposure, with many new tablets claiming near 180 degree screen viewing angles [8].

1.2 The dangers are everywhere

Visual exposure of sensitive information in the form of observation-based attacks can come in many forms. Mobile devices with cameras are nearly ubiquitous. There now exist more than 3 billion digital camera phones in circulation [4]. These devices are quickly becoming more capable with newer models capable of capture at over 40 megapixels and over 10 times optical zoom under $100 [7]. Visual exposure can also be captured by one of the billions of closed-circuit security devices. These high-resolution and often insecure cameras are everywhere, especially in major metropolitan areas; for example, figures suggest the average resident of London is captured on CCTV over 300 times every day [9]. Finally, but no less threateningly, sensitive data can be exposed by simple human sight.

Observation-based attacks can also be much more complex. More sophisticated tools and systems have been developed to capture and exploit private user data. Partial images can be merged, sharpened, and reconstructed, even from reflections. Optical Character Recognition (OCR) is becoming much more capable, with now over 40 years of innovation. Offline and cloud-based OCR solutions are highly accurate with only a small percentage of error in recognition. Embedded OCR solutions are cheap and capable even on low-end hardware devices [10].

Personal information exposure can also make other attacks possible. The capture of only a small number of personal information elements can greatly increase risk of other threats including social engineering attacks, phishing, and other personal identity theft threats.

1.3 The consequences can be severe

Observation-based information leaks can lead to significant personal and business loss. Recently, the publication of an S&P 500 company’s profit forecasts was leaked as a result of visual data exposure. The vice president was working on the figures on a flight, sitting next to a journalist [4]. In a different case, British government documents were leaked when a senior officer fell asleep on a train, permitting another passenger to photograph sensitive data on-screen [11]. In another case, the private details Bank of America clients were leaked through observation-based

attack. This information was captured by security cameras through the bank's windows [12]. In yet another case, sensitive personal information relating to Prince William was captured and published as a result of on-screen exposure to a bystander [13].

The risk of loss from shoulder surfing is also hurting business productivity as well. Figures show that 57% of people have stopped working in a public place due to privacy concerns and 70% believed their productivity would increase if they felt that no one could see their screen [2].

2 Existing solutions

Related works in securing against observation-based attacks is focused primarily on securing the act of authentication. They can be generalized as augmentation or replacement of password entry mechanisms.

2.1 Password managers

Perhaps the most common method of securing against an observation-based attack is the use of the password manager. These are software tools that allow the user to select a predefined username and password pair from a list for entry into the login fields [14]. Typically a master password is used to unlock the password vault to provide user security against loss. This also allows a user to use different passwords for different applications without the need to each of them individually.

Built in password management is now commonplace in nearly all web browsers including Google Chrome, Firefox, Internet Explorer, and Safari. Other typical features include automatic form filling, cloud-based storage, and synchronization. Commercial applications such as LastPass [15] and 1Password [16] extend this model with cross-platform and cross-browser support.

2.2 Hardware-based authentication

Other related work involves external physical devices to supplement or replace password-based authentication. They modify the problem from being strictly recall based (something you know) to possession based (something you have). Many different specialized pieces of hardware have been developed for this purpose, but they all conceptually function as a digital key to unlock another device. The simplest solutions do not require any connection or modification to the existing system and rely on the user to read a hardware display and enter the relevant data to authenticate. Other techniques utilize specialized USB dongles [17], audio jacks [18], short

range wireless communication using NFC [19], or Bluetooth connections [20] to connect to the authenticating machine.

The security credentials transmitted or manually entered by the user from these devices can be static passphrases [17], synchronously or asynchronously generated cryptographic tokens [18], or mechanisms that public key cryptographic challenge response techniques [17, 19] to that do not require the device to directly reveal the security token. Keypads or other input methods can also be used to further enhance the security of these hardware tokens by requiring the user to enter an additional pin or passphrase.

Closely related to hardware-based security token solution is device-based authentication. Commonly known as dual form, two-factor, or multi-factor authentication (MFA) [21], these techniques offer a secondary layer of authentication over strict hardware solutions. In addition to password entry (something you know), MFA supplements this with a hardware device (something you have), typically being a users smartphone. When the user attempts login to a service, a secondary, time-limited challenge response is sent from the service to a pre-registered device or address owned by the user. This message is sent via an alternate communication channel such as SMS message, e-mail, or through an application installed on the user's device. The user must enter this time limited, one-time security token in addition to their password to authenticate for the session.

2.3 Graphical passwords

Another technique to help guard against information leaks from visual attacks is the use of graphical passwords or Graphical User Authentication (GUA) [22]. Such techniques remove the alpha-numeric password from the equation and replace it with the use of series of images, shapes, and colors. Common techniques present the user with a series of human faces that must be clicked in sequence [23], object sequences as part of a story [24], or specific regions within a given image that must be clicked in sequence [25].

While these techniques may help to improve the security of password entries under specific circumstances they may actually serve to increase the risk of shoulder surfing observation. It may take an observer only a few login sessions to capture the image sequences required for authentication.

