

**IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION**

RFCyber CORP.,

Plaintiff,

v.

APPLE, INC.

Defendants.

§
§
§
§
§
§
§
§
§
§
§

Case No. 6:21-cv-00916-ADA

JURY TRIAL DEMANDED

AMENDED COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff, RFCyber Corp. (“RFCyber” or “Plaintiff”), files this Amended Complaint against Defendant Apple, Inc. (“Apple” or “Defendant”), for patent infringement under 35 U.S.C. § 271 and alleges as follows:

THE PARTIES

1. RFCyber is a Texas corporation with a place of business at 600 Columbus Avenue, Suite 106, Waco, Texas 76701. RFCyber is the owner of all right, title, and interest in and to, or is the exclusive licensee with the right to sue for U.S. Patent Nos. 8,118,218, 8,448,855, 9,189,787, 9,240,009, 10,600,046, and 11,018,724 (collectively, the “Patents-in-Suit” or “Asserted Patents”).

2. Defendant Apple, Inc. is a corporation organized and existing under the laws of California, with one or more regular and established places of business in this District at least at 12545 Riata Vista Circle, Austin, Texas 78727; 12801 Delcour Drive, Austin, Texas 78727; 6800 W Parmer Lane, Austin, Texas 78729, and 3121 Palm Way, Austin, Texas 78758. Apple may be served with process through its registered agent, the CT Corp System, at 1999 Bryan St., Ste. 900 Dallas, Texas 75201-3136. In November 2019, Apple stated that it had approximately

7,000 employees in Austin and that it expected to open, in 2022, a \$1 billion, 3 million-square-foot campus with capacity for 15,000 employees. *See*

<https://www.apple.com/newsroom/2019/11/apple-expands-in-austin/>. Apple is registered to do business in the State of Texas and has been since at least May 16, 1980.

JURISDICTION AND VENUE

3. This is an action for patent infringement arising under the patent laws of the United States, 35 U.S.C. §§ 1, et seq. This Court has subject matter jurisdiction over this action pursuant to 28 U.S.C. §§ 1331, 1332, 1338, and 1367.

4. The amount in controversy exceeds \$75,000.

5. This Court has specific and personal jurisdiction over Defendant consistent with the requirements of the Due Process Clause of the United States Constitution and the Texas Long Arm Statute. Upon information and belief, Defendant has sufficient minimum contacts with the forum because Defendant has physical locations and transacts substantial business in the State of Texas and in this Judicial District. Further, Defendant has, directly or through subsidiaries or intermediaries, committed and continues to commit acts of patent infringement in the State of Texas and in this Judicial District as alleged in this Complaint, as alleged more particularly below.

6. Venue is proper in this Judicial District pursuant to 28 U.S.C. §§ 1400(b) and 1391(b) and (c) because Defendant is subject to personal jurisdiction in this Judicial District, has committed acts of patent infringement in this Judicial District, and has a regular and established place of business in this Judicial District. Defendant, through its own acts, makes, uses, sells, and/or offers to sell infringing products within this Judicial District, regularly does and solicits

business in this Judicial District, and has the requisite minimum contacts with the Judicial District such that this venue is a fair and reasonable one.

PATENTS-IN-SUIT

7. On February 21, 2012, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 8,118,218 (the “’218 Patent”) entitled “Method and Apparatus for Providing Electronic Purse.” A true and correct copy of the ’218 Patent is attached as Exhibit A.

8. On May 28, 2013, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 8,448,855 (the “’855 Patent”) entitled “Method and Apparatus For Funding An Electronic Purse.” A true and correct copy of the ’855 Patent is attached as Exhibit B.

9. On November 17, 2015, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 9,189,787 (the “’787 Patent”) entitled “Method and Apparatus for Conducting E-Commerce and M-Commerce.” A true and correct copy of the ’787 Patent is attached as Exhibit C.

10. On January 19, 2016, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 9,240,009 (the “’009 Patent”) entitled “Mobile Devices for Commerce Over Unsecured Networks.” A true and correct copy of the ’009 Patent is attached as Exhibit D.

11. On January March 24, 2020, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 10,600,046 (the “’046 Patent”) entitled “Method and Apparatus for Mobile Payments.” A true and correct copy of the ’046 Patent is attached as Exhibit E.

12. On May 25, 2021, the United States Patent and Trademark Office duly and legally issued U.S. Patent No. 11,018,724 (the “’724 Patent”) entitled “Method and Apparatus for Emulating Multiple Cards in Mobile Devices.” A true and correct copy of the ’724 Patent is attached as Exhibit F.

13. RFCyber is the sole and exclusive owner of all right, title and interest to and in, or is the exclusive licensee with the right to sue for, the ’218, ’855, ’787, ’009, ’046, and ’724 Patents, and holds the exclusive right to take all actions necessary to enforce its rights to the Patents-in-Suit, including the filing of this patent infringement lawsuit. RFCyber also has the right to recover all damages for past, present, and future infringement of the Patents-in-Suit and to seek injunctive relief as appropriate under the law.

INFRINGEMENT ALLEGATIONS

14. The technologies of the Patents-in-Suit were variously invented by Liang Seng Koh, Hsin Pan, Xiangzhen Xie, Futong Cho, and Fuliang Cho. The Patents-in-Suit generally cover apparatus and methods for enabling secure contactless payment with a portable device. In one exemplary embodiment, a smart card module including a secure element may emulate a payment card over near field communications (“NFC”). For example, users may select one of a plurality of payment cards stored in a memory of the secure element, and carry out a transaction via NFC at a point of service (“POS”). In another embodiment, the device may securely conduct transactions over an open network with a payment server. By facilitating the settlement of charges using an NFC mobile device to read off data pertaining to an electronic invoice, the inventions of the Patents-in-Suit provide significant time-savings, particularly in situations where a payment process would otherwise involve more than one contact between a merchant and consumer.

15. Apple has manufactured, used, marketed, distributed, sold, offered for sale, and exported from and imported into the United States devices and software that infringe the Patents-in-Suit. Apple has distributed variants of Apple Pay that have included functionality to emulate a payment card and settle a transaction via NFC and/or MST at least since October 2014.¹ Apple Pay is operable on a range of Apple devices, including at least all devices from the iPhone 6, iPhone 6 Plus, and above, including, at least all variants of the following Apple devices: iPhone 6, iPhone 6 Plus, iPhone SE, iPhone 7, iPhone 7Plus, iPhone 8, iPhone 8 Plus, iPhone X, iPhone 11, iPhone 12, iPad Pro, iPad Air, iPad, and iPad mini models with Touch ID or Face ID, Apple Watch Series 1 and later, Mac models with Touch ID, Mac computers with Apple Silicon that are paired with a Magic Keyboard with Touch ID, and all Apple devices released since October 2014.² The current and previous versions of Apple Pay and devices running Apple Pay, alone and together, are non-limiting instances of the Accused Products. The Accused Products include, for example, the representative iPhone X running Apple Pay. The Accused Products practice the claims of the Patents-in-Suit to improve the shopping experience of their users, and to improve Apple's position in the market.

16. In August of 2016, LogicPatents, a broker, contacted Apple regarding the RFCyber patent portfolio, specifically the '218, '855, '787, and '009 Patents, as well as the applications that eventually issued as the '046 and '724 Patents.

17. No agreement was reached with Apple regarding any of the Patents-in-Suit.

18. At least as a result of LogicPatents contacting Apple, Apple has had knowledge of the '218, '855, '787, and '009 Patents since at least August 2016.

¹ See <https://www.apple.com/newsroom/2014/09/09Apple-Announces-Apple-Pay/>

² <https://support.apple.com/en-us/HT208531>

19. On information and belief, as a result of LogicPatents contacting Apple, Apple has had knowledge of the '046 and '724 Patents since they issued on March 24, 2020 and May 25, 2021, respectively.

20. Apple is the assignee of U.S. Patent No. 10,878,414 ("the '414 Patent"). On September 23, 2016, during prosecution of the '414 Patent, Apple listed the publication of the '009 Patent's application in an information disclosure statement to the United States Patent Office. Apple therefore had knowledge of the '009 Patent since at least September 23, 2016. On information and belief, Apple's investigation that discovered the '009 Patent provided it with knowledge of the other Patents-in-Suit at least as of the same time.

21. Apple is the assignee of U.S. Patent No. 10,929,843 ("the '843 Patent"). During prosecution, the Patent Examiner cited 15 RFCyber patents and applications, including the '009 and '046 Patents. Apple therefore had knowledge of at least the '009 and '046 Patents at least as of April 29, 2020, when the Examiner cited them during prosecution of the '843 Patent.

22. On information and belief, as a sophisticated company with an experienced legal department, Apple investigated RFCyber's issued patents after seeing numerous RFCyber patents and applications cited during prosecution of the '843 Patent. On information and belief, Apple further had knowledge of the Patents-in-Suit at least since November 29, 2017, when the Examiner cited six RFCyber patent applications during prosecution of the '843 Patent.

23. Apple has also had knowledge of the Patents-in-Suit, and the way that its products infringe those patents, since it received the original Complaint in this case, filed on September 7, 2021 and served on September 13, 2021.

24. Apple's infringement of the Patents-in-Suit is willful. Apple continues to commit acts of infringement despite a high likelihood that its actions constitute infringement, and Apple

knew or should have known that its actions constituted an unjustifiably high risk of infringement. Apple's infringement of patents that were specifically disclosed to it, and that it declined to reach an agreement on, and its continuing infringement after the filing of the original Complaint, is particularly egregious.

25. RFCyber has at all times complied with the marking provisions of 35 U.S.C. § 287 with respect to the Patents-in-Suit. On information and belief, any prior assignees and licensees have also complied with the marking provisions of 35 U.S.C. § 287.

