Leveraging Cellular Infrastructure to Improve Fraud Prevention

Frank S. Park, Chinmay Gangakhedkar, Patrick Traynor
College of Computing
Georgia Institute of Technology
Atlanta, Georgia
{frank, chinmay.g}@gatech.edu, traynor@cc.gatech.edu

***Abstract*—The relationship between physical security and critical infrastructure has traditionally been unidirectional - the former being necessary to sustain the latter. However, certain pieces of critical infrastructure hold the potential to significantly improve the security of individuals and their most sensitive information. In this paper, we develop a pair of mechanisms for cellular networks and mobile devices that augment the physical security of their users' financial credentials. In particular, we create FrauVent, a multi-modal protocol that provides users with information related to a pending questionable transaction (e.g., transaction value, location, vendor) in a way that improves the available context for approving or rejecting such exchanges. Through protocol design, formal verification and implementation of an application for the Android platform, we develop a robust tool to help reduce the costs of fraud without requiring financial institutions to significantly change their extensively deployed end systems (i.e., card readers). More critically, we provide a general framework that allows cellular infrastructure to actively improve the physical security of sensitive information.**

***Index Terms*—Infrastructure-Assisted Security; Cellular Networks; Multi-factor Authentication; Mobile Phones; Credit Card Authentication**

I. Introduction

The frequency with which financial credentials are stolen is well documented. Whether through phishing [25], [13], [22], spyware [42], [27], card skimming [51], [47] or physical theft of credit cards, traditional mechanisms for preventing fraud are becoming less effective. For instance, in all but the last of the above cases, users willingly provide their PIN codes to the attackers. As such attacks become increasingly automated, both banks and end users are being forced to deal with losses on a more regular basis.

The academic and industrial communities have responded with a number of solutions designed to improve the security of financial credentials. For instance, keychain fobs that generate pseudorandom streams of bits for use as a second authentication factor have become popular within defense and education related organizations [38], [16], [33], [48]. These additional authentication tokens make attacks attempting to forge credentials, such as password guessing, significantly more difficult. However, these and other similar systems have not been widely embraced by the financial industry for a number of reasons. First, these devices and their supporting infrastructure are relatively expensive to purchase and maintain, making them potentially unpalatable to many banks. Second, average consumers are unlikely to remember to carry these devices with them at all times, frustrating legitimate access attempts. Finally, carefully crafted attacks [14] can entirely circumvent protections offered by these devices. Accordingly, more effective and robust mechanisms must be investigated.

In this paper, we develop a suite of protocols that allow users and banks to more effectively collaborate on the detection and prevention of fraudulent financial credential use. We design our application, FrauVent (i.e., Fraud Preventer), around the increasingly likely probability that mobile phones are carried by their owners during most point-of-sale transactions. This multi-modal protocol allows clients to make informed decisions about transactions using information about their physical location in comparison to the source of a transaction. In combination with traditional fraud monitoring infrastructure, this mechanism can not only greatly improve the ease with which end users can assist in fraud prevention, but will also allow the nearly ubiquitously deployed cellular infrastructure to take a more active role in protecting user data.

In so doing, we make the following contributions:

- **Discuss shortcomings of currently available mechanisms:** While a number of other cellular-based solutions have been proposed, they fail to address the issues raised in this paper for a number of reasons. We discuss why our approach is more robust by design.
- **Design and formally verify the security of the FrauVent application:** We develop a multi-modal protocol that allows users to apply contextual information (e.g., location information) to determine the legitimacy of a transaction. We then use ProVerif [3], [4], [6] to provide guarantees of the protocol's security properties.
- **Implement and characterize the performance of our system:** We build a prototype version of our system for Android phones and then characterize the performance of each mode of the protocol. We show that the currently deployed credit card infrastructure (i.e., swipe terminals) requires no modifications to support this protocol.

Realizing these mechanisms faces a number of significant challenges. Exchanges should not be forgeable by an adversary positioned between the mobile device and the bank, including an insider within the cellular provider [34]. Accordingly, the user must participate at some level in such a system.

1063-9527/09 $26.00 © 2009 IEEE
DOI 10.1109/ACSAC.2009.40

IEEE computer society

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 1

These protocols must also overcome difficulties of usability when determining physical location. Making such connections intuitive is a necessary component of creating a robust system, as transactions that occur outside of a user's normal geographic range (e.g., neighborhood, city, state, country) are often those most likely to be marked as fraudulent. Finally, our proposed protocols must scale so that their elevated use does not overwhelm the often limited resources available to even high-speed 3G cellular networks.

The remainder of this paper is organized as follows: Section II provides an overview of related projects; Section III defines threats and design goals of the solution; Section IV gives background information on credit card systems and the architecture of cellular networks; Section V details the design of our system; Section VI discusses the implementation details for both the backend system and the Android application; Section VII offers additional discussion and challenges for such a system; Section VIII provides concluding remarks.

II. Related Work

Multi-factor authentication expands the set of credentials required to gain access to a resource. For instance, by requiring a legitimate user to present both a password (e.g., something they know) and a physical token (e.g., something they have), an adversary's ability to gain access to a resource through the seizure of any single factor is ideally eliminated. Recognizing this, solutions ranging from pseudorandom number generating keychain fobs [38], [16], [33], [48], [12] and one-time pads [28], [39] to assorted biometric scanners [41], [9] and USB tokens [17], [49], [36] have been suggested. Whether due to issues of solution scalability, overhead or simply the requirement that all users carry some object that is not broadly usable, such solutions have failed to be deployed on a very large scale.