2.4 Biometrics

Biometric authentication mechanisms can be generalized as changing or augmenting password entry (something you know), with a feature unique to your personal biology (something you are). There are many inherent physiological characteristics that are sufficiently unique to identify and differentiate one individual from another. The most commonly used of these biometric identifiers

include contours of the fingerprints [26], iris and retinal configuration of the eye [27], and geometries the face [28] and hand [29]. Behavioral characteristics, in contrast to biometric identifiers, including keystroke latency [30], gait [31], and voice [32] can also be used for authentication purposes.

This authentication typically requires additional hardware support in the form of fingerprint scanners, retina scanners, or brainwave detection devices. Notable exceptions are facial feature detection and voice recognition, which can be performed using only the standard digital cameras and microphones present on nearly every mobile device in circulation. Primitive systems using facial geometries or voice recognition were easily tricked by photographs or audio recordings of the user, though they have improved significantly in more recent designs.

There are many privacy concerns for biometrics especially for trivial authentication purposes. By design, biometric feature are something that is unique to the user, and not easy to change. In the case of security loss, an attacker permanently gains access to all other services that identify the user by that specific physiological identifier.

2.5 Gesture-based authentication

Closely related to both GUA techniques and biometric solutions are gesture based authentication techniques. This method allows the user to perform specific tap [33], multi-finger presses [34], or swipe sequences on-screen [35] to represent a password. The user essentially is able to draw a picture to authenticate, or ‘connect-the-dots’ in a grid in a specific order or pattern. Gesture-based solutions can offer enhanced security over traditional GUA methods due to variations of different users’ finger features and characteristics. It is also less invasive than more traditional biometric methods since these methods still provide a mechanism to change the login credentials. This method was primarily developed for convenience since password entry using small screened mobile device keyboards can be difficult.

However, screen smudges can also leave evidence of the swipe sequence requiring the user to constantly wipe the screen to remove the residual evidence of the authentication pattern [36]. In addition, similar to the visual inadequacies of graphical passwords, a visual observer may be able to learn the tap, click, or swipe gesture after a single viewing, recordings can be re-played as many times as necessary to deduce and memorize the sequence. Excessive caffeine consumption, certain physical conditions, or side effects of other medications can cause jitters or inadvertent pausing and make such techniques less accurate or even frustrating for practical use for some individuals.

2.6 Cognitive challenges

Other techniques have attempted to make games of the authentication procedure [37]. Instead of a single password or phrase, these techniques utilize challenge response questions and use of cognitive tasks to increase the difficulty of the login session [38]. In method such as these, an onlooker would likely need to observe multiple login sessions to gather sufficient information about the user to be able to impersonate the user and gain unauthorized access.

The addition of timing constraints between subsequent tasks can further enhance the difficulty for an individual other than the user to trick the system and gain unauthorized access. These mechanisms can also be used in conjunction with many other authentication techniques discussed here.

2.7 Obfuscation and confusion

Other techniques have attempted to remedy the shortcoming of password based authentication through obfuscation and confusion to a visual observer. Such systems are essentially security enhancements to other authentication techniques previously mentioned notably GUA and cognitive challenge based methods. They utilize the hiding of cursors [39], confusion matrices [40], and recognition [41] rather than recall-based methods to trick and confuse onlookers.

Instead of the user directly clicking on a series images within a grid, the challenge can be changed to asking the user a series of questions of whether or not they can see one or more of their graphical password elements on a given screen. This serves to further complicate the challenge of an attacker to know which elements are the actual components of the user's graphical password. Another method is to use a field of randomly moving cursors during the authentication process. From their own mouse movements, the user can tell which cursor is genuine and ignore all the others. This can dramatically increase the difficulty on an observer to see which sequence of images was clicked.

2.8 Alternate sensory inputs

Additional work has been done utilizing other biological sensory inputs to replace or augment password-based authentication. These systems can address two separate parts of authentication process, the cue to the input, or the actual input itself.

In the first case, the additional sensory input serves as a non-observable instruction or hint to the required passphrase entry. These systems utilize audio direction [42] or tactile and haptic feedback from the vibration motors on devices [43] to provide the user the appropriate cue to the necessary response. The user then responds with the phrase corresponding to the cue using traditional input methods.

In the second case, the auxiliary sense serves as the input mechanism itself. These systems extend GUAs by requiring sequential graphical inputs but use mechanics like eye tracking, blinking and gaze-based interaction for the user to input the graphical sequence [44]. Systems have even demonstrated the capability of using brain waves brain waves for this task; a user may only need to think a specific thought to authenticate with a system [45]. These methods are also useful alternatives to authenticate for people with visual or audio sensory disabilities [46].

2.9 Digital Communication Channel Protection

Many protocols and systems have been developed also handle other aspects of privacy oriented attacks through the encryption of the digital communication channel. Transport Layer Security and Secure Sockets Layer can enhance security by providing session based encryption [47]. Virtual Private Networks can be used to enhance security by offing point-to-point encryption to provide secure resources access across insecure network topologies [48]. Proxy servers [49] and onion routing protocols like Tor [50] can add extra privacy by providing obfuscation of location, and anonymization of IP addresses.

Many other solutions have been developed to enhance security and privacy at the browser level. Do Not Track requests be can included in HTTP headers to request the web server or application to disable its user and cross-site tracking mechanisms [51]. Many browser extensions and plug-ins exist to block advertising [52] as well as analytics, beacons, and other tracking mechanisms [53]. Specialized systems can also alert the user when specific privacy elements are leaked [54] and prevent the transmission without explicit user permission [55].