COUNT I
(Infringement of the '218 Patent)

26. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

27. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '218 Patent.

28. Apple infringes, contributes to the infringement of, and/or induces infringement of the '218 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '218 Patent, including, but not limited to, at least the Accused Products.

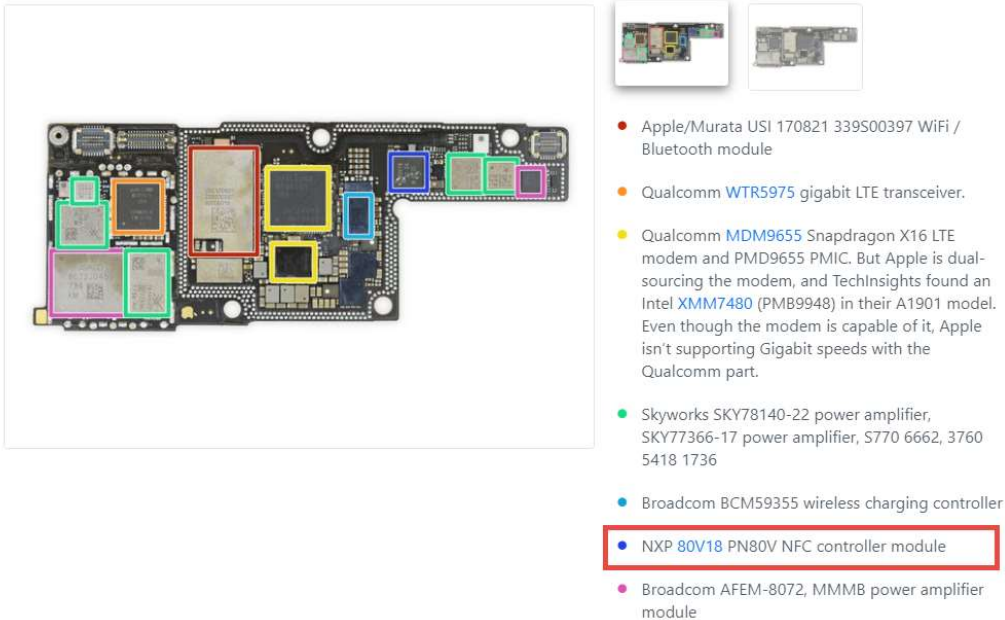
29. Apple has directly infringed and continues to directly infringe the '218 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '218 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '218 Patent, including, for example, card emulation and NFC

payment functionality implemented by Apple Pay running on an Apple device, such as the representative iPhone X. For example, these products infringe at least claim 1 of the '218 Patent.

30. For example, Apple has and continues to directly infringe at least claim 1 of the '218 Patent by making, using, offering to sell, selling and/or importing into the United States products that implement a method for providing an e-purse, the method comprising: providing a portable device including or communicating with a smart card pre-loaded with an emulator configured to execute a request from an e-purse applet and provide a response the e-purse applet is configured to expect, the portable device including a memory space loaded with a midlet that is configured to facilitate communication between the e-purse applet and a payment server over a wireless network, wherein the e-purse applet is downloaded and installed in the smart card when the smart card is in communication with the payment server, the portable device further includes a contactless interface that facilitates communication between the e-purse applet in the smart card and the payment server over a wired network; personalizing the e-purse applet by reading off data from the smart card to generate in the smart card one or more operation keys that are subsequently used to establish a secured channel between the e-purse applet and an e-purse security authentication module (SAM) external to the smart card, wherein said personalizing the e-purse applet comprises: establishing an initial security channel between the smart card and the e-purse SAM to install and personalize the e-purse applet in the smart card, and creating a security channel on top of the initial security channel to protect subsequent operations of the smart card with the e-purse SAM, wherein any subsequent operation of the emulator is conducted over the security channel via the e-purse applet.

31. The Accused Products provide a portable device, such as the Apple iPhone X, including or communicating with a smart card pre-loaded with an emulator configured to execute

a request from an e-purse applet and provide a response the e-purse applet is configured to expect. For example, the iPhone X includes or communicates with a smart card such as an NFC module, and/or assembly of an NFC module, secure element, processor, microcontroller, and/or memory, such as an NXP 80V18 PN80V NFC Controller. On information and belief, the smart card (e.g. NFC module) of the iPhone X is pre-loaded with an emulator configured to execute a request from an e-purse applet, such as a payment card applet within Apple Pay, and provide a response that the applet is configured to expect.



32. For example, Accused Products, such as the iPhone X, include a memory space loaded with a midlet, such as Apple Wallet or other software, that is configured to facilitate communication between the e-purse applet, such as a payment card stored on the product, and a payment server, such as a merchant and/or financial institution payment server, over a wireless network. For example, on information and belief, the Apple iPhone X comprises memory such as RAM, ROM, Flash, and/or EEPROM, including in both the NFC module and secure element.

For example, on information and belief, the secure element of the Apple iPhone X running Apple Pay further comprises a memory such as RAM, ROM, Flash, and/or EEPROM.

Apple Wallet

Apple Wallet is used to add and manage credit, debit, and store cards and to make payments with Apple Pay. Users can view their cards and may be able to view additional information provided by their card issuer, such as their card issuer's privacy policy, recent transactions, and more in Apple Wallet. Users can also add cards to Apple Pay in:

- Setup Assistant and Settings for iOS and iPadOS
- The Watch app for Apple Watch
- Wallet & Apple Pay in System Preferences for Mac computers with Touch ID

In addition, Apple Wallet allows users to add and manage transit cards, rewards cards, boarding passes, tickets, gift cards, student ID cards, and more.

<https://support.apple.com/guide/security/apple-pay-component-security-sec2561eb018/1/web/1>

How Apple Pay uses the Secure Element and NFC controller

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-sec2561eb018/1/web/1>

33. The Accused Products further perform a method wherein the e-purse applet is downloaded and installed in the smart card when the smart card is in communication with the payment server. For example, the Apple iPhone X running Apple Pay operates to download and

install a payment card applet when the NFC module is in communication with the payment institution's server:

How Apple Pay uses the Secure Element and NFC controller

Secure Element

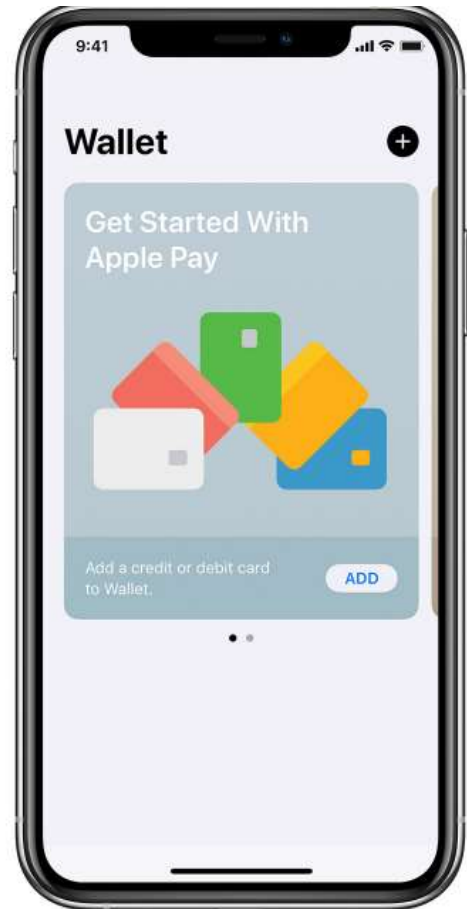
The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-seccb53a35f0/web>

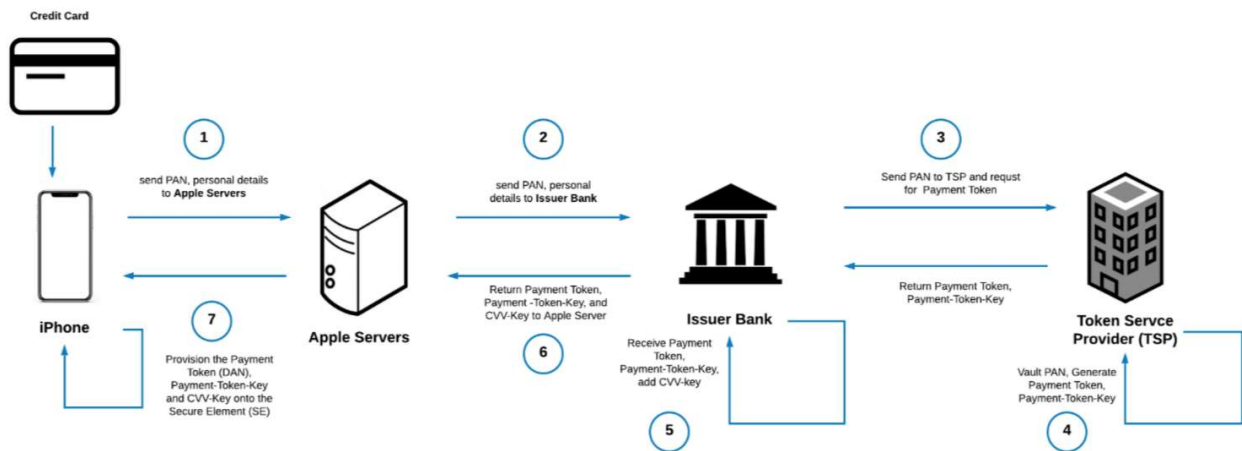
Add a card on your iPhone

1. Go to Wallet and tap **+**.
2. Follow the steps to add a new card. If you're asked to add **the card that you use with your Apple ID**, cards on other devices, or cards that you've recently removed, choose them, then enter the card security codes. You might be required to download an app from your bank or card issuer to add a card to Wallet. In China mainland, you might be required to create or update a 6-digit passcode.
3. Tap Next. Your bank or card issuer will verify your information and decide if you can use your card with Apple Pay. If your bank or issuer needs more information to verify your card, they'll ask you for it. When you have the information, go back to Wallet and tap your card.
4. After your bank or issuer verifies your card, tap Next. **Then start using Apple Pay.**

Get help [adding your card to Wallet](#).