Credit card companies have introduced a number of extensions to their cards to approximate multi-factor authentication. The most basic protection, PIN codes, attempt to prevent malicious third-parties (e.g., card skimmers [51], [7], [20], phishers [52], [26], etc) from being able to reuse or duplicate card information for transactions other than the one approved by their owners. However, the increasing prevalence of PIN code interception by adversaries has significantly limited the protection offered by this method. Banks in Europe have responded with chip-and-PIN systems [32], [50]. Instead of only requiring knowledge of the corresponding PIN code, the cryptographically capable smart-card embedded credit card must be in the possession of the person executing the transaction. In spite of eliminating certain kinds of attacks, researchers have demonstrated that such techniques are not a sufficient means of stopping fraudulent point-of-sale purchases [14].

Cellular phones have the potential to address many of these issues. Unlike solutions requiring users to carry a potentially cumbersome object only useful for authentication with one particular network or service, cellular phones are already carried by more than 84% of the population in the United States and Western Europe [10]. Increasingly expressive user interfaces and a wide range of communications interfaces also make this platform attractive for such a task. A number of researchers have attempted to address the problem of credit card fraud through text messages [53], [29], [37] and automated voice calls [30]. These approaches fail in reality for a number of reasons. First, because there is no cryptographic mutual authentication of the message itself and text messages are easily spoofable, no real protection is offered to users. Second, while cellular networks deliver a large number of text messages, constrained wireless interfaces are unlikely to be able to support an increase in text messaging equivalent to the total number of credit card purchases made every day [44], [45], [43]. Solutions that are more robust against the kinds of adversaries mentioned above and sensitive to the limitations of cellular networks must therefore be developed.

III. Design Considerations

A. Threat Model

Banks and financial institutions have long worked to reduce the cost associated with fraud. Current algorithms detect approximately 90% of such attempts through the use of a wide range of factors [19]. Unfortunately, preventing the remaining 10% of fradulent activity is often viewed as more costly than the fraud itself. We aim to help reduce such loss through the use of informed user participation in questionable situations.

Key in developing a system to combat these outlier cases is defining an appropriate threat model. We first argue that the majority of card theft cases are the result of electronic interception and not physical attacks. A user is far more likely to have their credit card number and PIN stolen via an infected browser/desktop machine, an altered card reader [14] or a poorly protected database [18] than they are to have their wallet stolen. While the second case certainly does happen, an attacker forfeits the stealth they may temporarily gain by surreptitiously recovering such information. As a second point, the changing nature of stolen credential use makes traditional fraud prevention mechanisms less effective. In particular, as adversaries increasingly use stolen credentials a single time to withdraw cash [18], [51], *post-facto* defenses are generally unable to recover losses. We therefore assert that the use of the right two factor authentication mechanism used at the time of a transaction of questionable integrity can potentially help mitigate such attacks.

Selecting the right second factor for authentication is challenging. As mentioned in Section II, a number of other researchers have attempted to use mobile phones to address this problem in the past. However, techniques of simply sending text messages [53], [29], [37] or relying on caller ID [8] as strong authenticators are easily spoofable. Systems such as key fobs [38], [16], [33], [48], [12] and USB tokens [17], [49], [36] have also been suggested, but often fail in the context of banking due to users losing or forgetting to carry them. Smart phones have the ability to perform strong cryptography, are carried at nearly all times by users and possess increasingly expressive interfaces, making them a good candidate to address these issues.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 2

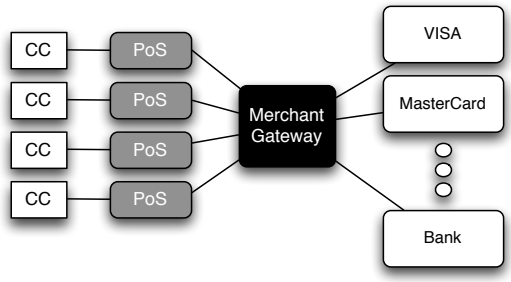

Fig. 1. High-Level Credit Card Infrastructure

A very simple mechanism to address this problem would simply allow a credit card company to ask a cellular provider for a user's current location. However, a user's location information is potentially sensitive. Such a solution may allow an adversary within a bank to maliciously track a user. Moreover, as demonstrated in the Remsburg vs. DocuSearch case [1], it may make a provider liable for the improper release of information to third parties and is therefore unlikely to be widely implemented. A successful solution in this space must therefore *make the user the ultimate arbiter of their location information.*

B. Design Goals

We used the above considerations to create a series of design goals. First, *no additional device* should be required by the accountholder. The added burden on the consumer tends to decrease the usability of the solution, forcing consumers to disable or abandon the security feature completely. Second, a receiving party should have the ability to verify the identity of the sender - there must be an *end-to-end authentication* mechanism to confirm the identity of the participants. This goal addresses possible forged attempts by adversaries. Once the communication channel is established, *confidentiality* of the contents exchanged should be enforced. This requirement is especially important, given the potential for insider attacks in such networks [34]. Fourth, an accountholder detecting a fraudulent transaction must have the *ability to prevent* such a transaction from proceeding.

Our fifth goal is *scalability*. The layered design of the Internet allows application designers to create programs without having a deep understanding of the networks over which their data will traverse. Cellular networks, however, are built with a different set of assumptions and are much more easily overloaded for reasons other than bandwidth constraints [46]. Accordingly, communications in our system will not be sent for every transaction as is proposed by other related protocols [53]. Instead, we reduce the stress on the network by communicating only in the rare case that fraud is suspected. Sixth, we must be sure that our protocol *fails in an acceptable fashion*. If a client does not respond to a query, a transaction should fail. Finally, we attempt to make as few changes as possible so that our solution can be easily implemented. In particular, our protocols will *make no changes* to deployed ATMs and credit card readers and will instead add minimal extra processing in the back-end.