2.10 Existing solutions inadequate

However, despite protection from many attacks, the visual channel is still open. A limited number of tools to are available to obfuscate sensitive data other than the act of authentication. All existing tools developed for encryption of data are not originally designed for such purposes. The exposure of personal data on screen remains unaddressed and the potential for loss from shoulder surfing observation remains a real threat.

3 Shoulder Surf Sentry

In response to the increasing threat of visual information exposure from shoulder surfing, the prevalence of mobile capture and surveillance devices, and the lack of existing solution we present the Shoulder Surf Sentry: mobile protection from observation-based attacks. The system provides proactive protection for a user to access sensitive personal information in insecure environments without the fear of visual privacy leaks.

The Shoulder Surf Sentry seeks to address three issues: (1) prevention of sensitive data (formatted in different ways) from being displayed on screen, (2) provides tools for users to enter sensitive data in insecure environments, and (3) maintain functionality, legacy compatibility across applications.

We define the threat model as passive, observation-based attacks. These attacks can come in the form of captured video or physical observation from a human. We assume the attacker can observe both the screen of the user as well as any touch sequences the user may make on screen or using physical buttons or keyboards. We also assume absence of an active attack as the observer cannot directly influence the user in any way.

Development of such a successful system to provide these goals must overcome several difficult challenges. Such challenges include identification and quantification of privacy, system architecture and design elements, and effectiveness evaluation.

The system will be designed based on data gathered from human subject studies. This must address and quantify how people compute differently in public and in private. The system prototype must find appropriate ways to intercept the display for sensitive data, handle sensitive data in different formats, provide a sufficiently usable user interface, and handle legacy compatibility across applications. The evaluation must demonstrate effectiveness on commonly used applications on different devices and manufacturer interfaces.

3.1 Challenge 1: Identification & Quantification

The first general challenge this project needs to overcome is to identify and quantify the threat of shoulder surfing. This step will involve identification sensitive data, the quantification of different behaviors in public and private settings, analysis of perception of privacy, as well as the identification secure and insecure computing environments. The general approach based on recall-based surveys via web interface, with raffle-based gift cards as rewards for participation gathered from FSU students and craigslist users.

Challenge 1a: Human Subject Survey-based solutions

Opinions and perception of privacy require an inherent level of human interaction; they cannot easily be collected or deferred from the system logs or Internet histories. However, a very limited number of such surveys involving human subjects have been conducted in the computer science discipline. An even fewer number of such surveys exist that focus on modern mobile usage and privacy concerns. Existing data sets are largely unavailable to privacy concerns and the possibility of sensitive information leaks from survey participants. In addition, the significant changes in the mobile environment in the past few years would make all but the very most recent surveys irrelevant.

Research involving human subjects is itself quite difficult. Permission for such surveys requires approval from Institutional Review Boards (IRB) and human subject committees. These applications are often overly complex and thorough, with documentation required down the finest detail. In addition, human subject research also requires investigators to receive special training, and can take months for the approval process to be finalized. The addition of privacy oriented issues and the potential for loss from participants can serve to make this process even more challenging to receive approval.

Challenge 1b: Participant solicitation

Once the questionnaires have been fully developed with every aspect of human subject interaction documented and IRB approval received, the next challenge is solicitation for survey participants. The general recruitment options include flyer-based methods, verbal solicitation, and mass emailing. Participants must also be sufficiently motivated to take part in the survey. The compensation must be sufficient enough to solicit participation, without excessive motivation to for participants to cheat. This could result in undesired data redundancy, or worse yet, the inaccuracy of collected data. The two general options for consideration are to offer a small reward for every participant, or to use a raffle-based solution with larger prizes.

A combination of flyer and mass mails was selected as the best choices for recruitment for the survey and a raffle-based solution with $1,000 worth of $25 Amazon gift cards was chosen as the appropriate compensation for survey participants.

Challenge 1c: Participant Demographics

College students represent a limited subset of all mobile users; college students at Florida State University most certainly represent an even smaller subset of users. The behaviors, locations for computing, as well as application usages could potentially be quite different from that of the general population. In addition it is reasonable to suspect their general opinions privacy, as well as usage of privacy connection tools could also be significantly different.

Thus, expansion of the survey beyond Florida State University students was necessary to capture more generalized usage and privacy opinions. Possible expansion options included solicitation for participants on craigslist, or using crowd-sourced services such as Amazon mechanical Turk, oDesk, or CloudCrowd. Crowd sourced solutions involve paying participants a small compensation for taking the time to complete the survey. Instead, it was decided use the same raffle-based option with Amazon gift cards, and recruit participants on craigslist in the top 10 most populous regions of the country.

Challenge 1d: Data Analysis

Hundreds of survey participants completing surveys with over one hundred questions generate a large amount of data. Normal, out-of-the-box statistical analysis toolkits on datasets of this size are difficult and tedious to use, if they can be used at all. Significant pre-processing and automation is therefore required to efficiently process the data. Custom tools need to be developed to identify and extract interesting trends, and to perform statistical significance tests with datasets of this magnitude.

