Open Access License Notice

Follow the Money Through Apple Pay

Dominicia Williams
Department of Computer Science
Norfolk State University
Norfolk, VA, USA
d.a.williams75096@spartans.nsu.edu

Yen-Hung (Frank) Hu
Department of Computer Science
Norfolk State University
Norfolk, VA, USA
yhu@nsu.edu

Mary Ann Hoppa
Department of Computer Science
Norfolk State University
Norfolk, VA, USA
mahoppa@nsu.edu

***Abstract*—Rapid growth in the number of mobile phones and their users has brought ecommerce applications and mobile payments to the forefront along with raising significant new cybersecurity concerns. Consumer enthusiasm for "tap-and-go" purchases must be tempered with knowledge about new risks and responsibilities that come along with these payment technologies. This paper highlights and analyzes key risks within end-to-end mobile-payment transactions through the lens of one of the most popular services: Apple Pay. Hackers are relentlessly adapting their ploys to breach these payment systems. Proactive approaches are identified to better secure vulnerabilities in smartphones, networks, communication, consumers, merchants and banks, along with practical, proactive countermeasure and action plans.**

Keywords—Apple Pay, Mobile Commerce (mCommerce), Near Field Communication (NFC)

I. Introduction

The wide penetration and personal nature of mobile phones, the overall stability of mobile communication, technology, and positive experiences with mobile commerce (mCommerce) payments have favored the adoption of mobile solutions for financial services [1].

In 2014, Apple Pay launched and has been built into every iPhone since 6/6 Plus, including the newly released iPhone X. The iPhone solution also includes a Near Field Communication (NFC) antenna (the standard for all contactless payments); the convenience and security of Touch ID and a Secure Element chip (SE). These features work together toward one goal: the ability to encrypt and securely store all payment information. All credit cards a consumer adds into their Apple Pay can be safely stored through Passbook. There are hundreds and millions of credit cards and debit cards from customers in their iTunes Store accounts. When a customer purchases an iPhone 6 or newer, they can place the card on file by inputting the card information, manually, or simply taking a picture of the card. With just a "touch", you can easily make payments through your mobile device [2].

Apple Pay originated in the United States with credit cards and debit cards from three major networks: American Express, MasterCard, and Visa. Moreover, Apple Pay is connected to the largest issuing banks in the United States including Citi Bank, Bank of America, Capital One, Wells Fargo, and Chase Bank, comprising 83 percent of all credit card volume across the nation. Apple Pay can be used in over 220,000 U.S. merchant locations that accept contactless payments and, since 2014, has been networking with some of the largest retailers to enable Apple Pay in all locations, nationally and internationally. Macy's, Bloomingdale, Walgreens, Staples, Subway, McDonalds, Whole Foods Market, Apple retail stores, and Disney are some of the businesses that welcome Apple Pay for their fast and reliable services [3].

Apple Pay security is realized through both hardware and software [4]. When a new card is added, a device-only account number is created for it and stored safely in the secure element; the card number is never stored or share with the merchant. For each consumer transaction, a one-time payment number is generated along with a dynamic security code.

Security and privacy are at the core of Apple Pay. Apple does not track what purchases the consumer makes, where the consumer makes them, nor the cost of the purchases. The transaction is among the consumer, the merchant, and the consumer's bank. According to Apple's encryption guidelines, even the cashier does not see the name of the consumer, credit card information, nor the security code [4].

The goals of this project were to develop proactive mitigation and objective strategies to:

- Develop an understanding of the many risks of mobile-payment technology methods and any associated lack of confidentiality.
- Identify damages caused by Apple Pay when affected by cybercrimes.
- Study detailed transaction mechanism of the Apple Pay system and identify security vulnerabilities of each.
- Identify if there are some solutions or a proactive approach to securing vulnerabilities and networks for Apple Pay.
- Summarize and seek optimized solutions for each vulnerability.
- Recommend policies for early detections of fraudulent activities.

The remainder of this paper is organized as follows: Section II summarizes related work and recent efforts to provide perspective on the scope and importance of Apple Mobile Payments to the mCommerce ecosystem. Section III

discusses known examples of Apple Pay attacks and threats. Section IV discusses relevant updates to Apple iOS. Section V introduces actions planning to address risks. Section VI proposes recommendations. Section VII concludes the paper with some reflections on findings and suggestions for future work to build upon them.

II. Concept of Apple Mobile Payments

The popularity of mobile payments is growing at an amazing rate. There are five million ecommerce transactions within the United States alone each day, representing over $1 billion of online purchases. On Black Friday 2016, online sales from mobile devices totaled $1.2 billion, or 36 percent of the day's total sales. This was an increase of 33 percent over 2015. By 2019, worldwide mobile payments are predicted to surpass $1 trillion [3].

With Apple Pay there is "one-touch" check out, no card number entry, no disclosure of addresses when shopping online, and no sharing of card information with the merchant. Online payments and physical Point of Sale (POS) are the two methods of mobile payments. Physical POS refers to methods such as Apple and Android Pay that are processed at checkout terminals in stores. Companies such as MasterPass (MasterCard), Samsung Pay and Chase Pay use these services, but most physical POS payments use NFC technology that is built into many smartphones. This is the same technology that is used for mobile payments at brick-and-mortar retailers. Card information is not stored on the smartphone but creates a token that replaces card details to realize a confidential transaction. For example, Apple Pay requires strict security measures whereby all transactions must be verified with biometric authentication or a passcode. However, mobile payment fraud still can occur.

Mobile payments are becoming more popular, but they still face some high barriers, such as consumers' continued loyalty to traditional payment methods and fragmented acceptance among merchants [5]. New data from the PULSE 2016 Debit Issuer study shows that despite increased availability, debit users are mostly uninterested in mobile wallets [6]. In 2016, a survey was taken where 67 percent of respondents expressed concerned about the security of mobile payments. Fiserv [7] found that another 47 percent of consumers avoid mobile payments because they do not trust the advancement of technology with their confidential information. A study from Auriemma Counseling Group found that 74 percent of consumers want to avoid the use of mobile payment, collectively, due to the risks they believe their devices will be exposed to [6]. Moreover, many consumers have decided to remain loyal and dependent on traditional payment methods.

A. Definitions and Characteristics of Apple Mobile Payment

Table I shows the types of mobile payments that currently are available and have reached their peak within the U.S. Mobile payments are centralized among consumers and merchants and involve direct purchases of goods and services that can be account-based and POS. Apple Pay is considered a proximity-technology involved mobile payment in which payment credentials are stored in the mobile device and exchanged over the air, based on NFC technology, with a dedicated and compatible payment terminal [8]. Additionally, this acts as a contactless reader or Personal Identification Number (PIN), for authentication purposes, in which the consumer purchases goods and services. Apple Pay can send and receive data in which it is highly aligned with the use of trusted computing media such as Subscriber Identity Module (SIM) cards and Trusted Platform Modules (TPM).