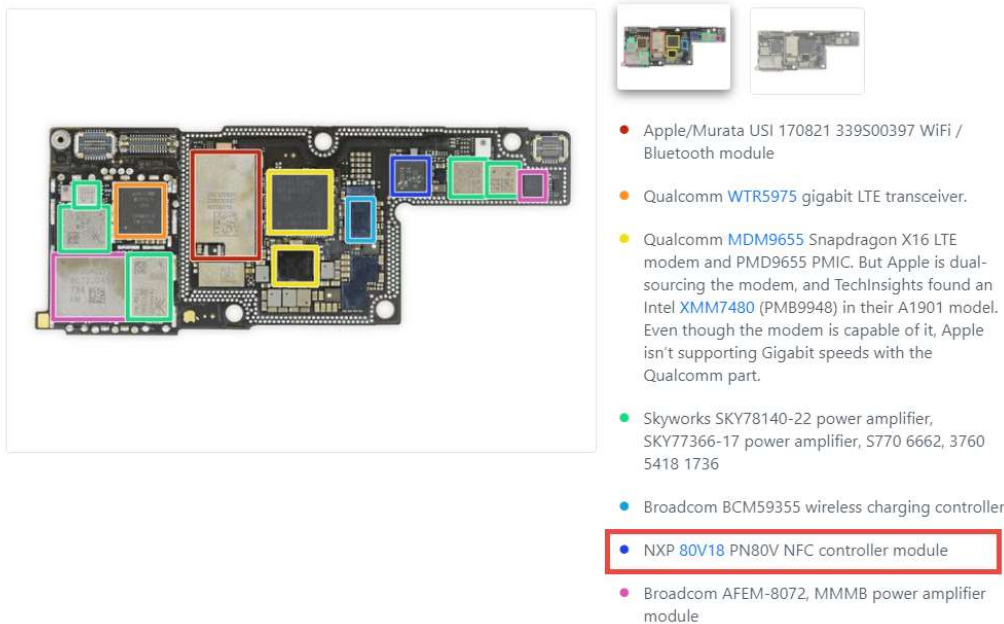
<https://support.apple.com/en-us/HT204506>



<https://codeburst.io/how-does-apple-pay-actually-work-f52f7d9348b7>

34. The Accused Products further include a contactless interface that facilitates communication between the e-purse applet in the smart card and the payment server over a wired

network. For example, on information and belief, the NFC module of the Apple iPhone X includes a contactless NFC interface that facilitates communication between a payment card applet and a payment server over a wired network, such as via a payment card reader at a POS connected to a payment server via wired network³:



<https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

35. The Accused Products further personalize the e-purse applet (e.g. payment card applet within Apple Pay) by reading off data from the smart card (e.g. NFC Module or secure element) to generate in the smart card one or more operation keys that are subsequently used to establish a secured channel between the e-purse applet and an e-purse security authentication module (SAM) external to the smart card. For example, on information and belief, Apple Pay establishes operations keys that operate to establish secure connections between a stored payment card and an authentication module at a server of the card issuer and/or merchant when adding a given card to the device for the first time, and/or subsequently during transactions:

³ https://developer.apple.com/documentation/apple_pay_on_the_web/setting_up_your_server

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

Apple doesn't store or have access to the original card numbers of credit, debit, or prepaid cards that you add to Apple Pay. Apple Pay stores only a portion of your actual card numbers and a portion of your Device Account Numbers, along with a card description. Your cards are associated with your Apple ID to help you add and manage your cards across your devices.

<https://support.apple.com/en-us/HT203027>

36. The Accused Products further practice a method wherein personalizing the e-purse applet (*e.g.* configuring the payment card applet within Apple Pay) comprises establishing an initial security channel between the smart card and the e-purse SAM to install and personalize the e-purse applet in the smart card. For example, on information and belief, Apple Pay operates to establish a security channel with at least an Apple server after a user enters details for a given payment card, and operates to install and personalize the applet in the smart card, such as to install the card with the user's personal information in the secure element:

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-secceb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

37. The Accused Products create a security channel on top of the initial security channel to protect subsequent operations of the smart card within the e-purse SAM, wherein any subsequent operation of the emulator is conducted over the security channel via the e-purse applet. For example, on information and belief, once a payment card applet is installed, operation of the emulator is conducted via operation of the e-purse applet using the security key installed during the personalization process.

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-seccb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

38. Apple has had knowledge and notice of the '218 Patent at least as of August 2016.
39. Apple has had knowledge of how its products infringe the '218 Patent since at least September 7, 2021. Despite that knowledge, Apple continues to infringe the '218 Patent both directly and indirectly.

40. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '218 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under the doctrine of equivalents, through their use of the inventions claimed in the '218 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly infringe the '218 Patent. Apple performs these affirmative acts with knowledge of the '218 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '218 Patent.

41. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '218 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '218 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '218 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are

known by Apple to be especially made or adapted for use in the infringement of the '218 Patent. Apple performs these affirmative acts with knowledge of the '218 Patent and with intent, or willful blindness, that they cause the direct infringement of the '218 Patent.

42. Because of Apple's direct and indirect infringement of the '218 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

43. Because of Apple's direct and indirect infringement of the '218 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

COUNT II
(Infringement of the '855 Patent)

44. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

45. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '855 Patent.

46. Apple infringes, contributes to the infringement of, and/or induces infringement of the '855 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '855 Patent, including, but not limited to, at least the Accused Products.

47. Apple has directly infringed and continues to directly infringe the '855 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '855 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '855 Patent, including, for example, card emulation and NFC

payment functionality implemented by Apple Pay running on an IOS device. For example, these products infringe at least claim 1 of the '855 Patent.

48. For example, Apple has and continues to directly infringe at least claim 1 of the '855 Patent by making, using, offering to sell, selling and/or importing into the United States products that practice a method for funding an e-purse, the method comprising receiving a PIN from a user of a portable device, wherein the portable device is a near field communication (NFC) enabled device that includes a card module; initiating a request from a midlet embedded in the portable device after the PIN is verified, wherein the midlet sends the request to an e-purse applet; causing the e-purse applet to compose a response to the request; sending the response by the e-purse applet over a wireless network to a server administrating the e-purse, the server configured to verify the response against an account in a financial institution across a network, a fund transfer request is initiated by the server to the financial institution when the response is successfully verified; receiving commands from the server in responding to the fund transfer request; and causing an emulator in the portable device to update a transaction log after an authenticity of the commands is verified by the e-purse applet wherein the e-purse in the portable device has been personalized by operations including: establishing an initial security channel between the card module and an e-purse security authentication module (SAM) external to the card module to install and personalize the e-purse applet in the card module, and creating a security channel on top of the initial security channel to protect subsequent operations of the card module with the e-purse SAM, wherein any subsequent transactions with the e-purse are conducted over the security channel.

49. The Accused Products practice a method of receiving a PIN from a user of a portable device, wherein the portable device is a near field communication (NFC) enabled device

that includes a card module. For example, on information and belief, the Apple iPhone X includes a card module, such as a NXP 80V18 PN80V NFC Controller, and requires a PIN to unlock, and further requires a PIN to carry out a transaction via NFC.⁴

50. The Accused Products practice a method of initiating a request from a midlet embedded in the portable device after the PIN is verified, wherein the midlet sends the request to an e-purse applet. For example, on information and belief, the Apple iPhone X practices a method of initiating a request from Apple Pay after the PIN, such as a passcode, is verified, where Apple Pay sends a request to a payment card applet.

Apple Pay uses security features built-in to the hardware and software of your device to help protect your transactions. In addition, to use Apple Pay, you must have a passcode set on your device and, optionally, Face ID or Touch ID. You can use a simple passcode, or you can set a more complex passcode for even greater security.

<https://support.apple.com/en-us/HT203027>

When you use Apple Pay in stores

When you use Apple Pay in stores that accept contactless payments, Apple Pay uses Near Field Communication (NFC) technology between your device and the payment terminal. NFC is an industry-standard, contactless technology that's designed to work only across short distances. If your iPhone is on and detects an NFC field, it will present you with your default card. To send your payment information, you must authenticate using Face ID, Touch ID, or your passcode (except in Japan if you designate a Suica card for Express Transit). With Face ID or with Apple Watch, you must double-click the side button when the device is unlocked to activate your default card for payment.

<https://support.apple.com/en-us/HT203027>

51. The Accused Products practice a method of causing the e-purse applet to compose a response to the request. For example, on information and belief, the payment card applet composes a response including the transaction, user, and/or device information, such as one or more operations keys, device account numbers, tokenized card information, and/or cryptograms.

⁴ See <https://support.apple.com/en-us/HT203027>

After you authenticate your transaction, the Secure Element provides your Device Account Number and a transaction-specific dynamic security code to the store's point of sale terminal along with additional information needed to complete the transaction. Again, neither Apple nor your device sends your actual payment card number. Before they approve the payment, your bank, card issuer, or payment network can verify your payment information by checking the dynamic security code to make sure that it's unique and tied to your device.