3.2 Challenge 2: System Design

The second major challenge category for this research project is system design. Operating systems can present sensitive data in many ways; these methods must be identified before a system can be prototyped. In addition, there are many locations within the operating system and software structures that can be used to intercept display requests; these must also be identified for an effective system to be developed. The generosity of the solution across hardware devices, operating system versions, and vendor variants are also important in the design of the system. The design must also incorporate a method for the user to override the system else the possibility for loss of functionality would be increased.

The general approach will employ the semantic transformation of data. Sensitive privacy elements will be mapped to benign aliases and not permitted to be displayed on the screen. The prototype will leverage the open-source Android operating system to ensure that all possible levels of the system are available for manipulation and transformation of private data elements. The prototype will also leverage existing designs for password vaults and auto-completion techniques found in existing systems.

Challenge 2a: Data representation

The first sub challenge in the area of system design is the issue of data representation. Data fields themselves can be represented in multiple formats. Names can be represented as combinations of first, last and middle initials, e.g. Michael Mitchell; Michael J Mitchell; Mitchell, Michael J. Accounts and Social Security numbers can be represented using different spacing and or hyphenation schemes, e.g. 123456789; 123-45-6789; 123 45 6789.

Orthogonal to this problem, the operating system can render these elements in multiple ways as well. Both native applications and WebKit (browser) have different toolkits for rendering different widgets on screen. The simplest and easiest cases to handle will be basic TextView labels and lists. More complex interaction will be necessary to handle user input text boxes, especially those that utilize type enforcement to restrict input. It is also possible, although less likely for navigation widgets such as buttons, overlays, and other user interface elements to

contain personal data. Graphics, images, and other non-standard rendering techniques could also contain private data that should not be displayed on screen.

Challenge 2b: System Architecture

The second sub challenge and system design will be the overall system architecture. Application-based solutions are functional only under a certain select few cases. Custom e-mail applications or browsers could offer protection for task specific usages. However, such solutions interfere with user preference and functionality by restricting them to use a specific tool for a specific task. In addition, statistics show mobile usage trending away browsers and toward more general application usage.

A general-purpose solution that will retain functionality across applications will mandate system level changes. There are two possible options to provide these changes: custom system firmware images (ROMs) or code injection frameworks such as Android Xposed modules or Cydida Substrate.

By utilizing a custom system firmware image, complete control of the operating system is provided. The full source is available for modification. In addition, ROM-based solutions can offer a more unified testing environment. However, the changes would be restricted to device-specific builds; only hardware for which the source is explicitly built for would have access to the modified system. This also limits user preference by restricting use only for specific system image. It would also require regular maintenance, and would break vendor over-the-air (OTA) update functionality. Finally, ROM-based development is tedious, and much slower than more application-based options.

The alternative to custom firmware is to utilize a code injection framework like Android Xposed or Cydida Substrate. These options offer more streamlined development as standard user application development tools can be used. In addition, these modules can be more easily deployed since they can be distributed as applications. Since the system image is modified rather than replaced, users are free to make their own selection of system firmware, and manufacturer OTA updates can retain their functionality. These solutions however would suffer from a slightly restricted amount of control, and would be more difficult to test on a wide range of devices.

Challenge 2c: Secure User Input

The third sub challenge in the area of system design is to provide a mechanism for the user to securely enter private data in an insecure environment. The closely related, this issue is largely orthogonal to the mechanism that is required to prevent the system from display private data. Overall, all mechanisms that are used for user input can be generalized under the umbrella of an

Input Method Editors (IMEs). IMEs cover all input mechanisms including hardware and software keyboards, gestures and handwriting inputs, as well as speech recognition. To offer the user a mechanism for securing input, for options are available: (1) custom, secondary IME, (2) replacement of the default IME, (3) overlays or alternate input channels, or (4) hash tags, keywords, and macro expansions.

The first and perhaps easiest of these methods is to create a custom IME. This essentially would serve as an alternate keyboard that the user would select when they desired to input secure personal elements. This would be the most straightforward with regard to development as custom IMEs are essentially user applications with elevated privilege to offer input across applications. This method would be device, vendor, and platform-independent and retain its functionality on modified manufacturer interfaces including HTC Sense and Samsung TouchWiz. Additionally, since a custom IME is just an app, it offers ease of deployment, install, and updates.

However, such methods may prove to be hard for the user to use in practice. The process of changing from IMEs is unintuitive and requires the user to enter the settings menu each time they desire to switch to from one to another. In addition, use of the custom IME would lose functionality offered in default keyboards including swipe based inputs, emoticon support, auto correction, and custom dictionaries. Finally, special consideration would be needed for aesthetics; such methods natively are dated looking and not nearly as polished as flagship solutions.

The second method for providing secure user input would be to replace the default IME, or LatinIME as it is known in the Android open source project. This method would offer the most complete and mature code base with every feature the user would expect in a standard keyboard including swipe based inputs, emoticon, auto correction, and custom dictionaries. In addition, it would be familiar for the user to use and would not require switching back and forth between IMEs when input of secure personal elements is desired.