IV. Overview

The ability to establish a secondary channel for communication and to request location information using the cellular network can significantly augment the authorization process. In this section, we will present an overview of credit card authentication and authorization processes as well as the features we will be using on cellular network.

A. Credit Card Systems

The credit card has become one of the most widely accepted forms of payment today with more than 500 million credit cards in circulation in the United States [23]. Its versatility, usability, and ability to defer payment allows consumers to generate more than 1.9 billion transactions a month [23]. We discuss credit card transactions at a high level and note a number of common practices within the industry.

Figure 1 illustrates a high-level credit/debit card infrastructure. Once a consumer or a cashier enters the credit card information into the point-of-sale (PoS) machine, the request is sent to a gateway where the authorizing party is determined and forwarded. This can be a credit card company such as Visa or MasterCard, or an issuing bank for debit cards. The issuer of the card authorizes the requested transaction based on available credit limit or a balance, previously known spending habits of the user, and any other credentials collected on the PoS such as the PIN number. Once the bank determines its response, it is sent back to the gateway, which then is forwarded to the PoS machine. This entire process typically takes less than 15 seconds, depending on the connection speed of the PoS machine.

In order to save time between transactions, some merchants choose to batch submit low-risk purchases, eliminating the authorization process until the end of the day. This batching is typically observed at vendors such as coffee shops where typical transactions are less than ten dollars and the flow of customers peak during a short period of time (e.g., start of the day). In this case, the credit card number is checked locally to make sure it is in valid format before storing the request information for batching. Note that these transactions are guaranteed by the banks issuing the credit cards as fraud of this magnitude represents an extremely small loss even in the aggregate.

Transactions involving larger amounts can also be batched at the transferral step. In particular, these transactions are typically sent to the bank to ensure that the necessary funds are available, but transferral of funds between accounts often takes place at some later time. Such transactions are typically run against the bank's fraud detection algorithms. These systems use a number of characteristics of the sale (e.g., price, distance from card holder's home, time of day) to "score" the likelihood of fraud. Should this score exceed some threshold, the bank prevents the transaction from occurring. Disputed or fraudulent

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 3

Method	Measurement Src	Accuracy
Cell ID	Base Station	100m - 3km
Cell ID + TA	Base Station	500m*
AOA	Base Station	100 - 200m
U-TDOA	Base Station	$<$ 50m
EOTD	Handset	50 - 200m
GPS / A-GPS	Handset	5 - 30m

** Depends on bandwidth*

TABLE I
POSITIONING TECHNOLOGY COMPARISON

transactions can easily be recovered from in such a system as money often takes two days to be transferred between accounts - however, loss from transactions in which cash is withdrawn is exceedingly difficult to recover. Lowering the detection threshold would allow banks to prevent more potentially fraudlent exchanges from occurring; however, such a change has traditionally been viewed as too expensive given the need to hire more people to address false positives. *FrauVent seeks to circumvent such expense by having the user themselves intervene in the rare instances when a transaction is questionable or receives a borderline score.*

B. GSM Localization

Knowledge of the merchant's and the consumer's location information during the authorization process can significantly enhance the decision made by the user and the authorizing bank. Such information allows a user to detect transactions that might have previously gone unnoticed. Verified location information of the consumer can offer assurance to the bank, removing false positives while allowing fraud detection algorithms to be more aggressive in what they classify as fraud. The mechanisms to discover the location of the consumer on-demand can be accomplished using the existing technology available in cellular networks.

There are several methods of location discovery in the GSM network as depicted in Table I. The most accurate method of localization has been standardized by 3GPP [2] to be used for E911 and other location based services. Uplink Time Difference Of Arrival (U-TDOA) uses multilateration, also known as hyperbolic positioning, to approximate the position within 50 meters of the actual device. This method, as depicted in Figure 2, measures the time difference of a cell signal reaching multiple Location Measurement Units (LMU) via multiple base stations. Although U-TDOA does not require modifications to phones, it can be used in conjunction with Assisted-GPS (A-GPS) or Angle Of Arrival (AOA) to improve the accuracy up to 20 meters from the actual location.

Even though this technology has been available to the telecommunication industry for the last several years, it has not yet been available for a third-party location reporting service due to the tremendous liability that the provider incurs in distributing private information. However, current services offered by major providers (e.g., turn-by-turn navigation) already take advantage of this infrastructure. We propose that FrauVent similarly avail itself of first-person query functionality as part of transaction confirmation. We discuss the advantages of this approach in the following section.

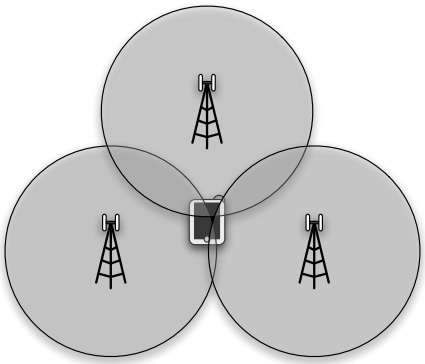

Fig. 2. Location discovery using multilateration

V. FRAUVENT ARCHITECTURE

Given the background information on the supporting infrastructure, we now present the details of the two protocols used by FrauVent. In the *reactive* protocol, users receive a modified text message when suspicious transactions occur. The *proactive* protocol allows a user suspecting that a transaction may be flagged (e.g., when traveling across the country) to log in and approve each request. We discuss how both protocols overcome the pitfalls of related work using similar techniques and then use tools to provide formal guarantees about the security of our constructions.