TABLE I. Types of Mobile Payments

Type	Technology Involved
Proximity Payment • Refers to contactless payment • Payment credential stored in mobile device and exchanged over air • Mobile device acts as a contactless payment card *Remote Payment* • Covers payments that take place via mobile browser or smartphone application • Mobile device is used to authenticate personal information stored remotely • Payment transactions: face-to-face and vending machine transactions	• Mobile phone is used by the consumer to pay for goods via contactless reader, text-based, or personal identification number using NFC • NFC: communication between consumer device, payment scheme operator, and retail merchant • NFC Compatible devices can send/receive data

B. Ecosystem of Apple Mobile Payments

Figure 1 presents the cycle of the functioning of Apple Pay's NFC transaction through mobile payments. This chart depicts the ecosystem of mobile payment functions. First, authorization of the NFC proximity mobile payment via an existing Payment Service Provider (PSP) network is needed. The financial institution prepares the account data and then transfers the payment information to Trusted Service Manager (TSM) [8]. Secondly, TSM manages the deployment of mobile applications and delivers consumers' payment information over-the-air (OTA) through the mobile network to the secure element in the mobile phone. Once the payment is in phone, the consumer can utilize the mobile device as a contactless payment with merchants who accept this specific payment method.

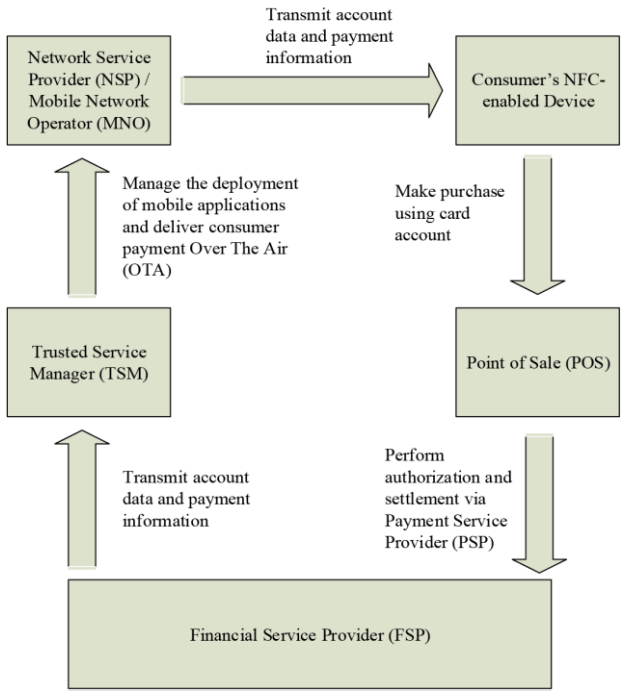

Fig. 1. Lifecycle of a Bank-Centric NFC Mobile Payment [8]

Table II presents various types of mobile payment services. They comprise independent communication service providers that own the complete telecom infrastructure for hosting and managing mobile communications among subscribed mobile users with users in the same and external wireless and wired telecom networks [9]. This creates a network of high-end telecommunication devices, specialized software, and client-end subscriber modules to issue end- to-end communication between wired and wireless telecom end-user devices. Mobile Network Operators (MNOs) install base stations, while the mobile subscribers use a circuit-like chip in iPhones to access network services.

TABLE II. MOBILE PAYMENT SERVICES

Service Provider Type	Services
Hybrid-Collaborative	Short Message Service (SMS) based payment service targeting the unbanked, prepaid mobile subscribers Google Checkout
Mobile Network Operator	SMS based system that has NFC system for mobile ticketing for mobile transport Mobile wallet services
Independent Payment Services	Peer to Peer (P2P) mobile payment company that enables mobile phone users to send/receive money through devices P2P money transfers from the sender's bank account to the recipients' bank account

C. *Security and Privacy Policy of Apple Mobile Payment*

Privacy is important when confidential information is being distributed across networks. Apple holds confidential information per device, encrypted, in which it is safely, individually available to the consumer [10]. Apple ensures privacy is effective through two policies:

1. No backdoors: There can be no backdoors (which may allow malicious activity in) in any software.
2. Encryption: According to the Legal Privacy Policy [4], Apple's websites, interactive application such as Apple Pay, online services, etc. use encryption such as Transport Layer Security (TLS). Encryption is a "must" in today's world.

According to Yunusov [11] Apple Pay's security measures also include using a separate microprocessor for payments, known as Secure Enclave, so that credit card data is not stored on the device or transmitted in plaintext during payments.

D. *Concerns of Apple Mobile Payment*

Apple Pay has received a fair amount of praise since the launch of mobile payment in September 2014. Headlines raved about this innovative category of service that has transformed mobile payments with an easy, secure, and private way to pay. Reports suggest that a lapse in verification between banks and Apple OTA transactions due to vulnerabilities within the NFC and many other payment mechanisms allows thieves to compromise information involved in mobile payments. This brings payments under the manifold protection of iOS, whether Apple's much-debated encryption or largely successful repellence of malware. Although Apple's mobile payment service can provide consumers with various benefits, it also introduces security concerns and vulnerabilities [12].

While mobile presents enticing business opportunities; it also stretches the boundaries of the threat landscape, expanding the attack surface to an increasing number of threats against the mobile banking revolution [3]. Many security researchers have confirmed that mobile provides criminals an "entrance" to stealing credit card details and hijacking transaction information. As an example, criminals who may acquire pilfered credit card data can add this information to their own Apple Pay account. Furthermore, mobile network operators are losing control of the mobile payment ecosystem [13]. Not only is the consumer and their lack of knowledge of security standards at fault, but also banks lack sufficient verification of information.

Financial institutions and MNOs compete to be the entity that will hold the customer account and receive payments. There are two models: bank-centric and nonbank-centric. The bank-centric model involves a customer account held by the bank that handles issues involving liability, transaction monitoring for fraud detection, and anti-money laundering. Apple Pay is bank centric. In the nonbank-centric model, the customer account is held at a nonfinancial organization such as an MNO or a third-party payment service [8]. When a

payment is initiated, it is imperative for the consumer's bank to authorize all transactions. However, important regulations, security, and profit-sharing question have been raised. With mobile payments on the rise, cyber criminals have begun to target their efforts against mobile opportunities. Which entity will be responsible for the regulation of these services if breaches take place? National telecommunication networks? Or national banking?