This method however is much more complex in terms of development. As it internally provides many of the actual input APIs itself, it cannot be developed using standard Android development toolkit. Instead, this method would require a complete from source build resulting in significantly increased development efforts and build time. In addition, this method would be newly impossible for an end-user to actually install as careful efforts would be required to remove and replace the existing Latin IME. It functionality issues may also prove to be challenging with conflicting dictionary providers and other more general API constructs. Finally, this option would be restricted only to AOSP -based builds and would not likely function on systems that utilize alternate manufacturer interfaces such as HTC Sense or Samsung TouchWiz.

The third method to provide secure personal elements input would be the use of overlays or other alternate input channels. Newer Android API permissions now allow applications to break the traditional security compartmentalization restrictions that formerly were in place and allow

applications to universally float above all other open applications. This is conceptually similar to a traditional desktop type environment with the ability to drag windows instead of the traditional design and usage pattern of a single application being active at once and filling the entire screen. This method, as an application, would be device, vendor, and platform independent retaining functionality on alternate manufacturer interfaces. In addition, it offers simple, straightforward, user space application level development as well as the ease of deployment, installation, and update offered by the standard SDK. Since this would not require the replacement of the default IME, swipe based inputs, emoticons, auto correction, and custom dictionaries would also remain function.

While this method may offer many benefits, such constructs are relatively new and not yet officially fully supported by the official Android SDK. The code base may change resulting in broken functionality and is development effort. Additionally, such interactions are relatively new and unfamiliar to the user, and may prove to be awkward for use in a practical environment.

The fourth option would be the use of hash tags, keywords, and macro expansion to let the user to enter secure personal elements without the need to type them on keep. This method would use predefined aliases to represent secure data inputs for example: #fname -> Michael; #lname -> Mitchell; #csemail -> mitchell@cs.fsu.edu. This method can leverage the existing custom dictionary auto completion which makes it easier for the user to remember and input aliases. This method can also utilize standard application level development techniques opening up the range of supported devices and decreasing development and installation efforts.

Using hash tags and keywords as aliases for secure personal elements will also have its drawbacks. A mechanism would also need to be provided for the user to define in advance all aliases they wish to use. This number may prove to be quite large and may be challenging for the user to memorize sufficient number of aliases for practical data entry. This method, however, could be used in conjunction with any of the other three methods previously discussed.

3.3 Challenge 3: Evaluation

Evaluation of the system will need to address multiple issues. Most importantly, the system must be evaluated based on its effectiveness. That is, determining if the system is capable of identification of all sensitive data and preventing that data from being displayed on the screen. The secure user input mechanism will also need to be evaluated to demonstrate that the system can be useful in practice. Relatively low effort must be required for effective use, along with installation and configuration difficulty. Robustness, coverage, and overhead will also need to be addressed, as well as any loss of functionality or issues with legacy compatibility.

The general approach will be to utilize task-based laboratory testing, user activity simulation, and system profiling tools to fully evaluate the system.

Challenge 3a: Effectiveness

System affecting this evaluation must address two different issues: display and input. The system must identify and prevent the display of sensitive information elements on-screen. The system must also provide a mechanism for the user to enter sensitive data without visual exposure or loss of functionality.

The Google Play market has millions of published applications accessible by thousands of different hardware devices making the enumeration of all possible user, device, and application scenarios infeasible. This will require selection of representative subset these applications for appropriate effectiveness test. The identification of all personal privacy elements that could possibly be conceived as being sensitive is also equally infeasible. Thus the selection of a representative subset must also be used in this case as well.

The proposed plan will demonstrate the effectiveness against the most popular applications in each application category. This will include email apps, web browsers, social media, finance/banking, and office/productivity applications. Testing will involve common use cases where exposure of personal information elements is possible. These elements will be selected using the most common privacy elements related to identity theft. The representative application subset can be selected using download metrics from the Google Play marketplace, usage statistics gathered from previous survey, or other published usage statistics from analytic companies. Effectiveness evaluation must make assumptions: the most popular apps in each category are appropriately diverse, and the selected personal privacy elements most common to identity theft are sufficiently representative.

Challenge 3b: Coverage

The evaluation proposal will also address robustness and operating system coverage across hardware devices, operating system versions, and hardware vendors. It is common for manufacturers to have quite different interfaces as well as UI toolkits. The three most common categories of these are: AOSP and Google Play based devices, HTC Sense based devices, and Samsung TouchWiz based devices. Thus to demonstrate coverage the evaluation testing will be repeated on these vendor variants.

Challenge 3c: Overhead & Usability

To be useful in practice, the system must: not be sufficiently cumbersome to use, be relatively easy to install, and have reasonable power overhead. While these are inherently subjective metrics they can be tested using straightforward techniques. Usage difficulty can be gauged by comparing the number of key-presses required for data entry against default IME entry.

Installation difficulty can be determined by the number of additional steps and duration and complexity of system training procedure. Finally, power overhead can be calculated using internal battery tracking metrics, additional hardware and software profiling tools, or by comparison of identical devices with and without the system running.

4 Milestones

The major research challenges for this research translate directly into the major milestones for the project. The first major milestone is the design of a system to prevent personal information exposure based on data gathered from human subject studies. This process has required gaining insight into how people compute differently in public and in private locations and quantifying these claims. It has also required addressing more subjective issues about the notion of privacy in general and how different this concept can vary from person to person. This major milestone has already been overcome through the course of the human subject privacy survey and submission for publication of its results on 3/2014.