A. Protocol Definitions

1) Reactive Protocol: Our protocol uses location information to provide users with improved context for decision making. Whereas previously proposed systems simply transmitted a text message saying that a transaction has occurred, we will help the user correlate their current location with the location of the transaction in question. However, conveying location to a user can be surprisingly difficult. While GPS information is accurate to within 50 feet for civilian devices, the coordinates themselves have little *semantic value* to a human being. The use of city names, while seemingly straightforward, may also provide significant confusion as many such areas are ambiguously defined to unfamiliar users (e.g., hotels in the vicinity of the NSF may be in the cities of Arlington or Alexandria and the line between the two is not always clear). We address this issue by leveraging the GPS and map interfaces in many recently released smart phones. These tools make location more intuitive and improve the chances that a user will correctly determine whether fraud has occurred.

Figure 3 offers a high level overview of the reactive phase of our proposed protocol. Like all standard point-of-sale transactions, our system is activated by the user swiping their credit card at a vendor. The vendor transmits a request to the bank to confirm that the client's transaction is approved. If the bank flags the transaction as anomalous, the purchase is traditionally denied. In the presence of our protocol, the

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 4

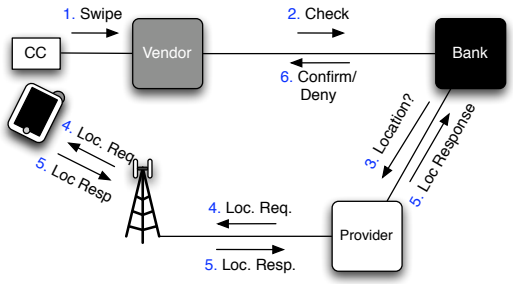

Fig. 3. *High-level overview of the reactive location confirmation protocol:* When a credit card is swiped, the vendor sends the credentials to the bank as normal. If the bank suspects fraud, it transmits a location request for the cellular phone corresponding to the client. Using a shared secret, the client is able to confirm or deny the use of the card. The bank then sends or denies permission for the vendor to complete the transaction.

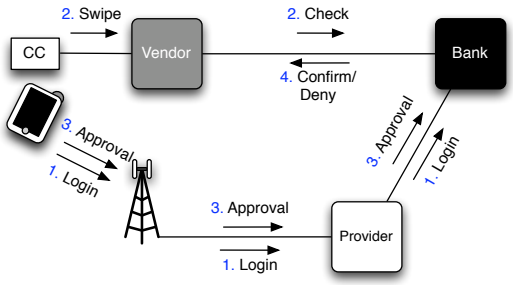

Fig. 4. *High-level overview of the proactive location confirmation protocol:* In this scenario, the user runs an application when they believe that their transaction is likely to be rejected and confirms their location proactively.

bank attempts an additional confirmation from the user. Using the mobile number the client has registered, the bank sends a location verification request through the mobile device's cellular provider. Note that this request contains the location, vendor and amount of the transaction, all encrypted with a key shared between the bank and the user. The provider responds by locating the device and approximates its GPS coordinates through a technique such as multilateration [40]. These coordinates are then signed using the network's private key and forwarded to the client, which then independently determines its current location via GPS. This allows a client to prevent an insider from forging their location. An application on the client device then uses all of the above supplied location information to render a map showing both the details of the transaction and the relative proximity of the user to it. If the user confirms that the transaction occurred, they respond by sending an encrypted copy of all of the location information and a confirmation of the transaction to the bank. If the client confirms the transaction, the bank allows it to proceed. A negative or non-existent response forces the bank to revert to its default response and reject the transaction. The protocol is defined formally in Figure 5.

This protocol overcomes the authentication issues that limit the effectiveness of previous protocols. Our mechanisms also make the user the arbiter of their location information and do not introduce any new means for a network operator to reveal a client's location. Instead of having the network automatically respond to the location verification request and potentially leak private information, only the user can correctly encrypt the returning packets.

2) Proactive Protocol: While the protocol described in the previous section addresses the authentication and privacy issues in current text messaging-based protocols, it fails to address real-time constraints. Even though our protocol only transmits a text message at the suspicion of fraud (and not after every transaction as previously suggested), the store-and-forward nature of text messaging means that there are no guarantees that such traffic will be delivered in a timely manner [43]. We therefore develop a protocol that attempts to operate before the reactive mechanism is initiated. A user having experienced a denied legitimate transaction can execute this protocol to ensure success.

Figure 4 offers a high level overview of the proactive protocol. When a user goes to make a transaction they feel is likely to be denied as potentially fraudulent (e.g., very high transaction value, location deviates from normal geographic range), they activate an application on their mobile device. Through this application, the user logs in to a transaction monitor on the bank's server, which updates the user when the transaction is received. Like our proposed reactive mechanism, the application in the proactive protocol presents the client with a map and compares the location of the purchase against the client's current position. The client either approves or denies the transaction via a data connection (e.g., GPRS/UMTS, 802.11x), and the bank then propagates this decision back to the vendor. Note that this phase can be used as a first option. The system can default to the reactive phase if a client does not remember to log in via the proactive application. Figure 6 provides a more formal definition of this protocol.

This approach is more scalable for a number of reasons. First, unlike the asynchronous service model of SMS, the user is able to initiate the protocol and exchange packets directly with the bank's server. This means that the multi-second delay associated with the store-and-forward behavior of text messaging is removed. Second, because locating a user is among the most expensive and time consuming operations in a cellular network [46], this approach significantly reduces the effort required by the infrastructure.