<https://support.apple.com/en-us/HT203027>

52. The Accused Products practice a method of sending the response by the e-purse applet over a wireless network to a server administrating the e-purse, the server configured to verify the response against an account in a financial institution across a network, a fund transfer request is initiated by the server to the financial institution when the response is successfully verified. For example, on information and belief, the Apple iPhone X performs a method of sending the response by a payment card applet to a payment server and/or gateway server over a wireless network, such a cellular network, Wireless WAN, Wireless MAN, Wireless PAN, Wireless LAN, and/or a Global Area Network. On information and belief, the payment and/or gateway server is configured to respond to the request, such as a request for funds to complete a transaction, when the response is verified.

When you use Apple Pay in stores

When you [use Apple Pay in stores](#) that accept contactless payments, Apple Pay uses Near Field Communication (NFC) technology between your device and the payment terminal. NFC is an industry-standard, contactless technology that's designed to work only across short distances. If your iPhone is on and detects an NFC field, it will present you with your default card. To send your payment information, you must authenticate using Face ID, Touch ID, or your passcode (except in Japan if you designate a Suica card for Express Transit). With Face ID or with Apple Watch, you must double-click the side button when the device is unlocked to activate your default card for payment.

After you authenticate your transaction, the Secure Element provides your Device Account Number and a transaction-specific dynamic security code to the store's point of sale terminal along with additional information needed to complete the transaction. Again, neither Apple nor your device sends your actual payment card number. Before they approve the payment, your bank, card issuer, or payment network can verify your payment information by checking the dynamic security code to make sure that it's unique and tied to your device.

<https://support.apple.com/en-us/HT203027>.

53. The Accused Products practice receiving commands from the server in responding to the fund transfer request. For example, on information and belief, the Apple iPhone X receives commands in response to a fund transfer request, such as to communicate transaction information to a card reader.

54. The Accused Products further practice causing an emulator in the portable device to update a transaction log after an authenticity of the commands is verified by the e-purse applet wherein the e-purse in the portable device has been personalized by operations including. For example, on information and belief, an emulator within the Apple iPhone X updates an Apple Pay transaction log once commands have been authenticated by an installed and configured payment card applet, such as based on operating keys, device account number, tokenized card information, and/or cryptograms.

View the information for a card and change its settings

1. In Wallet, tap the card.

Note: The last transaction may appear, showing an authorized amount that may differ from the amount of the payment charged to your account. For example, a gas station may request an authorization of \$99, even though you pumped only \$25 worth of gasoline. To see the final charges, see the statement from your card issuer, which includes all Apple Pay transactions.

2. Tap , then do any of the following:

- Tap Transactions to view your recent history. To hide this information, turn off Transaction History. To view all your Apple Pay activity, see the statement from your card issuer.
- View the last four digits of the card number and Device Account Number—the number transmitted to the merchant.
- Change the billing address.
- Remove the card from Wallet.

<https://support.apple.com/guide/iphone/manage-cards-and-activity-iph7b666943a/ios>

55. The Accused Products further practice establishing an initial security channel between the card module and an e-purse security authentication module (SAM) external to the card module to install and personalize the e-purse applet in the card module. For example, on information and belief, the Apple iPhone X personalizes payment card applets by establishing an initial security channel with a security authentication module located on or behind the card-issuer's payment server, to install and configure the payment cards with the user's personal information.

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS; is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

56. The Accused Products further practice a method of creating a security channel on top of the initial security channel to protect subsequent operations of the card module with the e-purse SAM, wherein any subsequent transactions with the e-purse are conducted over the

security channel. For example, on information and belief, an instance of Apple Pay operating on the Apple iPhone X operates to establish operating keys, device account numbers, tokenized card information, and/or cryptograms with which subsequent communications (*e.g.* subsequent transactions with a personalized card applet) are protected.

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-secceb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

57. Apple has had knowledge and notice of the '855 Patent at least as of August 2016.

On information and belief, Apple, as a sophisticated company, analyzed the '855 Patent when it became aware of it and was thus aware of how its products infringe the '855 Patent since at least August 2016.

58. On information and belief, Apple has further had knowledge of how it infringes the '855 Patent at least since around September 7, 2021. As a sophisticated company with knowledge of the '855 Patent, Apple analyzed its infringement of the related Patents-in-Suit after receiving the original Complaint and determined how it infringed the '855 Patent as well.

59. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '855 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by

others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under the doctrine of equivalents, through their use of the inventions claimed in the '855 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly infringe the '855 Patent. Apple performs these affirmative acts with knowledge of the '855 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '855 Patent.

60. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '855 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '855 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '855 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are known by Apple to be especially made or adapted for use in the infringement of the '855 Patent.

Apple performs these affirmative acts with knowledge of the '855 Patent and with intent, or willful blindness, that they cause the direct infringement of the '855 Patent.

61. Because of Apple's direct and indirect infringement of the '855 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

62. Because of Apple's direct and indirect infringement of the '855 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

COUNT III
(Infringement of the '787 Patent)

63. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

64. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '787 Patent.

65. Apple infringes, contributes to the infringement of, and/or induces infringement of the '787 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '787 Patent, including, but not limited to, at least the Accused Products.

66. Apple has directly infringed and continues to directly infringe the '787 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '787 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '787 Patent, including, for example, card emulation and NFC

payment functionality implemented by Apple Pay running on an Apple device, such as the representative Apple iPhone X. For example, these products infringe at least claim 1 of the '787 Patent.

67. For example, Apple has and continues to directly infringe at least claim 1 of the '787 Patent by making, using, offering to sell, selling and/or importing into the United States products that comprise a portable device for commerce, the portable device comprising an emulator loaded in a smart card module for storing security values and updated transaction logs, and an e-purse applet to cause the portable device to function as an electronic purse (e-purse), wherein both of the emulator and e-purse applet are already personalized via a personalization process built on a first security channel so that the emulator is set to store a set of keys for subsequent data access authentication and the e-purse applet is configured to conduct a transaction with a network server over a second security channel; a first interface configured to perform field communication (NFC) with a reader to perform electronic commerce with the e-purse applet against a fund stored in the emulator; a second interface configured to perform mobile commerce with a payment server via an application against the fund stored in the emulator; and a purse manager midlet being executed in the portable device to act as an agent to facilitate communications between the e-purse applet and a payment server to conduct transactions therebetween.

68. The Accused Products comprise an emulator loaded in a smart card module for storing security values and updated transaction logs. For example, the Apple iPhone X comprises an NFC Module and secure element with an emulator for storing security values, such as device account number, operating keys and/or a tokenized card and cryptogram, and for updating transaction logs, such as via Apple Pay:



<https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

View the information for a card and change its settings

1. In Wallet, tap the card.

Note: The last transaction may appear, showing an authorized amount that may differ from the amount of the payment charged to your account. For example, a gas station may request an authorization of \$99, even though you pumped only \$25 worth of gasoline. To see the final charges, see the statement from your card issuer, which includes all Apple Pay transactions.

2. Tap , then do any of the following:

- Tap Transactions to view your recent history. To hide this information, turn off Transaction History. To view all your Apple Pay activity, see the statement from your card issuer.
- View the last four digits of the card number and Device Account Number—the number transmitted to the merchant.
- Change the billing address.
- Remove the card from Wallet.

<https://support.apple.com/guide/iphone/manage-cards-and-activity-iph7b666943a/ios>

69. The accused products further comprise an e-purse applet, such as a payment card applet within Apple Pay, to cause the portable device (*e.g.* the Apple iPhone X) to function as an electronic purse. For example, applets within Apple Pay cause IOS devices to carry out a transaction, such as via NFC:

When you use Apple Pay in stores

When you [use Apple Pay in stores](#) that accept contactless payments, Apple Pay uses Near Field Communication (NFC) technology between your device and the payment terminal. NFC is an industry-standard, contactless technology that's designed to work only across short distances. If your iPhone is on and detects an NFC field, it will present you with your default card. To send your payment information, you must authenticate using Face ID, Touch ID, or your passcode (except in Japan if you designate a Suica card for Express Transit). With Face ID or with Apple Watch, you must double-click the side button when the device is unlocked to activate your default card for payment.

After you authenticate your transaction, the Secure Element provides your Device Account Number and a transaction-specific dynamic security code to the store's point of sale terminal along with additional information needed to complete the transaction. Again, neither Apple nor your device sends your actual payment card number. Before they approve the payment, your bank, card issuer, or payment network can verify your payment information by checking the dynamic security code to make sure that it's unique and tied to your device.

<https://support.apple.com/en-us/HT203027>

70. The Accused Products further comprise a portable device wherein both of the emulator (*e.g.* emulator of the NFC module) and e-purse applet (*e.g.* payment card applet) are already personalized via a personalization process built on a first security channel so that the emulator is set to store a set of keys for subsequent data access authentication and the e-purse applet is configured to conduct a transaction with a network server over a second security channel. For example, on information and belief, the emulator and applet of a iPhone X running Apple Pay are personalized during installation so that the emulator stores a set of keys (*e.g.* device account number, operating keys and/or a tokenized card and cryptogram) for subsequent access and authentication during transactions.