The second major milestone for the research project is the prototype of a functional system to provide observation resistant computing support with sensitive data input. Overall, this is expected to be completed by 7/2014. This first part of this will be the identification of the ways to intercept the display for sensitive data. This portion of the work is outlined in challenge section 2b and has been completed as of 4/2014. It will be necessary for the system handle to sensitive data in any form that the system may present it. These different data representations are described in challenge section 2a. This portion of the work is expected to be completed by 6/2014. The next task will be the design and development of a lightweight user interface. This will need to be sufficient for the user to enter sensitive data without leaking privacy elements to a visual observer. It will also need to be sufficiently unobtrusive and not break legacy functionality. This is planned to be completed by 7/2014.

The final major milestone will be the evaluation of the system. This will involve the proposed evaluation metrics and methods discussed in challenge section 3, and planned for completion by 10/2014. The primary evaluation metric of effectiveness will be the most challenging and time-consuming. Using the previously discussed methods, this will be completed by 9/2014. The following evaluations of coverage and overhead will be comparatively more straightforward and should be completed shortly thereafter, by 10/2014.

Research findings have been submitted for publication to W2SP as of 3/2014, with further research to be disseminated at around 6/2014. Finally, the last milestone will be the writing of the dissertation, and planned defense by the end of 12/2014.

References

[1] Honan, Brian. “Visual Data Security White Paper”, July 2012. BH Consulting & European Association for Visual Data Security. http://www.visualdatasecurity.eu/wp-content/uploads/2012/07/Visual-Data-Security-White-Paper.pdf. Retrieved 4/2014

[2] Thomson, Herbert H, PhD. "Visual Data Breach Risk Assessment Study." 2010. People Security Consulting Services, Commissioned by 3M. http://solutions.3m.com/3MContentRetrievalAPI/BlobServlet?assetId=1273672752407&assetType=MMM_Image&blobAttribute=ImageFile. Retrieved 4/2014

[3] Vikuiti Privacy Filters. "Shoulder Surfing Survey". 2007. Commissioned by 3M UK PLC. http://multimedia.3m.com/mws/mediawebserver?6666660Zjcf6lVs6EVs66SlZpCOrrrrQ-. Retrieved 4/2014

[4] European Association for Visual Data Security. "Visual Data Security", March 2013. http://www.visualdatasecurity.eu/wp-content/uploads/2013/03/Secure-Briefing-2013-UK.pdf. Retrieved 4/2014

[5] International Data Corporation. “Worldwide Mobile Worker Population 2011-2015 Forecast.” http://cdn.idc.asia/files/5a8911ab-4c6d-47b3-8a04-01147c3ce06d.pdf. Retrieved 4/2014

[6] Good Technology. "Americans are Working More, but on their Own Schedule", July 2012. http://www1.good.com/about/press-releases/161009045.html. Retrieved 4/2014

[7] Nokia, USA. "Nokia Lumia 1020", http://www.nokia.com/us-en/phones/phone/lumia1020/. Retrieved 4/2014

[8] NPD DisplaySearch. “Wide Viewing Angle LCD Technologies Gain Share Due to Tablet PC Demand”. January 2012. http://www.displaysearch.com/cps/rde/xchg/displaysearch/hs.xsl/120119_wide_viewing_angle_lcd_technologies_gain_share_due_to_tablet_pc_demand.asp. Retrieved 4/2014

[9] Pillai, Geetha. "Caught on Camera: You are Filmed on CCTV 300 Times a Day in London", *International Business Times*, March 2012. http://www.ibtimes.co.uk/britain-cctv-camera-surveillance-watch-london-big-312382. Retrieved 4/2014

[10] Loh Zhi Chang and Steven Zhou ZhiYing. “Robust pre-processing techniques for OCR applications on mobile devices”, *In Proceedings of the 6th International Conference on Mobile Technology, Application & Systems (Mobility '09)*. ACM, New York, NY, USA, Article 60 , 4 pages. DOI=10.1145/1710035.1710095 http://doi.acm.org/10.1145/1710035.1710095

[11] Owen, Glen. "The zzzzivil servant who fell asleep on the train with laptop secrets in full view", November 2008. http://www.dailymail.co.uk/news/article-1082375/The-zzzzivil-servant-fell-asleep-train-laptop-secrets-view.html. Retrieved 4/2014

[12] Penn, Ivan. "Simple fix to bank security breach: Close the blinds", *Tampa Bay Times*. December 2010. http://www.tampabay.com/features/consumer/simple-fix-to-bank-security-breach-close-the-blinds/1139356. Retrieved 4/2014

[13] Davies, Caroline. "Prince William photos slip-up forces MoD to change passwords", *The Guardian*, November 2102. http://www.theguardian.com/uk/2012/nov/20/prince-william-photos-mod-passwords. Retrieved 4/2014

[14] J. Alex Halderman, Brent Waters, and Edward W. Felten. "A convenient method for securely managing passwords", *In Proceedings of the 14th international conference on World Wide Web (WWW '05)*. ACM, New York, NY, USA, 471-479. DOI=10.1145/1060745.1060815 http://doi.acm.org/10.1145/1060745.1060815