B. Formal Verification

Having defined the protocols making up FrauVent, we next ensure that they provably provide the necessary guarantees. To accomplish this goal, we used a tool called ProVerif [5]. ProVerif is an automated cryptographic protocol verifier that can accurately prove the secrecy and equivalency of the protocols by translating them to Horn clauses. Using this tool, we can positively verify if the secrecy of the information transmitted as defined by the proposed protocol is preserved. We present sample output from our evaluation of the reactive protocol to provide additional insight into the process; however, both protocols provide the same guarantees.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 5

1. $C \rightarrow V : CC$
2. $V \rightarrow B : \langle CC, T \rangle_{k_{B,V}}$
3. $B \rightarrow P : \langle C, \langle T \rangle_{k_{B,C}} \rangle_{k_{B,P}}$
4. $P \rightarrow C : \langle L_C \rangle_{k_{C,P}}, \langle T \rangle_{k_{B,C}}$
5. $C \rightarrow B : \langle R, L_C, HMAC_{k_{B,C}}(T|L_C) \rangle_{k_{B,C}}$
6. $B \rightarrow V : \langle Approved\ or\ Declined, TransactionID \rangle_{k_{B,V}}$

B, C, P, V	:	Bank, Client, Provider, Vendor
CC	:	Credit Card
$HMAC_{k_A}(R\|S)$	:	HMAC with key k_A on R and S
$k_{A,B}$	:	Secret key between A and B
L_A	:	Location from A's perspective
$SIG_{k_{\bar{A}}}(M)$	:	Digital Signature of M using A's private key
R	:	Response (Approval, L_P, $SIG_{k_{\bar{P}}}(L_P)$, L_A, T)
$\langle S \rangle_k$	:	S encrypted with key k
T	:	Transaction (amount, timestamp, vendor location, id)

Fig. 5. *Reactive Protocol and Variable Definition:* The formal definition of the Reactive Protocol. We assume that there exists an encrypted channel (e.g. IPsec connection) between the vendor and bank and the bank and the provider.

1. $C \rightarrow B : LI_C$
2. $B \rightarrow C : LR_C$
3. $C \rightarrow V : \langle CC \rangle_{k_{C,V}}$
4. $V \rightarrow B : \langle CC, T \rangle_{k_{B,V}}$
5. $B \rightarrow C : \langle L_V, T \rangle_{k_{B,C}}$
6. $C \rightarrow B : \langle R, L_C, HMAC_{k_{B,C}}(T|L_C) \rangle_{k_{B,C}}$
7. $B \rightarrow V : \langle Approved\ or\ Declined, TransactionID \rangle_{k_{B,V}}$

B, C, V	:	Bank, Client, Vendor
LI_A, LR_A	:	Login Information, Login Response of A
CC	:	Credit Card
$HMAC_{k_A}(R\|S)$	:	HMAC with key k_A on R and S
$k_{A,B}$	:	Secret key between A and B
L_A	:	Location from A's perspective
R	:	Response (Approval, L_P, $SIG_{k_{\bar{P}}}(L_P)$, L_A, T)
$SIG_{k_{\bar{A}}}(M)$	:	Digital Signature of M using A's private key
$\langle S \rangle_k$	:	S encrypted with key k
T	:	Transaction (amount, timestamp, vendor location, id)

Fig. 6. *Proactive Protocol and Variable Definition:* The formal definition of the Proactive Protocol. We assume that there exists an encrypted channel (e.g. IPsec connection) between the vendor and bank and the bank and the provider.

The reactive protocol, which consists of six messages being exchanged as defined in Figure 5, requires 22 rules to be defined. An example is shown in Figure 7.

ProVerif operates on the assumption that an adversary is able to intercept every message between the vendor, the bank, the cellular provider, and the client. Here, an attacker captures the message carrying the phone number (defined as part of T) as it is transmitted between the bank B and provider P in Step 3 of Figure 5. In particular, the above rule says that if the attacker can understand the third message in this protocol, it will be able to deduce $\{TEL\}_{k_{BP}}$. Assuming that an adversary does not know the symmetric cryptographic key used to encrypt the content in all of the messages, ProVerif was able to confirm that an adversary cannot derive any of the information exchanged, which includes the credit card number, mobile telephone number, location and transaction data.

VI. Deployment

In this section, we discuss our observations from the implementation and testing of a prototype mobile application. We have also developed software executing the necessary back end functionality required of the bank, telecommunications provider and the PoS terminal. After swiping a card through a Posh MX3 card reader, the PoS terminal forwards the transaction information to the bank, which automatically flags all requests. Our provider emulator then transmits a text message that is received by our test phone. We discuss the details of the mobile application and provider emulator herein.

A. User Agent Platform

We selected the Android platform to serve as the basis of our mobile application [21]. We ran Android on an OpenMoko FreeRunner GTA02 [31], a Linux 2.66 kernel tri-band GSM/GPRS phone with 400MHz ARM4 processor, 128MB of SDRAM, and 256MB of integrated flash memory which was expanded with 512MB external microSD card. The FreeRunner also provides GPS, Bluetooth and 802.11 b/g support. Each application in Android is run as its own process within its own instance of the Dalvik virtual machine. FrauVent required over 2,000 lines of code in Java and XML using the Android SDK 1.1 release 2.

Running Android on an OpenMoko phone is challenging in and of itself. In particular, Android is designed to run on phones with an ARM5 processor, whereas our FreeRunner was

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 6

```
Rule 19:
  attacker:encrypt((TEL_97,T_98),kBP_8[]) -> attacker:encrypt(TEL_97,kBP_8[])
(If the message encrypt((TEL_97,T_98),kBP_8[]) is received from the
attacker at input {9}, then the message encrypt(TEL_97,kBP_8[]) may be
sent to the attacker at output {12}.)
```

Fig. 7. One of 22 rules defined in ProVerif.

equipped with an ARM4 chip. Through the assistance of the open source community, a number of opcodes were added to ARM4 to allow this device to understand ARM5 opcodes in the pre-compiled libraries. Android also lacked on screen and external keyboard interfaces at the beginning of this work.[1] We eventually were able to flash Koolu's beta 7 version of Android onto the device and gain support for basic cellular capabilities necessary to implement FrauVent.