Apple Pay has changed business in the field of communication and now has a method of generating financial transactions on and off the web. This in turn has helped consumers increasingly familiarize themselves with mobile payments and become accustomed to its conveniences.

III. Apple Mobile Payment Attacks and Threats

Apple Pay may have numerous benefits when it comes to ease of use and security. However, according to researchers from the anti-fraud firm Pindrop, Apple Pay and banking partners still are not doing enough to preventing stolen credit cards, citing vulnerabilities such as SSL interception, security gaps in the secure element, and ongoing use of jailbroken iPhones [14]. When a customer adds a card, Apple connects with the bank sending them encrypted credit card data. The bank, then, imposes its own authentication checks which may require a phone call where the consumer may have to provide additional information for authentication purposes.

Pindrop researcher, David Dewey, tested out his theories regarding the safety of Apple Pay by experimenting with bank cards donated by various banks. While not revealing all the results, Dewey did disclose that he remains skeptical that banks are investing sufficient effort to prevent stolen credit cards. According to Dewey [14], Apple Pay provides the easiest work around for fraudsters to evade the protections offered by Europay, MasterCard and Visa (EMV) chips.

Apple does not implement the "rate limiting" service that rejects hackers from making too many guesses as they attempt to gain access. In other words, Apple Pay does not prevent brute force attempts. Researcher Dewey constructed a tool that would guess the correct Card Verification Values (CVV) number of a credit card, at a rapid pace. There are only 1000 different combinations of three digits, something a computer can run through in seconds [14]. Dewey stated that communication going through Apple Pay is blinded, leaving providers with no protection against brute force attacks by hackers who may try to guess the CVV code [14] [15].

Through research it has been discovered that the following are different types of attacks and threats that allow Apple Pay users to fall victim to hackers.

A. Attack I: Apple Server

iOS is designed to be reliable and secure from the moment the device is in use, with built-in security features to help protect access to personal information and data. However, experts state that around 2 percent of iPhone users make unauthorized modifications to iOS – so-called "jailbreaking" – to allow customizations and to add applications that have not been approved by Apple [16]. Jailbreaking undermines the original security features implemented by Apple [17], making them susceptible to attacks including hijackings and malware installations.

Hackers initially infect a jailbroken device with malware, then eventually acquire root privileges to gain full access to the user's device. An attacker can run tools like Cycript, GDB and Snoop-it to perform runtime analysis and steal sensitive data, including intercepting traffic like payment data en route to an Apple Server [16], damage the device, attack the network through FaceTime, and many other nefarious deeds.

B. Attack II: SSL Transaction Traffic

The Secure Socket Layer (SSL) is the standard technology for keeping an internet connection secure and safeguarding any sensitive data exchanged among systems. This prevents criminals from reading and modifying *any* information transferred, not just personal details [18]. SSL creates a secure connection through public, private, and session keys. Encrypting and decrypting with private and public keys can take a lot of processing power but is used only during the SSL Handshake to create a symmetric session key [19]. After the secure connection is established, the session key is used to encrypt all transmitted data. The browser from the devices connects to a sever to begin the transaction, secured with the SSL. The server sends a copy of its SSL certificate along with the server's public key [20].

However, hackers have mastered the art of hijacking the transaction and manipulating the traffic before the server has the opportunity to decrypt the symmetric session key. However, in the most recent years, SSL has fallen vulnerable to hackers. An attack can be performed against Apple devices by exploiting and taking advantage of jailbroken devices to inject malware and then intercept and manipulate SSL transaction traffic that users perform using Apple Pay. Hackings can intercept SSL transaction traffic, tamper with transaction data and change the amount or currency being paid using Apple Pay [21].

The first step in this attack, where hackers can compromise data, is stealing the payment token from a victim's phone. Some consumers are not aware of the risks that results from using public Wi-Fi. As remarked earlier, hackers can offer their own "fake" Wi-Fi hotspot and ask users to create a profile. This, give hackers the opportunity to steal the Apple Pay cryptogram, the key to encrypting the data. Since the delivery information is sent in clear text, hackers can use an intercepted cryptogram to make payments on the same website where the victim charged transactions [11]. These vulnerabilities can lead to additional damages such as malware dispersed throughout the network where it will spread rapidly.

To patch these vulnerabilities, consumers must disable outdated SSL servers and continuously upgrade their devices to remain in compliance with the most up-to-date security measures.

The followings are some incidents caused by SSL vulnerabilities.

1) POODLE (CVE-2014-3566)

According to the National Vulnerability Database (NVD) [22], Padding Oracle On Downgraded Legacy Encryption (POODLE) was published in October 2014 and takes advantage of two vulnerabilities. First, some Apple users still support SSL 3.0 for interoperability and compatibility with legacy systems. In this case, victims voluntarily interact with attack mechanisms resulting in unauthorized disclosure of information. The second vulnerability relates to Block Padding in SSL v3.0. POODLE uses nondeterministic Cipher Block Chaining (CBC) padding; this makes it easier for a Man-in-The-Middle (MitM) attacker to obtain clear-text data via a padding-oracle attack such as POODLE [23].

When the Apple Pay user initiates the Handshake, and sends the list of supported SSL versions, the attacker can intercept the traffic and then perform the MitM attack. This impersonates the Server until the Client agrees to downgrade the connection to the vulnerable state [24]. Once the connection between the Apple User and Server is established on the vulnerable SSL, the attacker can then perform the POODLE attack. Moreover, the vulnerability exists in CBC mode. Since Block Ciphers have fixed length, padding is added to fill the extra space. The issue here is the padding value is ignored by the Server which merely checks if padding length is accurate along with Message Authentication Code (MAC) of the plaintext [25]. In other words, the receiver will not be able to verify if the padding value has been manipulated in transit. The attacker thus will have the opportunity to decipher the plaintext value of the encryption block by modifying the padding bytes, and then seeing the corresponding response from the server.

Until systems are patched, mitigation steps need to be taken to initiate an action plan. To patch against POODLE and keep it from affecting Apple Pay, users and merchants need to implement Intrusion Prevention Systems (IPS) to secure network traffic through network scanning [25].

2) BEAST (CVE-2011-3389)

The Browser Exploit Against SSL/TLS (BEAST) attack affects SLL 3.0 and TLS 1.0. An attacker can decrypt data exchanged between two parties by taking advantage of a vulnerability in the implementation of the CBC mode in TLS 1.0. This is the tool that allows hackers to perform an attack. According to the NVD, the SSL protocol encrypts data that allows MitM attackers to obtain plaintext Hypertext Transfer Protocol (HTTP) headers via a Block-wise Chosen Boundary Attack (BCBA) on an HTTP Secure (HTTPS) session [26]. Using MitM, the attacker can inject packets into the SSL stream; the attacker guesses the initialization vector used in XORing with the message and compare the results to the ones of the block the attackers want to "decrypt" [23].