[15] LastPass, "About LastPass", 2014. https://lastpass.com/about-lastpass. Retrieved 4/2014

[16] AgileBits, Inc. "1Password", 2014. https://agilebits.com/onepassword. Retrieved 4/2014

[17] Yubico, Inc. "About YubiKey", 2014. http://www.yubico.com/about. Retrieved 4/2014

[18] Square, Inc. "About Square", 2014. https://squareup.com/news. Retrieved 4/2014

[19] Google, Inc. "Google NFC YubiKey Neo", September 2013. http://online.wsj.com/news/articles/SB10001424127887323585604579008620509295960

[20] Wayne Jansen and Vlad Korolev. "A Location-Based Mechanism for Mobile Device Security", *In Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering (CSIE '09)*, Vol. 1. IEEE Computer Society, Washington, DC, USA, 99-104. DOI=10.1109/CSIE.2009.719 http://dx.doi.org/10.1109/CSIE.2009.719

[21] Bryan Parno, Cynthia Kuo, and Adrian Perrig. "Phoolproof phishing prevention", *In Proceedings of the 10th international conference on Financial Cryptography and Data Security (FC'06)*, Giovanni Crescenzo and Avi Rubin (Eds.). Springer-Verlag, Berlin, Heidelberg, 1-19. DOI=10.1007/11889663_1 http://dx.doi.org/10.1007/11889663_1

[22] Blonder, Greg E. "Graphical Passwords". United States patent 5559961, Lucent Technologies, Inc. 1996.

[23] Passfaces Corporation. "The Science Behind Passfaces", June 2004. http://www.realuser.com/published/ScienceBehindPassfaces.pdf

[24] Darren Davis, Fabian Monrose, and Michael K. Reiter. "On user choice in graphical password schemes", *In Proceedings of the 13th conference on USENIX Security Symposium - Volume 13 (SSYM'04)*, Vol. 13. USENIX Association, Berkeley, CA, USA, 11-11.

[25] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir Memon. “PassPoints: design and longitudinal evaluation of a graphical password system” International *Journal of Human-Computer Studies*. 63, 1-2 (July 2005), 102-127. DOI=10.1016/j.ijhcs.2005.04.010 http://dx.doi.org/10.1016/j.ijhcs.2005.04.010

[26] Jain, A.K.; Hong, L.; Pankanti, S.; Bolle, R., "An identity-authentication system using fingerprints," *Proceedings of the IEEE*, vol.85, no.9, pp.1365, 1388, Sep 1997. doi: 10.1109/5.628674

[27] J. Daugman. “How iris recognition works”, *IEEE Transactions on Circuits and Systems for Video Technology*. 14, 1 (January 2004), 21-30. DOI=10.1109/TCSVT.2003.818350 http://dx.doi.org/10.1109/TCSVT.2003.818350

[28] Anil K. Jain, Arun Ross, Sharath Pankanti. "A Prototype Hand Geometry-based Verifcation System", *In Proceedings of 2nd International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, Washington D.C., pp.166-171, March 22-24, 1999.

[29] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. “Face recognition: A literature survey”. *ACM Computing Surveys*. 35, 4 (December 2003), 399-458. DOI=10.1145/954339.954342 http://doi.acm.org/10.1145/954339.954342

[30] Rick Joyce and Gopal Gupta. “Identity authentication based on keystroke latencies”, *Communications of the ACM* ,33, 2 (February 1990), 168-176. DOI=10.1145/75577.75582 http://doi.acm.org/10.1145/75577.75582

[31] Davrondzhon Gafurov, Kirsi Helkala, Torkjel Søndrol. "Biometric Gait Authentication Using Accelerometer Sensor", *Journal of Computers*, Vol. 1, No. 7, October 2006.

[32] Roberto Brunelli and Daniele Falavigna. “Person Identification Using Multiple Cues”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 17, 10 (October 1995), 955-966. DOI=10.1109/34.464560 http://dx.doi.org/10.1109/34.464560

[33] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. “Touch me once and I know it's you!: implicit authentication based on touch screen patterns”, *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 987-996. DOI=10.1145/2207676.2208544 http://doi.acm.org/10.1145/2207676.2208544

[34] Ioannis Leftheriotis. “User authentication in a multi-touch surface: a chord password system” In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 1725-1730. DOI=10.1145/2468356.2468665 http://doi.acm.org/10.1145/2468356.2468665

[35] Ming Ki Chong, Gary Marsden, and Hans Gellersen. “GesturePIN: using discrete gestures for associating mobile devices”, *In Proceedings of the 12th international conference on Human computer interaction with mobile devices and services (MobileHCI '10)*. ACM, New York, NY, USA, 261-264. DOI=10.1145/1851600.1851644 http://doi.acm.org/10.1145/1851600.1851644

[36] Adam J. Aviv, Katherine Gibson, Evan Mossop, Matt Blaze, and Jonathan M. Smith. “Smudge attacks on smartphone touch screens”, *In Proceedings of the 4th USENIX conference on Offensive technologies (WOOT'10)*. USENIX Association, Berkeley, CA, USA, 1-7.