Using the standard Android SDK, FrauVent's user interface incorporates Google Map and SMS/MMS trigger functionalities. Once initialized, FrauVent deploys an SMS trigger to capture all incoming text messages that are specially formatted. A text message is used for initial communication during reactive mode in order to wake-up the device when the device is in stand-by mode. This provides the application with a push technology, enabling the requests to be alerted to the user in real-time. When a transaction request is received by the device, FrauVent's dashboard opens with the pending request information. A user noticing this request is then presented with an interactive map, shown in Figure 8, depicting the location of its mobile device relative to that of the vendor. With this given information, the user can determine if the transaction is originating from the expected vendor, eliminating the possibility of relay attack. Once the location is confirmed, the user is provided with the textual information of the requested transaction, similar to the screen shown in Figure 8, for further verification of the request. A user authorizing the transaction would be required to provide an authentication token to complete the request. Although we have currently implemented a PIN pad for authentication, we believe that it should eventually be replaced with a less intrusive way of generating the token.

B. WhereAmI

In order to approximate the role of the cellular provider, we modified a localization service deployed on our campus. Developed by the Research Network Operations Center (RNOC), WhereAmI [35] is a location discovery service for devices connected to the campus network. It uses the associate and disassociate records along with various other available information from more than 2,700 WiFi access points on campus. Currently WhereAmI uses two discovery methods. The first method incorporates the proprietary discovery method of the Cisco Location Appliance [11], enabling the directional and distance discovery relative to the access point. This appliance is deployed in limited parts of the campus and can determine the location of the device with 10-meter accuracy. The other method simply uses the association of the access points and the signal strength of the associated devices to estimate the location. In both cases, the location server calculates the longitude and latitude values of the device using the known relative location of the associated access points.

WhereAmI's third-party lookup feature enables location query of other connected devices on the network. This specific feature is used for our implementation to emulate the location discovery function of the cellular provider. In addition to what is expected from the providers location function, WhereAmI's output will not only return the longitude and latitude values, but also the discovery method, error margin, and any other location labels such as the building name, room number, and access point name.

C. Performance

We investigate the performance of FrauVent for three different components of operation: delivery of the text message during the reactive phase, application overhead during user interaction and impact on deployed credit card infrastructure (i.e., PoS terminals).

The activity requiring the most time and effort was the location of the mobile device and subsequent delivery of our text message. While such messaging is generally viewed as a highly reliable, real-time communications channel, this store-and-forward architecture does not provide any actual delivery guarantees. A performance test [24] conducted in 2002 over the period of three months has shown that the average delivery time by all major cellular providers is 11.8 seconds with AT&T and Verizon Wireless being the top two carriers with 7.1 and 7.8 seconds respectively. The reliability of the delivery is also important and the study showed an average of 94.7% reliability rate with Verizon Wireless being the most reliable at 98.5%. Over the course of months dedicated to designing and implementing FrauVent, 100% of the text messages sent by our server arrived within 10 seconds with no dropped messages.

While mobile devices are becoming increasingly capable, they are still largely considered to be computationally limited devices when compared to desktop machines. Applications requiring a large amount of memory or computational effort may therefore function poorly on this platform. We used the Android process dump and the Eclipse memory analyzer to better characterize the footprint of our application. FrauVent uses approximately 30.8 KB of heap memory out of a maximum possible 16 MB per process. Because we used a number of native applications (e.g., Google Maps), the user experience remained interactive and did not display any noticeable latency.

[1] Android has since introduced an onscreen keyboard.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 7

Fig. 8. *Sample Phone Interface:* This is a preliminary version of the map interface. In the left image, the user is asked to confirm their location. After doing so, they are given details about the transaction (vendor, amount) and asked to confirm this as well. If the user confirms both checks, they will be asked to enter a PIN/password.

Although the majority of credit card authorization requests occur in under 15 seconds, the majority of PoS terminals allow transactions to hang for as long as 45 seconds. Accordingly, the extra time to deliver a text message (7-10 seconds) and for the user to respond is well within the default timeout period, meaning that the currently deployed PoS infrastructure can be augmented with an application such as FrauVent without requiring any changes. However, should our user study (see Section VII for a discussion of future work) reveal that more time is necessary, the bank can simply use mechanisms such as TCP KEEPALIVE messages to extend this time period.

VII. Discussion

Protocol designers are often required to make a number of tradeoffs based on practical issues. In this section, we briefly discuss some of those decisions. While we have designed our system to be easily adopted, we note that there are a number of mechanisms that real deployments could consider to individually strengthen or weaken the guarantees offered by our current system.

Chief among these tradeoffs is the use of public key cryptography. In their current form, our protocols have opted for the use of a symmetric key between banks and customers to protect the confidentiality of communications. While the use of public/private pairs by both sides would provide additional guarantees (e.g., non-reputability for both parties), overcoming the challenges of fully deploying a public key infrastructure is beyond the scope of this work [15]. We instead argue that users can easily learn the public keys of their bank and their provider when they download FrauVent. Moreover, a symmetric key can be generated and established between the two using an initial pairing/registration process.

Key to the widespread use of an application such as FrauVent is user acceptance. In particular, a wide range of users must feel comfortable interacting with their bank through the interfaces we are providing. From a usability standpoint, we argue that the work in this paper represents a significant step forward over prior works as we take advantage of the increasingly rich interfaces available to mobile phones. However, a number of issues remain. For instance, we have yet to determine the "ideal" range of the map presented to a user. While the current interface allows users to zoom out if the current view is too restrictive, we believe that automatically displaying the right amount of geographic detail will enhance acceptance. To meet this goal, we are currently discussing a large scale user acceptance test with a major US financial institution using actual customers. We believe that such a study requires its own research effort and therefore leave it to future work.