Despite the client-to-server relationship between Apple Pay users and merchants, these attacks can take place by simply browsing the web on public Wi-Fi. For this to be a successful attack, hackers must have control of the Apple Pay user's browser. Hardening TLS 1.1 and banning the Java Plug-in from the browser will prevent this attack from occurring [27].

3) CRIME (CVE-2012-4929)

Compression Ratio Info-lead Made Easy (CRIME) is a vulnerability found in TLS compression; meaning the connection can be established without any compression [23]. This feature is known to reduce bandwidth usage while preserving integrity and security when exchanging large amounts of information.

An attacker targets a victim's network to hijack. The Apple Pay user may have signed into the browser through a public Wi-Fi that contains malicious JavaScript and is controlled by the attacker. Then, the script will initiate a connection to a third party so the attacker can inject plaintext into the victim's cookies and then monitor the size of the response [28]. If the size of the response is lower than the initial response, it means the character the attacker injected is contained in the cookie value. Using this method, an attacker can brute force the cookie's value based on the feedback from the merchant.

4) BREACH (CVE-2013-3587)

Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext (BREACH) is similar to CRIME, but targets HTTP compression where TLS compression is not required for this attack to be execute [23]. An attacker forces the victim's browser to connect to the TLS enabled third party network as they are monitoring the traffic between the Apple Pay consumer and merchant (server) by performing a MitM attack. Taken together, these factors constitute a vulnerable web application [29]:

- Being server from a server that utilizes HTTP-level compression
- Reflect user-input in HTTP response bodies
- Reflect a secret in HTTP response bodies

To prevent this attack from happening consumers must disable HTTP compression, mask secrets, protect vulnerable pages with Cross-Site Request Forgery (CSRF), and rate limit requests.

5) Heartbleed (CVE-2014-0160)

This attack compromises the TLS heartbeat extension [30] [23]. Heartbeat is found in the heartbeat extension of the cryptography library OpenSSL. The TLS heartbeat extension is used as a method between two parties to ensure the connection is not closed. The Apple Pay user sends a request to the retailer with a payload that contains the data-size of the data. The retailer must respond with the exact same request containing the data and size to reciprocate what the Apple Pay user requested. However, if the Apple Pay user sends a falsified data length, the retailer would respond with the data received by the client – including random data from its memory to meet the length requirements contingent of the

Apple Pay user's request [31]. There are known cases where the retailer's private key leaked through the Heartbeat vulnerability, which means the attacker would be able to decrypt all the traffic of the server. The flaw allows a remote attack to retrieve private memory of an application that uses the vulnerable OpenSSL library [32].

Maintaining updated iOS and SSL is imperative to preventing security breaches. It also is important that retailers implement TLS to keep data secure over a network. There are some instances where badly configured servers place Apple Pay users at risk and expose confidential information. As a solution, OpenSSL 1.0.1 has been released to patch the damages, vulnerabilities, and leaks.

C. Attack III: Masque Attack

The 2015 release of iOS 8.4 fixed various vulnerabilities that allowed attackers to deploy two Masque Attacks: CVE-2015-3722/3725 and CVE-2015-3725. These exploits are known as Manifest Masque and Extension and can be used to demolish apps and other resources Apple has to offer, such as Apple Pay, Apple Watch, Apple Heath, etc. These exploits also have the ability to destroy and corrupt the app data container. Plugin Masque bypasses iOS security measures and hijacks Virtual Private Network (VPN) traffic.

A year after the launch of Apple Pay, one-third of iOS devices had not been updated to version 8.1.3 or above. Consequently, five months after the release of 8.1.3, these devices remained vulnerable to all Masque Attacks [31]. The Table below show three different types of Masque Attack that are threats to Apple Pay.

TABLE III. MASQUE ATTACKS [33] [31]

Name	Consequences Disclosed	Mitigation Status
Manifest Masque	• Demolish apps (such as Apple Pay, Apple Watch, etc.) during OTA installations	Partially fixed in iOS 8.4
Extension Masque	• Access to other application data • Prevent Apple Pay transactions and access to consumer's own data	Partially fixed in iOS 8.4
Plugin Masque	• Bypass prompt of trust • Bypass VPN plugin entitlement • Replace an existing VPN plugin • Hijack device traffic • Prevent device from rebooting • Exploit more kernel vulnerabilities	Fixed in iOS 8.1.3

1) Manifest Masque Attack

In 2014, Apple was notified of this vulnerability. Manifest Masque Attack leverages the CVE-2015-3722/3725 vulnerability to demolish an existing app on iOS when a victim installs an in-house iOS app. The demolish app (the attacker target) can either be a regular downloaded from official App Store or even an important system app, such as Apple Watch, Apple Pay, App Store, Safari, Settings, and so on during OTA installations [34]. Additionally, this vulnerability affects all version iOS 7.x and iOS 8.x devices which are Apple Pay compatible yet still vulnerable to being attacked due to being just "partially patched."

2) Extension Masque Attack

This attack takes advantage of the introduction of app extensions in iOS 8. While an app extension can execute code and is restricted to access data within its data container, a malicious extension using the same bundle identifier as the target app could give the attacker full access to the data container of the target app [35]. In this attack, an attacker can lure the victim to install an in-house app by using enterprise provisioning from a website to enable the malicious extension of the in-house app on the victim's device, thus leading to the end game of stealing data [34] [36]. On June 14, 2015, security researchers validated various severe issues on OS X which can, in fact, be leveraged by an attacker to steal all data in a target app's data container. Apple was notified and fixed this issue as part of CVE-2015-3725.

3) Plugin Masque

The final attack is the Plugin Masque. The vulnerability of Plug-in was disclosed to Apple in November 2014 and as a result it was patched on iOS 8.1.3 when Apple patched App Masque [34]. This attack is known to be more troubling than Manifest Masque and Extension Masque, because it allows for the replacement of the VPN plugin. In turn, this gives an attacker the ability to monitor all the network traffic on the device, not just Apple Pay, during transactions. It can perform authorized operations, including VPN traffic, without the user's knowledge. Although patched since iOS 8.1.3, it is still causing problems.

D. Attack Via Distributed Denial of Service Attack (DDoS)

The DDoS attack affects the availability of network resources of services by preventing Apple Pay users from accessing network assets; denying the use of services from authorized users [37]. In addition to denial, this attack delays time critical operation by preventing the Apple Pay customer, or merchant, from responding to a user's request. Hackers create these delays through resource exhaustion, where they exhaust all available bandwidth, disk space, or memory capacity.