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-secceb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

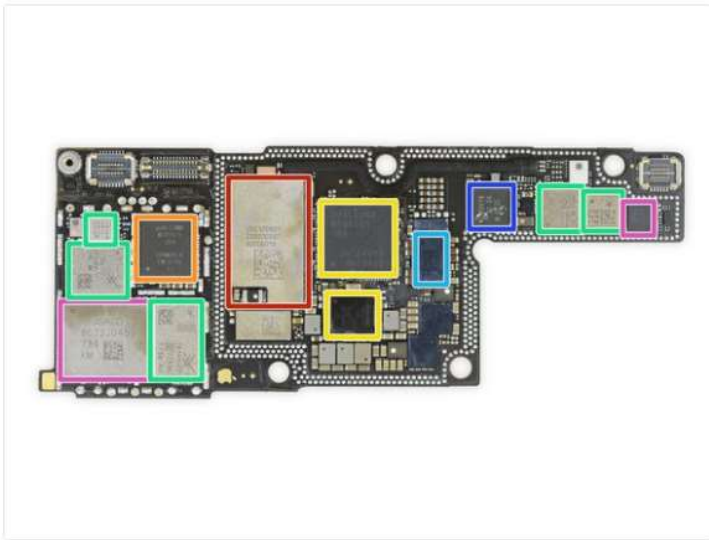
Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

71. The Accused Products further comprise a first interface configured to perform field communication (NFC) with a reader to perform electronic commerce with the e-purse applet against a fund stored in the emulator. For example, the Apple iPhone X comprises an NFC Module, such as an NXP 80V18 PN80V NFC Controller, including an NFC interface to perform electronic commerce with a card reader.



- Apple/Murata USI 170821 339S00397 WiFi / Bluetooth module
- Qualcomm WTR5975 gigabit LTE transceiver.
- Qualcomm MDM9655 Snapdragon X16 LTE modem and PMD9655 PMIC. But Apple is dual-sourcing the modem, and TechInsights found an Intel XMM7480 (PMB9948) in their A1901 model. Even though the modem is capable of it, Apple isn't supporting Gigabit speeds with the Qualcomm part.
- Skyworks SKY78140-22 power amplifier, SKY77366-17 power amplifier, S770 6662, 3760 5418 1736
- Broadcom BCM59355 wireless charging controller
- NXP 80V18 PN80V NFC controller module
- Broadcom AFEM-8072, MMMB power amplifier module

<https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

72. The Accused Products further comprise a second interface configured to perform mobile commerce with a payment server via an application against the fund stored in the emulator. For example, on information and belief, the Apple iPhone X comprises a second interface to perform mobile commerce with a payment server, such as the payment server of an issuer and/or a merchant, against a fund stored in the emulator, such as a gift card fund stored in the emulator of an NFC module via the payment servers of Apple Pay-enabled applications.

When you use Apple Pay within apps or on the web

When you use an app or a website that uses Apple Pay in iOS, watchOS, or macOS, the app or website can check if you have Apple Pay enabled on that device. You can manage this option in Settings > Safari on your iOS device, and in the Privacy tab in Safari preferences on your Mac.

To securely transmit your payment information when you pay in apps or on the web, Apple Pay receives your encrypted transaction and re-encrypts it with a developer-specific key before the transaction information is sent to the developer or payment processor. This key helps ensure that only the app or the website that you're purchasing from can access your encrypted payment information. Websites must verify their domain every time they offer Apple Pay as a payment option. Like with in-store payments, Apple sends your Device Account Number to the app or website along with the transaction-specific dynamic security code. Neither Apple nor your device sends your actual payment card number to the app.

Apple retains anonymous transaction information, including the approximate purchase amount, app developer and app name, approximate date and time, and whether the transaction completed successfully. Apple uses this data to improve Apple Pay and other products and services. Apple also requires apps and websites in Safari that use Apple Pay to have a privacy policy that you can view which governs their use of your data.

When you use Apple Pay on your iPhone or Apple Watch to confirm a purchase from your Mac in Safari, your Mac and the authorizing device communicate over an encrypted channel via Apple servers. Apple doesn't retain any of this information in a form that personally identifies you. You can disable the ability to use Apple Pay on your Mac in Settings on your iPhone. Go to Wallet & Apple Pay and turn off Allow Payments On Mac.

<https://support.apple.com/en-us/HT203027>

73. The Accused Products further comprise a purse manager midlet, such as Apple Wallet or other software, being executed in the portable device to act as an agent to facilitate communications between the e-purse applet and a payment server to conduct transactions therebetween. For example, on information and belief, the Apple iPhone X executes Apple Wallet to facilitate communications between payment cards (*e.g.* cards within an emulator and/or secure element of an NFC module) and a payment server (*e.g.* an issuer and/or merchant payment server) during transactions conducted via NFC and/or via Apple Pay-enabled application.

Apple Wallet

Apple Wallet is used to add and manage credit, debit, and store cards and to make payments with Apple Pay. Users can view their cards and may be able to view additional information provided by their card issuer, such as their card issuer's privacy policy, recent transactions, and more in Apple Wallet. Users can also add cards to Apple Pay in:

- Setup Assistant and Settings for iOS and iPadOS
- The Watch app for Apple Watch
- Wallet & Apple Pay in System Preferences for Mac computers with Touch ID

In addition, Apple Wallet allows users to add and manage transit cards, rewards cards, boarding passes, tickets, gift cards, student ID cards, and more.

<https://support.apple.com/guide/security/apple-pay-component-security-sec2561eb018/1/web/1>

74. Apple has had knowledge and notice of the '787 Patent at least as of August 2016.

75. Apple has had knowledge of how its products infringe the '787 Patent since at least September 7, 2021. Despite that knowledge, Apple continues to infringe the '787 Patent both directly and indirectly.

76. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '787 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under the doctrine of equivalents, through their use of the inventions claimed in the '787 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly

infringe the '787 Patent. Apple performs these affirmative acts with knowledge of the '787 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '787 Patent.

77. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '787 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '787 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '787 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are known by Apple to be especially made or adapted for use in the infringement of the '787 Patent. Apple performs these affirmative acts with knowledge of the '787 Patent and with intent, or willful blindness, that they cause the direct infringement of the '787 Patent.

78. Because of Apple's direct and indirect infringement of the '787 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

79. Because of Apple's direct and indirect infringement of the '787 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

COUNT IV
(Infringement of the '009 Patent)

80. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

81. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '009 Patent.

82. Apple infringes, contributes to the infringement of, and/or induces infringement of the '009 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '009 Patent, including, but not limited to, at least the Accused Products.

83. Apple has directly infringed and continues to directly infringe the '009 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '009 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '009 Patent, including, for example, card emulation and NFC payment functionality implemented by Apple Pay running on an IOS device, such as the representative Apple iPhone X. For example, these products infringe at least claim 1 of the '009 Patent.

84. For example, Apple has and continues to directly infringe at least claim 1 of the '009 Patent by making, using, offering to sell, selling and/or importing into the United States products that comprise a mobile device for conducting a secured transaction over a network, the mobile device comprising: a network interface; an interface to receive a secure element; a memory space for storing at least a module and an application downloaded from the network; a processor coupled to the memory space and configured to execute the module to perform operations including: sending to a server via the network interface an identifier identifying the application together with device information of a secure element, wherein the application is

downloaded from the network in the mobile device; establishing a secured channel between the secure element and the server using a key set installed on the secure element, wherein the server is configured to prepare data necessary for the application to function as designed on the mobile device; and receiving the data from the server to associate the application with the secure element, wherein the application subsequently functions in conjunction with the secure element.

85. The Accused Products comprise a network interface. For example, on information and belief, the Apple iPhone X comprises interfaces such as the NFC interface and antenna of an NXP 80V18 PN80V NFC Controller, a Qualcomm LTE Module, and/or Apple/Murata USI 170821 339S00397 WIFI/Bluetooth module.⁵

86. The Accused Products further comprise an interface to receive a secure element. For example, on information and belief, the Apple iPhone X comprises a secure element.

Apple Pay component security

Secure Element

The Secure Element is an industry-standard, certified chip running the Java Card platform, which is compliant with financial industry requirements for electronic payments. The Secure Element IC and the Java Card platform are certified in accordance with the EMVCo Security Evaluation process. After the successful completion of the security evaluation, EMVCo issues unique IC and platform certificates.

The Secure Element IC has been certified based on the Common Criteria standard.

<https://support.apple.com/guide/security/apple-pay-component-security-sec2561eb018/1/web/1>

87. The Accused Products further comprise a memory space for storing at least a module and an application downloaded from the network. For example, on information and

⁵ See <https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

belief, the Apple iPhone X includes memory such as RAM, ROM, Flash, and/or EEPROM for storing an application downloaded from the network, such as Apple Pay, applications configured to accept Apple Pay, and/or payment cards within Apple Pay.⁶

88. The Accused Products further comprise a processor coupled to the memory space and configured to execute the module to perform operations. For example, the Apple iPhone X comprises a processor such as an Apple A11 SoC and/or a Secure Enclave Processor, coupled to memory such as RAM, ROM, Flash, and/or EEPROM.⁷

89. The Accused Products further comprise a processor configured to execute the module to perform operations including, sending to a server via the network interface an identifier identifying the application together with device information of a secure element, wherein the application is downloaded from the network in the mobile device. For example, on information and belief, a processor of the Apple iPhone X are configured to execute sending an identifier, such as tokenized card information, device account number, operating keys, and/or one or more cryptograms associated with an instance of Apple Pay and/or a payment card within Apple Pay to an issuer and/or merchant payment server.

90. The Accused Products further comprise a processor configured to execute the module to perform operations including establishing a secured channel between the secure element and the server using a key set installed on the secure element, wherein the server is configured to prepare data necessary for the application to function as designed on the mobile device. For example, on information and belief, a processor of the Apple iPhone X is configured

⁶ <https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>;
<https://support.apple.com/guide/security/apple-pay-component-security-sec2561eb018/1/web/1>;
<https://support.apple.com/guide/security/secure-enclave-sec59b0b31ff/web> .

⁷ *Id.*

to establish a secure channel between a secure element (*e.g.* of the a secure element of its NFC Module) using a server key installed on the secure element, such as an operating key, device account number, token, and/or cryptogram associated with a payment card, and a payment server configured to prepare data sufficient to enable an NFC transaction.