Our protocol currently requires cellular providers to sign approximations of user location. In particular, we aim to provide banks with a third-party's corroboration of location information to prevent a number of attacks (e.g., GPS jamming). We approximate these steps in our deployment using the WhereAmI infrastruture. However, such localization by the provider (and the subsequent signature) may be viewed as too expensive. This step can be eliminated by a provider at the potential risk that a client is unable to reliably detect its location. We leave the determination of the risk of such an attack to a provider considering cooperating with the deployment of FrauVent.

There are a number of other issues that we have not addressed. Malware targeting mobile devices is of growing importance. In particular, devices that fail to provide even the most basic security mechanisms (e.g., memory protection) are at risk of being compromised. We argue that the use of techniques such as the process isolation provided by Android's Dalvik virtual machine help reduce such threats. However, solving the problem of mobile malware is explicitly beyond the scope of this work.

Finally, we believe that this work makes an important step towards making mobile phones into more generally applicable authentication tokens. While we are by no means the first to suggest the use of this device, the vast majority of solutions simply use the cellular infrastructure as a means of delivering unauthenticated messages. Our scheme attempts to provide strong guarantees and leverage some of the strengths of this infrastructure - its ubiquity, processing ability and awareness of location information, to provide a framework for more secure interaction.

VIII. Conclusion

The cost of stolen financial credentials is becoming increasingly burdensome on both consumers and industries.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 8

While a wide array of two-factor authentication systems have been proposed to address many such issues, none have been widely adopted in this space for a variety of reasons. In this paper, we present FrauVent, an application designed to take advantage of the emerging set of expressive interfaces available to mobile phones to provide users with the context necessary to approve or reject suspicious transactions. We used a combination of protocol design, formal verification and a small scale deployment on phones running the Android platform to demonstrate the potential of our approach. We believe that this initial framework provides an important step towards effectively leveraging cellular infrastructure to improve the physical security of sensitive information.

ACKNOWLEDGEMENTS

This work was supported in part by the US National Science Foundation (CNS-0916047). Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] The State of New Hampshire Supreme Court: Estate of Helen Remsburg v. DocuSearch, Inc. http://epic.org/privacy/boyer/brief.html, November 2002.

[2] 3G Newsroom. U-TDOA technology standardized by 3GPP. *3GPP*, May 2003.

[3] M. Abadi and B. Blanchet. Analyzing Security Protocols with Secrecy Types and Logic Programs. In *Proceedings of the ACM Symposium on Principles of Programming Languages (POPL)*, 2002.

[4] M. Abadi and B. Blanchet. Computer-Assisted Verification of a Protocol for Certified Email. 2003.

[5] B. Blanchet. ProVerif: Cryptographic protocol verifier in the formal model. http://www.proverif.ens.fr/, 2003.

[6] B. Blanchet and A. Chaudhuri. Automated Formal Analysis of a Protocol for Secure File Sharing on Untrusted Storage. In *Proceedings of the IEEE Symposium on Security and Privacy (OAKLAND)*, 2008.

[7] L. Bruce. Skimming the cash out of your account. http://www.bankrate.com/brm/news/atm/20021004a.asp, 2003.

[8] calleridspoofing.info. calleridspoofing.info: The definitive resource on Caller ID spoofing. http://www.calleridspoofing.info/, 2008.

[9] CardTechnology. UAE ID Card To Support Iris Biometrics. http://www.cardtechnology.com/article.html?id=20070423V0XCZ91L, 2007.

[10] Cellular Telecommunications Industry Association. Wireless Quick Facts. http://www.ctia.org/advocacy/research/index.cfm/AID/10323, 2008.

[11] Cisco Systems, Inc. Cisco Wireless Location Appliance. www.cisco.com/en/US/prod/collateral/wireless/ps5755/ps6301/ps6386/product_data_sheet0900aecd80293728.pdf, 2007.

[12] CNET News. PayPal to offer password key fobs to users. http://news.cnet.com/PayPal-to-offer-password-key-fobs-to-users/2100-7355_3-6149722.html, 2007.

[13] D. Danchev. Phishers apply quality assurance, start validating credit card numbers. http://blogs.zdnet.com/security/?p=2095, 2008.

[14] S. Drimer and S. J. Murdoch. Keep Your Enemies Close: Distance Bounding Against Smartcard Relay Attacks. In *Proceedings of USENIX Security Symposium (SECURITY)*, 2007.

[15] C. Ellison and B. Schneier. Ten Risks of PKI: What You're Not Being Told About Public Key Infrastructure. *Computer Security Journal*, 16(1):1–7, 2000.

[16] Entrust. Multifactor Authentication with IdentityGuard. http://www.entrust.com/strong-authentication/identityguard/index.htm, 2009.

[17] EuroSmart. White Paper: Smart USB Token. www.rfidinfo.jp/whitepaper/741.pdf.

[18] J. Evers. T.J. Maxx hack exposes consumer data. http://news.cnet.com/T.J.-Maxx-hack-exposes-consumer-data/2100-1029_3-6151017.html, 2007.

[19] Experian Group. Increasing the effectiveness and efficiency of fraud detection for Bank Zachodni WBK. http://www.experian-da.com/news/enews_0903/Story10.html, 2009.

[20] J. Fredericks. Armenian Mob Imiplicated in Credit Card Scheme. http://www.beaconcast.com/articles/20090103, 2009.

[21] Google Inc. Google Android. http://code.google.com/android/, 2008.

[22] J. Heary. Black Hat 2008 Day 1 Phishers posting credit card info for all to see and a new DNS cache poisoning trick. http://www.networkworld.com/community/node/30822, 2008.