There are three types of threats that can flood this service: consuming system resources; wasting the communication link by repeatedly downloading a large file from the server; and using the Structured Query Language (SQL) injections [38]. These flood attacks can be launched from botnet, viruses, or open Denial of Service (DoS) tools to disrupt the

network service. Malicious users launch such attacks by sending a huge number of bogus requests to the servers to consume the processing power from the Apple Pay user to the merchant and flood the network bandwidth.

Apple Pay users are vulnerable to the Connectionless Volumetric Attack, where the attack does not require a session to be established before sending data packets to the Apple Pay user [39]. Volumetric Attacks are known as floods where it congests a system sending an abundant amount of traffic, on a network, that it overwhelms the bandwidth. This attack has become a frequent menace as it is commonly used to exact revenge, conduct extortion, and even to wage cyber war [40].

Causing a system malfunction during the transaction between the Apple Pay consumer and merchant creates damages to bandwidth affecting the connectivity of the Apple Pay user through flooding [41]. Moreover, hardware becomes corrupted through amplification-based flood attacks where the adversary sends requests with spoofed IP addresses in a reflection manner to a large number of reflectors exploiting the IP packet broadcast feature.

Many merchants are investing in an open source memcached project to eliminate this attack vector. The memcached 1.5.6 update disables the User Datagram Protocol (UDP) protocol, which is what DDoS attacker are using to amplify attacks. This "kill switch" flushes all commands to merchant networks and decreases the vulnerability of network traffic [42].

E. Additional Vulnerabilities, Risks, Threats, and Countermeasures

There is a synergy of both business risks and technical risks encountered as Apple Pay continues to become adopted into modern industrial society. There are some challenges and cost-value considerations for businesses preventing them from investing into Apple Pay services. Fraudsters are targeting Apple Pay; so upfront analysis and countermeasures are imperative to mitigate the risk to these devices. Traditional risks involve denial or theft of services leading to the loss of revenue, negative reputation, and lack of confidentiality. Emerging risks involve the use of mobile payment leading to the loss of revenue, exposure of confidentiality, and theft of transfers through transactions.

Risks for Apple Pay depend on the role of the entity user, network, or communication provider, or payment service providers [8]. Listed below in Tables IV and V are the various types of threats, risks, and vulnerabilities that are likely to play a role in potential malicious attacks on Apple Pay. These data further confirm that users and service providers have weaknesses that make them vulnerable to malicious attacks. A user is likely to be vulnerable to an attack through OTA transmission between an iPhone and POS due to the interception of traffic [43]. Apple Pay is susceptible to identity theft, information disclosure, and a potential re-launch of this attack will likely take place if countermeasures are not in place. A TPM, secured protocols, and data encryption need to be consistently enforced on the network.

Service providers are known to be the "backdoor" to mobile payment compromises. POS accepting OTA transmissions fall victim to malicious party flooding on POS systems with meaningless requests [8]. Consequently, this leads to the risk of DoS attack. Another vulnerability occurs when POS devices are installed at merchant premises; then masquerade attacks become a threat leading to POS tampering. Services and message modifications become a risk to consumers in which data traffic may be rerouted to complete a hacker's end game.

TABLE IV. APPLE MOBILE PAYMENT RISKS TO USERS [43] [8]

Vulnerability	Threat	Risk	Counter measures
OTA transmission between phone and POS	Interception of traffic	Identity theft, Information disclosure, Relay attacks	TPM, secure protocols, encryption
Inadvertent installation of malicious software on mobile phone	Interception of authentication data	Theft of authentication parameters, Information disclosure, Transaction repudiation	Authentication of both user and application, TPM
Absence of two-factor authentication	User masquerading	Fraudulent transactions	Two-factor authentication

TABLE V. APPLE MOBILE PAYMENT RISKS UPON SERVICE PROVIDER [43] [8]

Vulnerability	Threat	Risk	Counter measures
POS system accepts OTA transmissions	Malicious party floods POS system with meaningles requests	DoS	Request filtering at reader based on mobile device reader relative geometry
POS devices are installed at merchant premises	Masqueradea ttacks, Tampering with POS	Theft of service, Relay attack, Message modification	POS vendor vetting, Message authentications
Lack of digital rights management on mobile device	Mobile device user illegally distributes content	Theft of content and digital piracy, Risk to provider for	Digital rights management (DRM) incorporated in smartphone TPM design,

Vulnerability	Threat	Risk	Counter measures
		digital rights infringement	Cryptographically supported DRM
Global System for Mobile (GSM) communication encryption for On The Scene (OTS) transmission	Message modification, Relay of transactions, Evasion of fraud control	Theft of service of content	Strong cryptographic protocols, SMS messaging authentications, Encryption

IV. Apple Mobile Payment Updates

In January 2015, iOS 8.1.3 was released. App Masque, Uniform Resource Locator (URL) Masque, and Plugin Masque issues were patched or partially fixed [34]. Recently, researchers monitoring iOS web traffic in high-profile networks showed that one-third of all iOS traffic is still vulnerable to all the Masque Attacks [33]. Consumers should continuously update their Apple devices, when prompted, to ensure software remains up-to-date, and thereby reduce the potential for malicious attempts that can also affect Apple Pay.

A. Apple iOS Internal Feature

Apple is said to place heavy emphasis on security within Apple Pay to ensure consumers' payment information is safe and protected. When a credit or credit card is scanned into the Wallet for use with Apple Pay, it is assigned a unique device number, or a 'token' which is stored in the phone as a "code" than a card number [44] [45]. There is a special chip, secure element, containing payment information data that is said to never become exposed or uploaded to iCloud to Apple's servers.

When transactions are initiated, the Device Account Number (DAN) is sent via NFC with a dynamic security code; both which are needed for a successful transaction. The security code is a one-time use cryptogram that replaces the credit card's Credit Card Validation (CCV) functions to ensure that a transaction processes accordingly. As mentioned before, Apple Pay fosters secure enclave and secure element; both storing the payment applet certified by the payment networks and specializes in encrypted cardholder data and keys.

In addition to security, dynamic security codes and DAN, such as tokens and cryptograms are built into the NFC specification. However, recent studies have proven that Apple's "security" standards are not effective in protective Apple Pay users as they are portrayed.

B. Storing Keys in Secure Element

Apple has built in a security method designed to protect user data through Secure Enclave. This secure element ensures that a user's sensitive payment data is stored only on a user's device. To make a payment, an alias is generated that the processing backend can recognize. When a user taps their device against the POS to pay, that alias alone is transmitted along with a cryptographic code. The code is decrypted by the backend, which then compares the alias to the one it stores [46].