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-secceb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

91. The Accused Products further comprise a processor configured to execute the module to perform operations including, receiving the data from the server to associate the application with the secure element, wherein the application subsequently functions in conjunction with the secure element. For example, on information and belief, a processor of the Apple iPhone X is configured to execute the module, such as Apple Pay, to perform operations including receiving data from a card-issuer payment server to associate the application, such as the payment card application, with the secure element, such as by generating a device-specific account number, device account number, token, cryptogram, and/or operating key associated with the payment card. For example, on information and belief, the payment card application

subsequently functions in conjunction with the secure element, such as during transactions performed via contactless payment at a point of sale.

92. Apple has had knowledge and notice of the '009 Patent at least as of August 2016.

93. Apple has had knowledge of how its products infringe the '009 Patent since at least September 7, 2021. Despite that knowledge, Apple continues to infringe the '009 Patent both directly and indirectly.

94. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '009 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under the doctrine of equivalents, through their use of the inventions claimed in the '009 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly infringe the '009 Patent. Apple performs these affirmative acts with knowledge of the '009 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '009 Patent.

95. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '009 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the

United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '009 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '009 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are known by Apple to be especially made or adapted for use in the infringement of the '009 Patent. Apple performs these affirmative acts with knowledge of the '009 Patent and with intent, or willful blindness, that they cause the direct infringement of the '009 Patent.

96. Because of Apple's direct and indirect infringement of the '009 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

97. Because of Apple's direct and indirect infringement of the '009 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

COUNT V
(Infringement of the '046 Patent)

98. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

99. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '046 Patent.

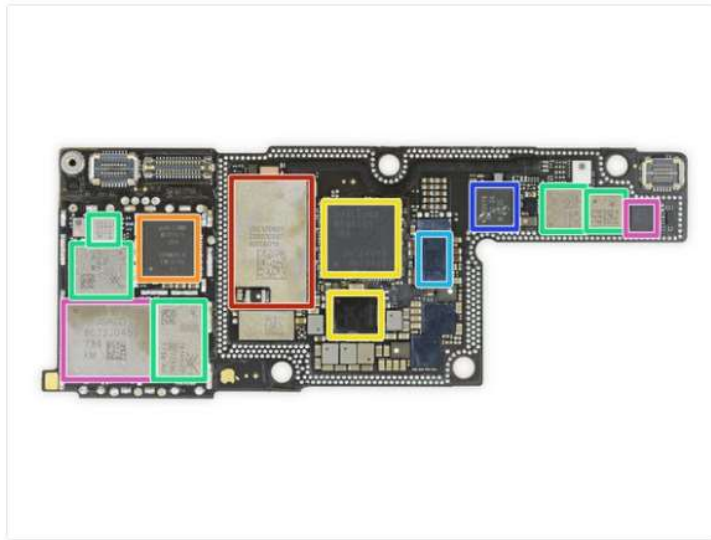
100. Apple infringes, contributes to the infringement of, and/or induces infringement of the '046 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '046 Patent, including, but not limited to, at least the Accused Products.

101. Apple has directly infringed and continues to directly infringe the '046 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '046 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '046 Patent, including, for example, card emulation and NFC payment functionality implemented by Apple Pay running on an Apple device, such as the representative Apple iPhone X. For example, these infrastructures infringe at least claim 1 of the '046 Patent.

102. For example, Apple has and continues to directly infringe at least claim 1 of the '046 Patent by making, using, offering to sell, selling and/or importing into the United States products that practice a method for mobile payment, the method comprising: causing a mobile device to capture data directly from a tag physically presented thereto, wherein the tag receives the data directly from a POS device and allows the mobile device to capture the data, the data embedded in the tag includes an electronic invoice and settlement information with a merchant associated with the POS device; extracting the electronic invoice from the captured data in the mobile device; displaying the electronic invoice on a display of the mobile device to show an amount to be paid by a user of the mobile device, wherein the mobile device is configured to execute an installed application therein to capture the data from the tag; receiving an entry by the mobile device, the entry including the amount for the invoice and optionally an additional amount from the user; calculating a total amount by adding the additional amount to the amount in the electronic invoice; generating a payment request in the mobile device in response to the electronic invoice after the user has chosen an electronic purse (e-purse) maintained locally in

the mobile device; displaying the electronic invoice on the display of the mobile device for the user to verify the payment request verifying the total amount with a balance in the e-purse, wherein said verifying the total amount with a balance in the e-purse is performed within the mobile device without sending the payment request to a payment gateway; displaying a denial of the payment request when the balance is less than the total amount; sending the payment request from the mobile device to the payment gateway, wherein the balance is sufficient to honor the payment request, the payment gateway sends a message directly to the POS device that a monetary transaction per the payment request sent from the mobile device has been successfully completed; and displaying a confirmation in the mobile device that the balance in the e-purse has been reduced by the total amount.

103. The Accused Products practice a method comprising causing a mobile device to capture data directly from a tag physically presented thereto, wherein the tag receives the data directly from a POS device and allows the mobile device to capture the data, the data embedded in the tag includes an electronic invoice and settlement information with a merchant associated with the POS device. For example, on information and belief, Apple Pay causes a mobile device, such as the Apple iPhone X, to capture data from an NFC tag, such as an NFC tag of a card reader at a POS, and allows the Apple iPhone X to capture data embedded in the tag including an electronic invoice and settlement information, such as the merchant's payment address.



- Apple/Murata USI 170821 339S00397 WiFi / Bluetooth module
- Qualcomm WTR5975 gigabit LTE transceiver.
- Qualcomm MDM9655 Snapdragon X16 LTE modem and PMD9655 PMIC. But Apple is dual-sourcing the modem, and TechInsights found an Intel XMM7480 (PMB9948) in their A1901 model. Even though the modem is capable of it, Apple isn't supporting Gigabit speeds with the Qualcomm part.
- Skyworks SKY78140-22 power amplifier, SKY77366-17 power amplifier, S770 6662, 3760 5418 1736
- Broadcom BCM59355 wireless charging controller
- NXP 80V18 PN80V NFC controller module
- Broadcom AFEM-8072, MMBB power amplifier module

<https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

104. The Accused Products further practice a method of extracting the electronic invoice from the captured data in the mobile device. For example, on information and belief, Apple Pay extracts the electronic invoice, such as the tokenized payment request identifying an amount, recipient, merchant, and financial institution.

When you use Apple Pay in stores

When you [use Apple Pay in stores](#) that accept contactless payments, Apple Pay uses Near Field Communication (NFC) technology between your device and the payment terminal. NFC is an industry-standard, contactless technology that's designed to work only across short distances. If your iPhone is on and detects an NFC field, it will present you with your default card. To send your payment information, you must authenticate using Face ID, Touch ID, or your passcode (except in Japan if you designate a Suica card for Express Transit). With Face ID or with Apple Watch, you must double-click the side button when the device is unlocked to activate your default card for payment.

After you authenticate your transaction, the Secure Element provides your Device Account Number and a transaction-specific dynamic security code to the store's point of sale terminal along with additional information needed to complete the transaction. Again, neither Apple nor your device sends your actual payment card number. Before they approve the payment, your bank, card issuer, or payment network can verify your payment information by checking the dynamic security code to make sure that it's unique and tied to your device.

<https://support.apple.com/en-us/HT203027>

When you use Apple Pay within apps or on the web

When you [use an app or a website that uses Apple Pay](#) in iOS, watchOS, or macOS, the app or website can check if you have Apple Pay enabled on that device. You can manage this option in Settings > Safari on your iOS device, and in the Privacy tab in Safari preferences on your Mac.

To securely transmit your payment information when you pay in apps or on the web, Apple Pay receives your encrypted transaction and re-encrypts it with a developer-specific key before the transaction information is sent to the developer or payment processor. This key helps ensure that only the app or the website that you're purchasing from can access your encrypted payment information. Websites must verify their domain every time they offer Apple Pay as a payment option. Like with in-store payments, Apple sends your Device Account Number to the app or website along with the transaction-specific dynamic security code. Neither Apple nor your device sends your actual payment card number to the app.

<https://support.apple.com/en-us/HT203027>

105. The Accused Products further practice a method of displaying the electronic invoice on a display of the mobile device to show an amount to be paid by a user of the mobile device, wherein the mobile device is configured to execute an installed application therein to capture the data from the tag. For example, on information and belief, Apple Pay displays the

amount of an invoice to be paid during a transaction on the display of a mobile device, such as the Apple iPhone X.⁸

Send and receive money with Apple Pay

Use Apple Cash or a debit card in the Wallet app to send and receive money in the Messages app.



<https://support.apple.com/en-us/HT207875>

106. The Accused Products practice a method of receiving an entry by the mobile device, the entry including the amount for the invoice and optionally an additional amount from the user. For example, on information and belief, Apple Pay receives an entry from an IOS device in a transaction log, the entry including the amount of an invoice and optionally an additional amount from the user, such as a tip entered at a POS terminal.⁹

⁸ See e.g. <https://www.cnet.com/news/apple-pay-google-pay-samsung-pay-best-mobile-payment-system-compared-nfc/>

⁹ See e.g. <https://squareup.com/help/us/en/article/6540-square-terminal-payments-faq.>

107. The Accused Products practice a method of calculating a total amount by adding the additional amount to the amount in the electronic invoice. For example, on information and belief, Apple Pay calculates a total amount to be paid and recorded by adding an amount of taxes (*e.g.* sales tax) and/or tips to the amount in the electronic invoice.¹⁰

108. The Accused Products practice a method of generating a payment request in the mobile device in response to the electronic invoice after the user has chosen an electronic purse (e-purse) maintained locally in the mobile device. For example, on information and belief, Apple Pay generates a payment request in an IOS device after a user has chosen an electronic purse (*e.g.* Apple Pay) maintained locally in the device. For example, given selection of a payment card applet within Apple Pay, the payment card applet generates a payment token, such as by generating transaction information based on operations keys, device account numbers, tokenized card information, and/or cryptograms.