[23] HSN Consultants Inc. The Nilson Report. http://www.nilsonreport.com/recentissues.htm, 2009.

[24] Keynote Systems. AT&T Wireless and Verizon Wireless Lead in Performance on Keynote Wireless SMS Index for April through June. http://www.allbusiness.com/media-telecommunications/5990369-1.html, 2002.

[25] B. Krebs. 14 Arrested for Credit Card, Phishing Scams. http://voices.washingtonpost.com/securityfix/2006/11/14_arrested_for_credit_card_ph_1.html, 2006.

[26] B. Krebs. FBI Tightens Net Around Identity Theft Operations. http://www.washingtonpost.com/wp-dyn/content/article/2006/11/02/AR2006110201579.html, 2006.

[27] E. Larkin. Massive Theft of Credit Card Numbers Reported. http://www.pcworld.com/article/158003/heartlandtheft.html?tk=rss_news, 2009.

[28] J. Leyden. Visa trials PIN payment card to fight online fraud. http://www.theregister.co.uk/2008/11/10/visa_one_time_code_card/, 2008.

[29] Microsoft Corporation. Bank Employs SMS Messaging System to Protect Customers from Credit-Card Fraud. http://download.microsoft.com/download/8/f/0/8f02f193-320c-4d0c-b4df-6578e9254ad6/RaiffeisenBankCaseStudy.doc, 2006.

[30] MobiClear Ltd. http://www.mobiclear.com, 2008.

[31] OpenMoko. Openmoko - Open. Mobile. Free. http://openmoko.com, 2009.

[32] T. Pantin. Chip and pin system to be intrduced. http://www.thenational.ae/article/20090202/NATIONAL/836355499/0/NEWS, 2009.

[33] PayPal. PayPal Security Key. https://www.paypal.com/cgi-bin/webscr?cmd=xpt/Marketing_CommandDriven/securitycenter/PayPalSecurityKey-outside, 2009.

[34] V. Prevelakis and D. Spinellis. The Athens Affair. *IEEE Spectrum*, March 2005.

[35] Research Network Operations Center. WhereAmI. http://ANONYMIZED/, 2004.

[36] P. Roberts. US Bancorp teams up with VeriSign on banking security. http://lists.virus.org/cryptography-0410/msg00110.html, 2004.

[37] L. Rohde. MasterCard offers SMS to detect credt card fraud. http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=99660, 2005.

[38] RSA Security. RSA SecurID. http://www.rsa.com/node.aspx?id=1156, 2009.

[39] SiPix Imagining, Inc. World's First ISO Compliant Payment DisplayCard using SiPix and SmartDisplayer's Flexible Display Panel. http://www.businesswire.com/portal/site/google/index.jsp?ndmViewId=news_view&newsId=20060510006193&newsLang=en, 2006.

[40] M. Spirito. On the accuracy of cellular mobile station location estimation. *IEEE Transactions on Vehicular Technology*, 50(3), May 2001.

[41] A.-B. Stensgaard. Biometric breakthrough - credit cards secured with fingerprint recognition made feasible. http://www.ameinfo.com/58236.html, 2006.

[42] TMCnews. Credit Card Theft by Spyware Elite Toolbar (EliteBar); SaferSurf.com Warns: Spyware Elite Sends Credit Card Information to Third Parties. http://www.tmcnet.com/usubmit/2005/jul/1166104.htm, 2005.

[43] P. Traynor. Characterizing the Limitations of Third-Party EAS Over Cellular Text Messaging Services. Technical report, 3G Americas Whitepaper, 2008.

[44] P. Traynor, W. Enck, P. McDaniel, and T. La Porta. Exploiting Open Functionality in SMS-Capable Cellular Networks. *Journal of Computer Security (JCS)*, 2008.

[45] P. Traynor, W. Enck, P. McDaniel, and T. La Porta. Mitigating Attacks On Open Functionality in SMS-Capable Cellular Networks. *IEEE/ACM Transactions on Networking (TON)*, To appear 2009.

[46] P. Traynor, P. McDaniel, and T. La Porta. On Attack Causality in Internet-Connected Cellular Networks. In *Proceedings of the USENIX Security Symposium (SECURITY)*, 2007.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.

IPR2024-00400
Apple EX1045 Page 9

[47] US Department of Justice. Retail Hacking Ring Charged for Stealing and Distributing Credit and Debit Card Numbers from Major U.S. Retailers. http://www.usdoj.gov/opa/pr/2008/August/08-ag-689.html, 2008.
[48] VASCO. DigiPass Products. http://www.vasco.com/products/product.html?product=47, 2009.
[49] VeriSign. A Guide to Providing Proactive Protection to Consumer Online Transactions. http://www.verisign.com/authentication/authentication-resources/proactive-consumer-protection-guide/index.html, 2008.
[50] Visa Europe. Chip and PIN - The Facts and the Future. http://www.visaeurope.com/pressandmedia/factsheets/chipandpin.jsp, 2009.
[51] H. Weisbaum. Paying at the pump just got more risky. http://www.msnbc.msn.com/id/27085818/, 2008.
[52] B. Woolsey. Credit card 'phishing': What it means, how to prevent it. http://www.creditcards.com/credit-card-news/phishing-credit-card-scam-fraud-1282.php, 2008.
[53] W. Yan and D. Chiu. Enhancing E-Commerce Processes with Alerts and Web Services: A Case Study on Online Credit Card Payment Notification. In *Proceedings of the International Conference on Machine Learning and Cybernetics*, 2007.

Authorized licensed use limited to: Callie Pendergrass. Downloaded on December 13,2023 at 19:18:55 UTC from IEEE Xplore. Restrictions apply.