Maintaining a private key in a keychain is an amazing asset to ensure privacy and security. Secure Enclave is a hardware-based key manager that is isolated from the main processor to provide an extra layer of security [47]. As an example, when a consumer stores a private key in the Secure Enclave, the likelihood of the key becoming compromised is slim. Instead, the user instructs the Secure Enclave to create the key, securely store the key, and perform operations with the key. The consumer only receives the output of operation such as encrypted data or a cryptographic signature verification outcome.

Free Wi-Fi hotspot services, in public settings, could allow an attacker into the consumer's mobile devices. Security researchers have alerted Apple users about potential security flaws regarding the use of free Wi-Fi since exploiters can hack into iOS users' operating systems and/or set up a rogue Wi-Fi spot. This can lead consumers to insert their credit card information which attackers can then intercept. According to researchers, spoofers can loaf around a POS machine with an Apple Pay terminal and continuously launch such an attack.

Access to the secure element also creates a weakness the Relay Attack. This attack cannot be prevented by the application layer cryptography protocol. The timing requirements by International Organization for Standardization (ISO) 14443 are too lax to prevent relay over longer channels [48]. Possible countermeasures are shielding countless interface and distance bounding protocols; but this requires faster communication channels. Application accesses the secure element and relays Application Protocol Data Unit (APDU) commands/responses over a network interface.

Adding a secure element to a mobile phone opens a new attack vector, such as DoS and Relay Attack, which has fallen to negligence of being considered.

C. Near Field Communication

POS on the NFC Interface is potentially vulnerable to relay attacks for low- value payments. Some merchants welcome cloud-based systems where intercepted data may become spoofed, manipulating the identity of the user [49]. This factor leads to one of the vulnerabilities of NFC. When consumers send their information through the network, attackers tamper with the data with the possibility of being execute by the hacker's collection of keys. The biggest challenges are secure access and authentication of the user to the cloud.

To patch these vulnerabilities, Apple and many merchants have implemented security mechanisms such as tokenization and Point-to-Point Encryption (P2PE) [49] [50] [51]. Tokenization is known to offer a substantial measure of

security for financial transactions, as opposed to host-based card emulation which is unsafe due to sensitive data being stored on mobile devices. Even without equipment or skill, attackers would be able to intercept the SSL transaction traffic and also manipulate data. Consequently, Apple Pay uses the EMV Payment Tokenization Specification to offer secure payment transactions [17].

To patch the vulnerabilities of NFC, tokenization captures card information, stores, and secures the data and keys on the mobile device [52]. It is also used during payment verification to identify the user and the keys of the payment product. Then, the user's data is tokenized to devaluate the contained sensitive information, discouraging attackers from hacking. Tokenization makes use of session keys, single keys, or limited use keys that have to be validated and confirmed. However, tokenization and other payment card security technologies are only as secure as their implementation.

V. Action Planning to Address Risks in Apple Mobile Payment

Fraud is an intentional deception or misrepresentation intended to result in personal or financial gain. Security threats in mCommerce may be passive (e.g., information being monitored and released for fraudulent purposes), or active (e.g., modification of information through DoS and unauthorized access) [38]. Apple Pay brings new opportunities and new risks.

Due to the many parties that are involved in making a single transaction during a mobile payment, the network is left exposed and vulnerable to risks and threats. This can be exacerbated if important services are outsourced to potentially unregulated third parties without clear lines of accountability and oversight, or which are located overseas [8]. Multiparty transaction environments are conducive to exploitation by fraudsters using technological and sociological attack if the appropriate protection mechanisms and accountability controls are not established throughout Apple Pay's mobile payment ecosystem. With effective planning, favorable circumstances exist to make security an element of Apple Pay systems.

The fraud that can occur in the mCommerce environment are specific to e-payment systems through Apple Pay. It is imperative that Financial Service Providers (FSPs), PSPs, and Network Service Providers (NSPs) employ appropriate protections, safeguards and privacy and security governance programs. Transactions being undertaken, in an assurance manner by the authorized person, is a concern for many stakeholders. Some of the intrusions take place in mobile commerce environments; attempts from competitors, entry attempts into customer's private accounts, and attempts to spoil the reputation of the merchant vendor [38]. However, using two-factor authentication provides a substantial amount of identity protection for the consumer and high assurance of confidentiality for the merchant.

In the case of Apple Pay, through NFC transactions, protection from transactions originating from unauthorized users can be accomplished by the use of dynamic CVV [4]. The NFC chip on iPhone supports CVV as opposed to the CVV located on the magnetic strip of a credit card. If a bogus mobile device is used with Apple Pay, it will display the incorrect CVV and the transaction will be unsuccessful. In turn, this will protect the consumer, the merchant, and the service provider from foul play.

Techniques analogous to SSL should be used to ensure that only legitimate POS or service providers interact with mobile phones [20]. This represents a large pool of issues in relation to trustworthiness or identities and credentials for Apple Pay users. In a 2011 White House publication, these issues and potential strategies were discussed as part of a national strategy for trusted identities in cyberspace [8].

Data classification during transmission and storage at various nodes is another factor that needs to be addressed. It is imperative that organizations identify the data considered to be private to ensure appropriate countermeasures and mechanisms are in full effect to protect it. For example, such data could be appropriated for marketing services, and organizations could potentially be found liable for wrongful business practices for using it without consumer knowledge or consent. In terms of financial data, another important facet, aside from encryption, is the matter of data integrity. Organizations must take data integrity take into account as part of Apple Pay security.

In the case of proximity payments, risks to POS systems also must be addressed. Organizations should ensure that third parties with which they interact have robust security governance in place [8]. Immediate attention should be focused on TSM as this performs a compatibility check on the vendor suppled mobile device for Apple Pay.

VI. Recommendations

Many experts say mobile payment methods offered by major providers are more secure than physical cards and traditional cash due to encryption and tokenization to mask payment card account numbers. Despite protections provided by technology advancements, Apple Pay remains vulnerable to hackers and identity theft. Cyber thieves can "spoof" consumers' mobile wallet via public Wi-Fi. Major mobile wallet providers use randomly generated payment tokens for validation of privacy. However, consumers still add cards to Apple Pay using unsecured public Wi-Fi networks, inviting hackers to lurk on them to spoof registration systems.

Tim Cook, Chief Executive Officer (CEO) of Apple, has called for stronger privacy regulations for tech companies and merchants due to recent data scandals. Cook states, "I am worried about the number of people around the world who easily handed over their information without fully understanding the affects. [10]" Consumers should load credit cards onto Apple Pay using their own password-protected Wi-Fi network or simply invest in a personal VPN.