109. The Accused products further display the electronic invoice on the display of the mobile device for a user to verify the payment request. For example, on information and belief, Apple Pay causes an IOS device, and/or a POS to display the amount of a transaction for a user to verify, such as by actuating a payment button, entering a PIN or other security information, or tapping the device to effect payment.

¹⁰ See *e.g.* <https://squareup.com/help/us/en/article/6540-square-terminal-payments-faq>.

Send and receive money with Apple Pay

Use Apple Cash or a debit card in the Wallet app to send and receive money in the Messages app.



110. The Accused Products further practice verifying the total amount with a balance in the e-purse, wherein said verifying the total amount with a balance in the e-purse is performed within the mobile device without sending the payment request to a payment gateway. For example, on information and belief, Apple Pay verifies a balance of existing funds or available credit by checking information stored in a secure element, without the need for sending a request to a payment gateway.

What you need

You and the person that you're sending money to or receiving the money from must:^{2,3}

- Have a compatible device with the latest iOS or watchOS.
- Use [two-factor authentication with your Apple ID](#) and [sign in to iCloud and iMessage with the same Apple ID](#) on any device that you want to use to send or receive money.
- If you're sending money to someone, make sure there's enough money on [your Apple Cash card](#) or [an eligible debit card in Wallet](#).

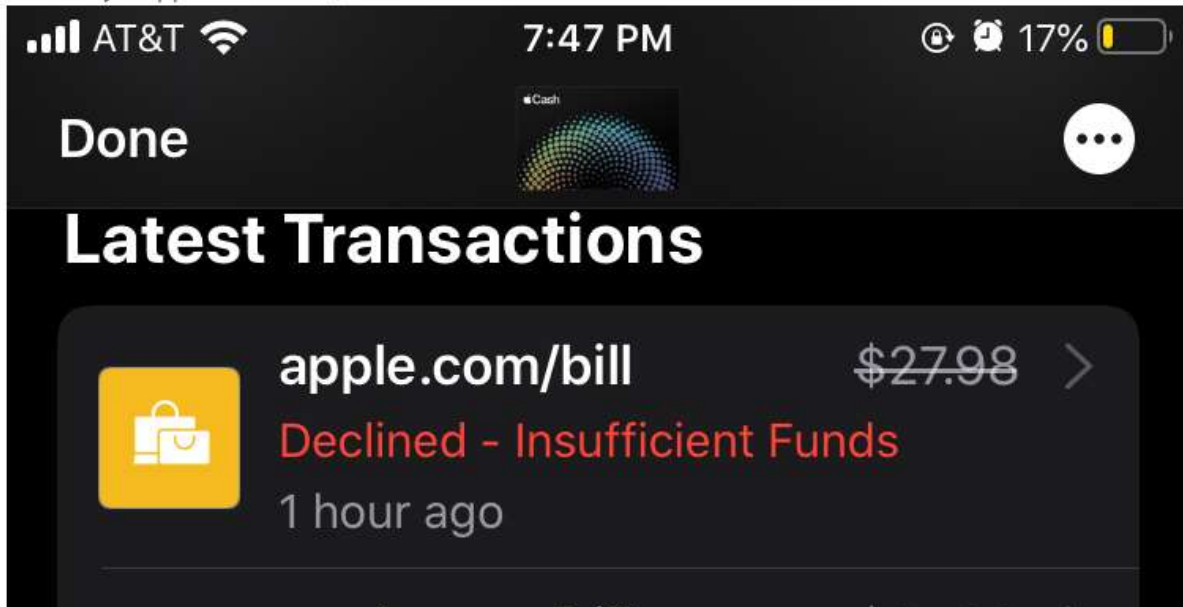
<https://support.apple.com/en-us/HT207875#whatyouneed>

111. The Accused Products display a denial of the payment request when the balance is less than the total amount. For example, on information and belief, Apple Pay causes an IOS device to display a screen showing that a payment was declined when there are insufficient funds to settle a transaction.

Try to send money

After each step, see if you can send money:


1. Restart your device.
2. See if you, or the recipient, need to [verify your identity](#).
3. Make sure that the recipient is eligible to receive Apple Cash payments. The recipient must [meet the requirements](#), and they need the latest versions of [iOS](#) or [watchOS](#). If the recipient doesn't meet the requirements, you'll see a message that they aren't eligible in Messages.
4. If you're asked to choose a lower amount or try again later, you might have encountered a limit. [Learn about the limits for sending and receiving money](#).
5. If you're trying to send money directly from your debit card in Apple Pay, and you see a message that the payment couldn't be completed because it was declined by your bank, contact your bank or card issuer.



112. The Accused Products further practice sending the payment request from the mobile device to the payment gateway, wherein the balance is sufficient to honor the payment request, the payment gateway sends a message directly to the POS device that a monetary transaction per the payment request sent from the mobile device has been successfully completed. For example, on information and belief, Apple Pay sends the payment request from the IOS device to the payment gateway, such as the payment server of a card issuer and/or merchant. For example, on information and belief, when there is sufficient balance in a given payment card of Apple Pay, such as funds or credit available based on a value in a secure element, the payment gateway sends a message to the POS that the transaction is successful, and the POS displays a success message.

113. The Accused Products further practice displaying a confirmation in the mobile device that the balance in the e-purse has been reduced by the total amount. For example, on information and belief, Apple Pay causes an IOS device to display a confirmation that balance in the e-purse has been reduced by the total amount, such as by displaying a lower account balance.

Find your Apple Cash card info

- On your iPhone, open the Wallet app, tap your Apple Cash card, then tap the more button .
- On your iPad, open the Settings app, tap Wallet & Apple Pay, then tap your Apple Cash card
- For Apple Watch, open the Apple Watch app on your iPhone, tap Wallet & Apple Pay, then tap your Apple Cash card.

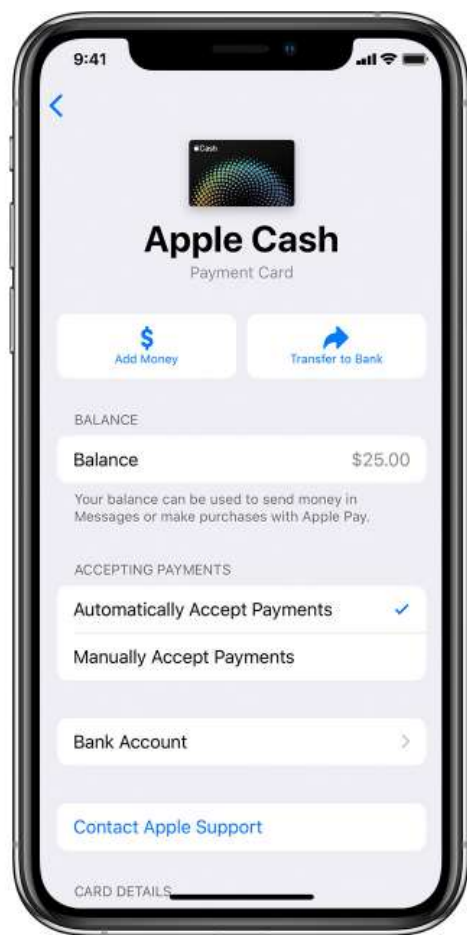
From here, you can see your balance, the Privacy Policy, and the Terms and Conditions. To make sure that your balance is up to date, you need a cellular or WI-FI connection.

If you're part of a Family Sharing group, you can also see family members who are eligible for [Apple Cash Family](#).

If you're asked for a PIN

Apple Cash doesn't require a PIN because every payment is authenticated by Face ID, Touch ID, or a secure passcode.

If you're prompted by a terminal to enter a PIN to complete a debit transaction, enter a four-digit code, like 0000.



<https://support.apple.com/en-us/HT207883>

114. Apple has had knowledge and notice of the '046 Patent since it issued on March 24, 2020.

115. Apple has had knowledge of how its products infringe the '046 Patent since at least September 7, 2021. Despite that knowledge, Apple continues to infringe the '046 Patent both directly and indirectly.

116. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '046 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under

the doctrine of equivalents, through their use of the inventions claimed in the '046 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly infringe the '046 Patent. Apple performs these affirmative acts with knowledge of the '046 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '046 Patent.

117. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '046 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '046 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '046 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are known by Apple to be especially made or adapted for use in the infringement of the '046 Patent. Apple performs these affirmative acts with knowledge of the '046 Patent and with intent, or willful blindness, that they cause the direct infringement of the '046 Patent.

118. Because of Apple's direct and indirect infringement of the '046 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

119. Because of Apple's direct and indirect infringement of the '046 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

COUNT VI
(Infringement of the '724 Patent)

120. Paragraphs 1 through 25 are incorporated herein by reference as if fully set forth in their entireties.

121. RFCyber has not licensed or otherwise authorized Apple to make, use, offer for sale, sell, or import any products that embody the inventions of the '724 Patent.

122. Apple infringes, contributes to the infringement of, and/or induces infringement of the '724 Patent by making, using, selling, offering for sale, distributing, exporting from, and/or importing into the United States products and/or methods covered by one or more claims of the '724 Patent, including, but not limited to, at least the Accused Products.

123. Apple has directly infringed and continues to directly infringe the '724 Patent, either literally or under the doctrine of equivalents, without authority and in violation of 35 U.S.C. § 271, by making, using, offering to sell, selling and/or importing into the United States products that satisfy each and every limitation of one or more claims of the '724 Patent. Upon information and belief, these products include the Accused Products that practice the methods and systems covered by the '724 Patent, including, for example, card emulation and NFC payment functionality implemented by Apple Pay running on an Apple device, such as the representative Apple iPhone X. For example, these products infringe at least claim 1 of the '724 Patent.