[37] Volker Roth, Kai Richter, and Rene Freidinger. “A PIN-entry method resilient against shoulder surfing”, *In Proceedings of the 11th ACM conference on Computer and communications security (CCS '04)*. ACM, New York, NY, USA, 236-245. DOI=10.1145/1030083.1030116 http://doi.acm.org/10.1145/1030083.1030116

[38] T. Perkovic, M. Cagalj, and N. Rakic. “SSSL: shoulder surfing safe login”, In *Proceedings of the 17th international conference on Software, Telecommunications and Computer Networks (SoftCOM'09)*. IEEE Press, Piscataway, NJ, USA, 270-275.

[39] Alice Boit, Thomas Geimer, and Jorn Loviscach. “A random cursor matrix to hide graphical password input”, *In SIGGRAPH '09: Posters (SIGGRAPH '09)*. ACM, New York, NY, USA, Article 41, 1 pages. DOI=10.1145/1599301.1599342 http://doi.acm.org/10.1145/1599301.1599342

[40] Rohit Ashok Khot, Ponnurangam Kumaraguru, and Kannan Srinathan. “WYSWYE: shoulder surfing defense for recognition based graphical passwords”, *In Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12)*, ACM, New York, NY, USA, 285-294. DOI=10.1145/2414536.2414584 http://doi.acm.org/10.1145/2414536.2414584

[41] Rachna Dhamija and Adrian Perrig. “Deja; Vu: a user study using images for authentication”, *In Proceedings of the 9th conference on USENIX Security Symposium - Volume 9 (SSYM'00)*, Vol. 9. USENIX Association, Berkeley, CA, USA, 4-4.

[42] Mary Brown and Felicia R. Doswell. “Using passtones instead of passwords”, *In Proceedings of the 48th Annual Southeast Regional Conference (ACM SE '10)*. ACM, New York, NY, USA, Article 82, 5 pages. DOI=10.1145/1900008.1900119 http://doi.acm.org/10.1145/1900008.1900119

[43] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. “The secure haptic keypad: a tactile password system”, *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 1089-1092. DOI=10.1145/1753326.1753488 http://doi.acm.org/10.1145/1753326.1753488

[44] Alain Forget, Sonia Chiasson, and Robert Biddle. “Shoulder-surfing resistance with eye-gaze entry in cued-recall graphical passwords”, *In Proceedings of the SIGCHI Conference on*

Human Factors in Computing Systems (CHI '10). ACM, New York, NY, USA, 1107-1110. DOI=10.1145/1753326.1753491 http://doi.acm.org/10.1145/1753326.1753491

[45] Julie Thorpe, P. C. van Oorschot, and Anil Somayaji. “Pass-thoughts: authenticating with our minds”, *In Proceedings of the 2005 workshop on New security paradigms (NSPW '05)*. ACM, New York, NY, USA, 45-56. DOI=10.1145/1146269.1146282 http://doi.acm.org/10.1145/1146269.1146282

[46] Nitesh Saxena and James H. Watt. “Authentication technologies for the blind or visually impaired”, *In Proceedings of the 4th USENIX conference on Hot topics in security (HotSec'09)*. USENIX Association, Berkeley, CA, USA, 7-7.

[47] T. Dierks, E. Rescorla. "The Transport Layer Security (TLS) Protocol, Version 1.2", August 2008.

[48] Mason, Andrew G. “Cisco Secure Virtual Private Network”. Cisco Press, 2002, p. 7.

[49] Marc Shapiro. "Structure and Encapsulation in Distributed Systems: the Proxy Principle", *In Proceedings of the 6th IEEE International Conference on Distributed Computing Systems (ICDCS)*, Cambridge MA (USA), May 1986.

[50] Roger Dingledine, Nick Mathewson, and Paul Syverson. “Tor: the second-generation onion router”, *In Proceedings of the 13th conference on USENIX Security Symposium (SSYM'04)*, Vol. 13. 2004 USENIX Association, Berkeley, CA, USA, 21-21.

[51] Do Not Track. "Do Not Track - Universal Web Tracking Opt Out", http://donottrack.us. Retrieved 4/2014

[52] Adblock Plus. "Adblock Plus : About", https://adblockplus.org/en/about. Retrieved 4/2014

[53] Evidon, Inc. "About Ghostery", https://www.ghostery.com/en/about. Retrieved 4/2014

[54] Braden Kowitz and Lorrie Cranor. “Peripheral privacy notifications for wireless networks”, *In Proceedings of the 2005 ACM workshop on Privacy in the electronic society (WPES '05)*. ACM, New York, NY, USA, 90-96. DOI=10.1145/1102199.1102217 http://doi.acm.org/10.1145/1102199.1102217

[55] Sunny Consolvo, Jaeyeon Jung, Ben Greenstein, Pauline Powledge, Gabriel Maganis, and Daniel Avrahami. “The Wi-Fi privacy ticker: improving awareness & control of personal information exposure on Wi-Fi”, *In Proceedings of the 12th ACM international conference on Ubiquitous computing (Ubicomp '10)*. ACM, New York, NY, USA, 321-330. DOI=10.1145/1864349.1864398 http://doi.acm.org/10.1145/1864349.1864398