A. Using Cryptogram Only Once

Using a cryptogram introduces EMV cryptographic strength to remote payments, not only for in-app payments

on the mobile device, but also for interconnected Apple devices used for Digital Secure Remote Payment (DSRP) [53]. Apple continues to recommend to consumers and merchants that the cryptogram token should be used only once; yet both parties frequently use this token multiple times. Using a cryptogram multiple times creates a vulnerability whereby hackers can manipulate delivery details to authorize fraudulent payments as the information is sent in clear text without integrity checking.

B. Precautionary Measures

Based on the findings in this study, the following measures should be undertaken to better security mobile payment services:

- Be wary about "https://" on websites. Fraudulent websites may also obtain "https://".
- Avoid Public Wi-Fi
- If public Wi-Fi use is unavoidable, then do not share any credentials (e.g., user-id, password)
- Never perform any financial transactions on public Wi-Fi

C. Be Aware of Masque Attacks

Although Apple has [partially] fixed the original Masque Attack on version 8.1.3, there still are other iOS attack surfaces and vulnerabilities to exploit during the installation process [34]. In addition, one in every three iOS devices is likely to be vulnerable to all Masque Attacks due to consumers neglecting to upgrade their devices. Users must keep their devices up-to-date with the latest software releases to ensure transactions are secured and no unforeseen interruptions take place during daily use.

VII. CONCLUSION

The iPhone has become a ubiquitous device for communication, entertainment, computation, and now contactless payment methods [5]. Apple Pay is undergoing transformations that hint at a seductive and promising future, where consumers and sellers alike will enjoy even more convenience and time savings. But areas of uncertainty remain. Key way-ahead considerations regarding the security and assurance of Apple Pay include the following:

- Key drivers from the consumer perspective
- Hardware secure element in the mobile device
- Trusted Execution Environment (Secure Enclave iOS)
- Device-specific Personal Area Network (PAN) with unique cryptogram (keys)
- NFC connectivity: EMV versions need to be consistently and continuously updated
- Taking advantage of EMV specifications
- Tokenization implementation
- Verification through biometrics (e.g., Touch ID, facial recognition)

ACKNOWLEDGEMENTS

"This work was supported [in part] by the Commonwealth Cyber Initiative, an investment in the advancement of cyber R&D, innovation and workforce development. For more information about CCI, visit cyberinitiative.org."

REFERENCES

[1] N. Mallat, M. Rossi and V. K. Tuunainen, "Mobile Banking Services," *Communications of the ACM*, vol. 47, no. 5, pp. 42-46, 2004.

[2] C. Xinru, "Information Security of Apple Pay," 2016. https://www.theseus.fi/bitstream/handle/10024/118948/Chen_Xinru.pdf?sequence=1&isAllowed=y.

[3] "Mobile Banking Security," VASCO, https://www.vasco.com/solutions/banking-cyber-security/mobile-banking-security.html.

[4] "Apple Pay Security and Privacy Overview," Apple, https://support.apple.com/en-us/HT203027.

[5] X. Lu, Y. Zhu, D. Li, B. Xu, W. Chen and Z. Ding, "Minimum cost collaborative sensing network with mobile phones," IEEE Explore, 13 June 2014. https://ieeexplore.ieee.org/document/6654784/.

[6] "This One Group is not Interested in Mobile Wallets," Business Insider, 15 August 2016. http://www.businessinsider.com/this-one-group-is-not-interested-in-mobile-wallets-2016-8.

[7] "Millennials Could Drive Mobile Wallet Adoption," Business Insider, 23 December 2016. http://www.businessinsider.com/millennials-could-drive-mobile-wallet-adoption-2016-12?r=UK&IR=T.

[8] "Mobile Payments: Risk, Security and Assurance Issues," November 2011. https://www.isaca.org/Groups/Professional- English/pci-compliance/GroupDocuments/MobilePaymentsWP.pdf.

[9] "Mobile Network Operator," Techopedia, https://www.techopedia.com/definition/27804/mobile-network-operator-mno.

[10] "Apple CEO Wants Stronger Privacy Laws," PYMNTS, 26 March 2018. https://www.pymnts.com/apple/2018/apple-ceo-privacy- laws-regulations-facebook/.

[11] J. Leyden , "Wallet-Snatch Hack: Apple Pay 'Vulnerable to Attack', Claim Researchers," The Register, 28 July 2017. https://www.theregister.co.uk/2017/07/28/applepay_vuln/.

[12] A. S. Jawale and J. S. Park, "A Security Analysis on Apple Pay," 6 March 2017. https://ieeexplore.ieee.org/document/7870214/?anchor=references.

[13] M. de Reuver and J. Ondrus, "When Techological Superiority is Not Enough: The Struggle to Impose the SIM Card as the NFC Secure Element for Mobile Payment Platforms," *Telecommunications Policy*, vol. 41, no. 4, pp. 253-262, 2017.

[14] T. Fox-Brewster, "Here's Proof Apple Pay is Useful for Stealing People's Money," Forbes, 1 March 2016. https://www.forbes.com/sites/thomasbrewster/2016/03/01/apple-pay-fraud-test/#330b50a46c6d.

[15] "Apple Pay's Low-Tech Security Problem," PYMNTS, 4 March 2016. https://www.pymnts.com/apple-pay-tracker/2016/apple-pays-low-tech-security-problem/.

[16] "Unauthorized Modification of iOS Can Cause Security Vulnerabilities, Instability, Shortened Battery Life, and other Issues," Apple, https://support.apple.com/en-us/HT201954.

[17] J. Sowells, "Apple Pay is Vulnerable to Malware," Hacker Combat, 31 July 2017. https://hackercombat.com/apple-pay-vulnerable-malware-attacks/.

[18] "What is SSL, TLS and HTTPS?," Symantec Website Security, https://www.websecurity.symantec.com/security-topics/what-is-ssl-tls-https.

[19] W. Chou, "Inside SSL: Accelerating Secure Transactions," *IT Professional*, vol. 4, no. 5, pp. 37-41, 2002.

[20] "What is an SSL Certificate and How Does it Work?," Digicert, https://www.digicert.com/ssl/.

[21] "New Malware Attack Techniques Expose Security Flaws in Apple Pay," TEISS, 27 July 2017. https://teiss.co.uk/news/apple-pay-malware-attack-techniques/.

[22] "POODLE CVE-2011-3389 Detail," National Vulnerability Database, 6 September 2011. https://nvd.nist.gov/vuln/detail/CVE-2011-3389#vulnCurrentDescriptionTitle.

[23] A. Prodromou, "TLS/SSL Explained- Examples of a TLS Vulnerability and Attack," Acunetix, 22 March 2017. https://www.acunetix.com/blog/articles/tls-vulnerabilities-attacks-final-part/.

[24] "Vulnerability and Exploit Database," Rapid7, https://www.rapid7.com/db/modules/auxiliary/scanner/http/ssl_version.

[25] "SSL V3.0 "Poodle" Vulnerability - CVE-2014-3566," Oracle https://www.oracle.com/technetwork/topics/security/poodlecve-2014-3566-2339408.html.

[26] "BEAST (CVE-2011-3389)," National Vulnerability Database, https://nvd.nist.gov/vuln/detail/CVE-2011-3389.

[27] L. Constantin, "Oracle Patches Java Flaw Exploited in SSL Beast Attack," Computer World, 19 October 2011. https://www.computerworld.com/article/2499203/enterprise-applications/oracle-patches-java-flaw-exploited-in-ssl-beast-attack.html.

[28] "CRIME CVE-2012-4929 Detail," National Vulnerability Database, https://nvd.nist.gov/vuln/detail/CVE-2012-4929.

[29] "SSL/TLS BREACH Vulnerability CVE-2013-3587," F5 Networks, https://support.f5.com/csp/article/K14634.

[30] "OpenSSL 'Heartbleed' Vulnerability (CVE-2014-0160)," United States Computer Emergency Readiness Team, https://www.us-cert.gov/ncas/alerts/TA14-098A.

[31] "Masque Attack: All Your iOS Apps Belong to Us," Fire Eye, 10 November 2014. https://www.fireeye.com/blog/threat-research/2014/11/masque-attack-all-your-ios-apps-belong-to-us.html.

[32] "The Heartbleed Bug," Heartbleed, http://heartbleed.com/.

[33] M. Gokey, "Masque Attack News: Researcher Find New Risk in iOS 'Masque Attack' Bug," Digital Trends, 6 August 2015. https://www.digitaltrends.com/mobile/ios-bug-masque-attack-news/.

[34] Z. Chen, T. Wei, H. Zue and Y. Zhang, "Three New Masque Attacks against iOS: Demolishing, Breaking and Hijacking," Fire Eye, 30 June 2015. https://www.fireeye.com/blog/threat-research/2015/06/three_new_masqueatt.html.

[35] D. Gilbert, "Masque Attacks are Back: IPhones and IPads Vulnerable to Sensitive App Data Theft," IB Times, 1 July 2015. https://www.ibtimes.co.uk/masque-attacks-are-back-iphones-ipads-vulnerable-sensitive-app-data-theft-1508816.

[36] L. Constantin, "One Third of Enterprise iOS Devices Vulnerable to App, Data Hijacking Attacks," CIO from IDG, 1 July 2015. https://www.cio.com/article/2943154/one-third-of-enterprise-ios-devices-vulnerable-to-app-data-hijacking-attacks.html.

[37] W. Alosaimi, M. Alshamrani and K. Al-Begain, "Simulation-Based Study of Distributed Denial of Service Attacks Prevention in the Cloud," 11 September 2015. https://ieeexplore.ieee.org/document/7373219/.

[38] P. Venkataram, B. S. Babu, M. K. Naveen and G. S. Gungal, "A Method of Fraud & Intrusion Detection for E-payment Systems in Mobile e-Commerce," 15 May 2017. https://ieeexplore.ieee.org/document/4197955/.

[39] "Types of DDoS Attacks," Verisign, https://www.verisign.com/en_US/security-services/ddos-protection/types-of-ddos-attacks/index.xhtml.

[40] D. Meyer, "Cloudflare and Apple Deal Fresh Blows to Neo-Nazi Sites," Fortune, 17 August 2017. http://fortune.com/2017/08/17/cloudflare-apple-neo-nazi-charlottesville/.

[41] G. V. Hulme, "DDoS Explained: How Distributed Denial of Service Attacks Are Evolving," CSO, 12 March 2018. https://www.csoonline.com/article/3222095/network-security/ddos-explained-how-denial-of-service-attacks-are-evolving.html.

[42] S. M. Kerner, "Memcached DDoS Attacks Slow Down as Patching Ramps Up," eWeek, 9 March 2018. http://www.eweek.com/security/memcached-ddos-attacks-slow-down-as-patching-ramps-up.

[43] P. L. Chatain, R. Hernandez-Coss, K. Borowik and A. Zerzan, "Integrity in Mobile Phone Financial Services- Measures for mitigating risks from money laundering and terrorist financing," May 2008. http://siteresources.worldbank.org/INTAML/Resources/WP146_Web.pdf.

[44] B. Sullivan, "How Tokenization May Change the Way You Pay," CNBC, 14 Dec 2014. https://www.cnbc.com/2014/12/12/how-tokenization-may-change-the-way-you-pay.html.

[45] E. Kovacs, "Tokenization: Benefits and Challenges for Securing Transaction Data," Security Week, 4 November 2014. https://www.securityweek.com/tokenization-benefits-and-challenges-securing-transaction-data.

[46] D. Etherington, "Tech Crunch," 16 January 2014. https://techcrunch.com/2014/01/16/apple-patents-mobile-payments-method-with-secure-element-for-protecting-account-info/.

[47] "Storing Keys in the Secure Enclave," https://developer.apple.com/documentation/security/certificate_key_and_trust_services/keys/storing_keys_in_the_secure_enclave.

[48] M. Roland, "Practical Attack Scenarios on Secure Element-Enabled Mobile Devices," Near Field Communication Research Lab Hagenberg, 13 March 2012. https://pdfs.semanticscholar.org/6826/325bb2721fafcd04fc28b1fa6220faade48c.pdf.

[49] F. D. Evans, "Digital Payments Security Discussion Secure Element (SE) vs Host Card Emulation (HCE)," Booz | Allen | Hamilton, 14 October 2014. https://www.itu.int/en/ITU- D/Regional-Presence/CIS/Documents/Events/2014/10_Baku/Session_5_Evans.pdf.

[50] "Encryption at the Point of Interaction," Velocity, https://nabvelocity.com/articles/encryption/.

[51] J. Heggestuen, "Payments Security is Undergoing a Revolution and Apple Pay is Leading the Way," Business Insider, 21 April 2015. http://www.businessinsider.com/apple-pay-leads-new-security-protocols-2015-4.

[52] "What Data Thieves Don't Want you to Know: The Facts About Encryption and Tokenization," https://www.firstdata.com/downloads/thought-leadership/TokenizationEncryptionWP.pdf.

[53] "Digital Secure Remote Payment: How Apple Pay Can Change the Future of Remote Payments," Computer Weekly, 27 March 2018. https://www.computerweekly.com/ehandbook/Digital-Secure-Remote-Payment-How-Apple-Pay-can-change-the-future-of-remote-payments.