124. For example, Apple has and continues to directly infringe at least claim 1 of the '724 Patent by making, using, offering to sell, selling and/or importing into the United States products that comprise a mobile device for emulating a plurality of cards, the mobile device comprising: a display screen showing a list of a plurality of applications for a user of the mobile device to select one therefrom, each application corresponding to one card in the plurality of cards; a secure element (SE) including: an emulator device; a memory storing a module, when the module is executed by the secure element, the secure element configured to: receive and install key sets of a Supplementary Secured Domain (SSD); establish, by the secure element based on the key sets, a secure communication channel with a dedicated server; receive and install an application from the dedicated server, each application including corresponding application data sets and a locked or unlocked status; receive, from the plurality of applications, a user selection of a first application corresponding to a first card; determine that the first application has a locked or unlocked status and is activated, in response to said determining that the first application has an unlocked status and is activated, load the first application to the emulator device, along with corresponding first application data sets; receive, from the plurality of applications, a user selection of a second application corresponding to a second card; determine that the second application has a locked or unlocked status and is activated; in response to said determining that the second application has an unlocked status and is activated, replace out of the emulator device, a portion of or in entirety, the first application, wherein said replacing out of the emulator device a portion of the first application further comprises retaining the portion of the corresponding first application data sets to be utilized by the second application; load the second application to the emulator device along with corresponding second application data sets; and increment a counter for each successful application replacement,

wherein the mobile device performs functions of the second card when the first application is replaced out of the emulator device and the second application is loaded in the emulator device.

125. The Accused Products comprise a display screen showing a list of a plurality of applications for a user of the mobile device to select one therefrom, each application corresponding to one card in the plurality of cards. For example, the Apple iPhone X comprises a display screen that shows a list of applications, such as applications corresponding to various payment cards in Apple Pay:

Pay with a different card instead of your default card

Here's how to switch cards:

- iPhone with Face ID: Double-click the side button. When your default card appears, tap it, then tap to choose another card. Glance at your iPhone to authenticate with Face ID, then hold the top of your device near the reader to pay.
- iPhone with Touch ID: Hold your device near the reader without resting your finger on Touch ID. When your default card appears, tap it, then tap to choose another card. Rest your finger on Touch ID to pay.
- Apple Watch: Double-click the side button. When your default card appears, swipe left or right to choose another card. Hold your watch near the reader to pay.

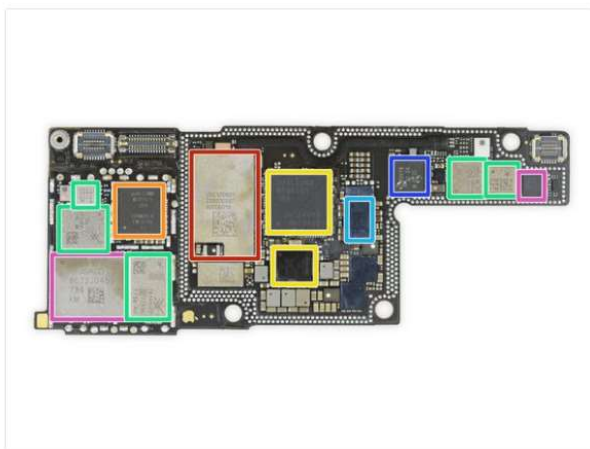
<https://support.apple.com/en-us/HT201239>



(Apple Card and Apple Cash are available only in the U.S.)

<https://support.apple.com/guide/iphone/set-up-apple-pay-iph9b7f53382/ios>

126. The Accused Products comprise a secure element that includes an emulator device and a memory storing a module. For example, the Apple iPhone X comprises an NFC Module and secure element with an emulator:



- Apple/Murata US1 170821 339S00397 WiFi / Bluetooth module
- Qualcomm WTR5975 gigabit LTE transceiver.
- Qualcomm MDM9655 Snapdragon X16 LTE modem and PMD9655 PMIC. But Apple is dual-sourcing the modem, and TechInsights found an Intel XMM7480 (PMB9948) in their A1901 model. Even though the modem is capable of it, Apple isn't supporting Gigabit speeds with the Qualcomm part.
- Skyworks SKY78140-22 power amplifier, SKY77366-17 power amplifier, S770 6662, 3760 5418 1736
- Broadcom BCM59355 wireless charging controller
- NXP 80V18 PN80V NFC controller module
- Broadcom AFEM-8072, MMMB power amplifier module

<https://www.ifixit.com/Teardown/iPhone+X+Teardown/98975>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.


After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

View the information for a card and change its settings

1. In Wallet, tap the card.

Note: The last transaction may appear, showing an authorized amount that may differ from the amount of the payment charged to your account. For example, a gas station may request an authorization of \$99, even though you pumped only \$25 worth of gasoline. To see the final charges, see the statement from your card issuer, which includes all Apple Pay transactions.

2. Tap , then do any of the following:

- Tap Transactions to view your recent history. To hide this information, turn off Transaction History. To view all your Apple Pay activity, see the statement from your card issuer.
- View the last four digits of the card number and Device Account Number—the number transmitted to the merchant.
- Change the billing address.
- Remove the card from Wallet.

<https://support.apple.com/guide/iphone/manage-cards-and-activity-iph7b666943a/ios>

127. The module is executed by the secure element, the secure element configured to receive and install key sets of a Supplementary Secured Domain (SSD), and establish, by the secure element based on the key sets, a secure communication channel with a dedicated server. For example, the Apple iPhone X comprises a secure element that receives and installs key sets, such as keys that are known only to the payment network or card issuer and the secure element applets' security domain. The iPhone X then communicates with the payment network servers over a secured channel using those keys.

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-seccb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>

128. The module is executed by the secure element, the secure element configured to receive and install an application from the dedicated server, each application including corresponding application data sets and a locked or unlocked status. For example, the iPhone X receives an application, such as a card issuer applet, from the server. The applet includes data sets (such as the Device Account Number) and a locked or unlocked status (such as a card locked or unlocked status):

Secure Element

The Secure Element hosts a specially designed applet to manage Apple Pay. It also includes applets certified by payment networks or card issuers. Credit, debit, or prepaid card data is sent from the payment network or card issuer encrypted to these applets using keys that are known only to the payment network or card issuer and the applets' security domain. This data is stored within these applets and protected using the Secure Element's security features. During a transaction, the terminal communicates directly with the Secure Element through the near-field-communication (NFC) controller over a dedicated hardware bus.

<https://support.apple.com/guide/security/secure-element-and-nfc-controller-seccb53a35f0/web>

When you add credit, debit, prepaid, or transit cards

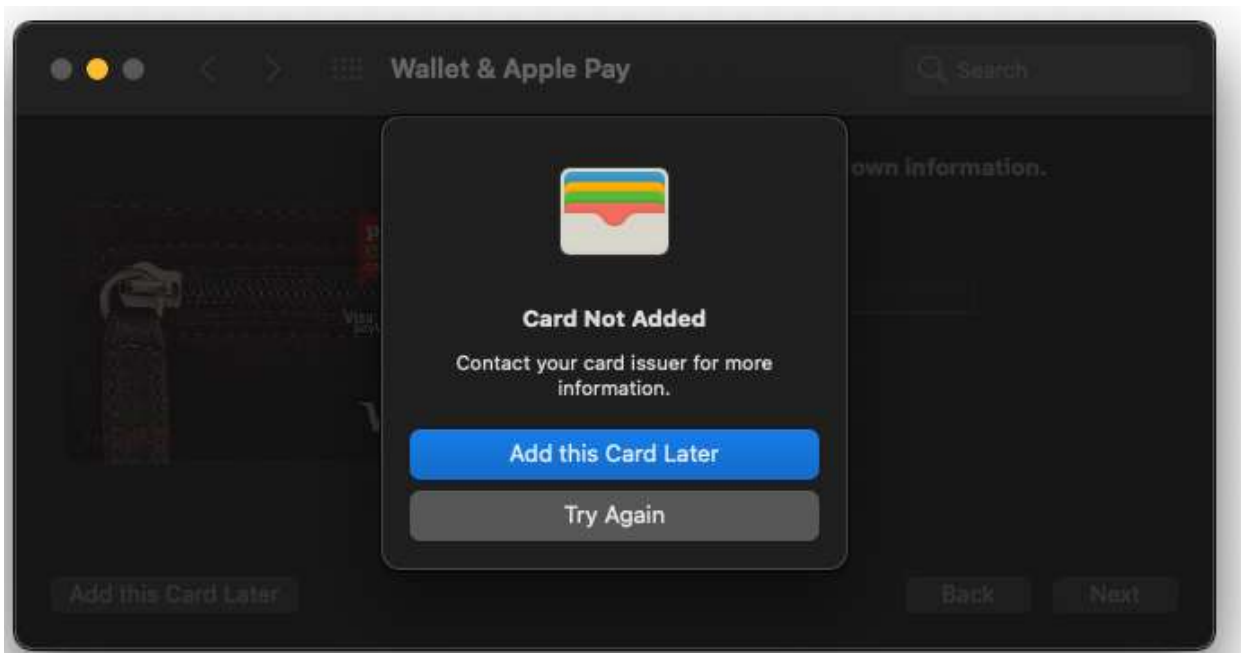
When you [add a credit, debit, prepaid, or transit card](#) (where available) to Apple Pay, information that you enter on your device is encrypted and sent to Apple servers. If you use the camera to enter the card information, the information is never saved on your device or photo library.

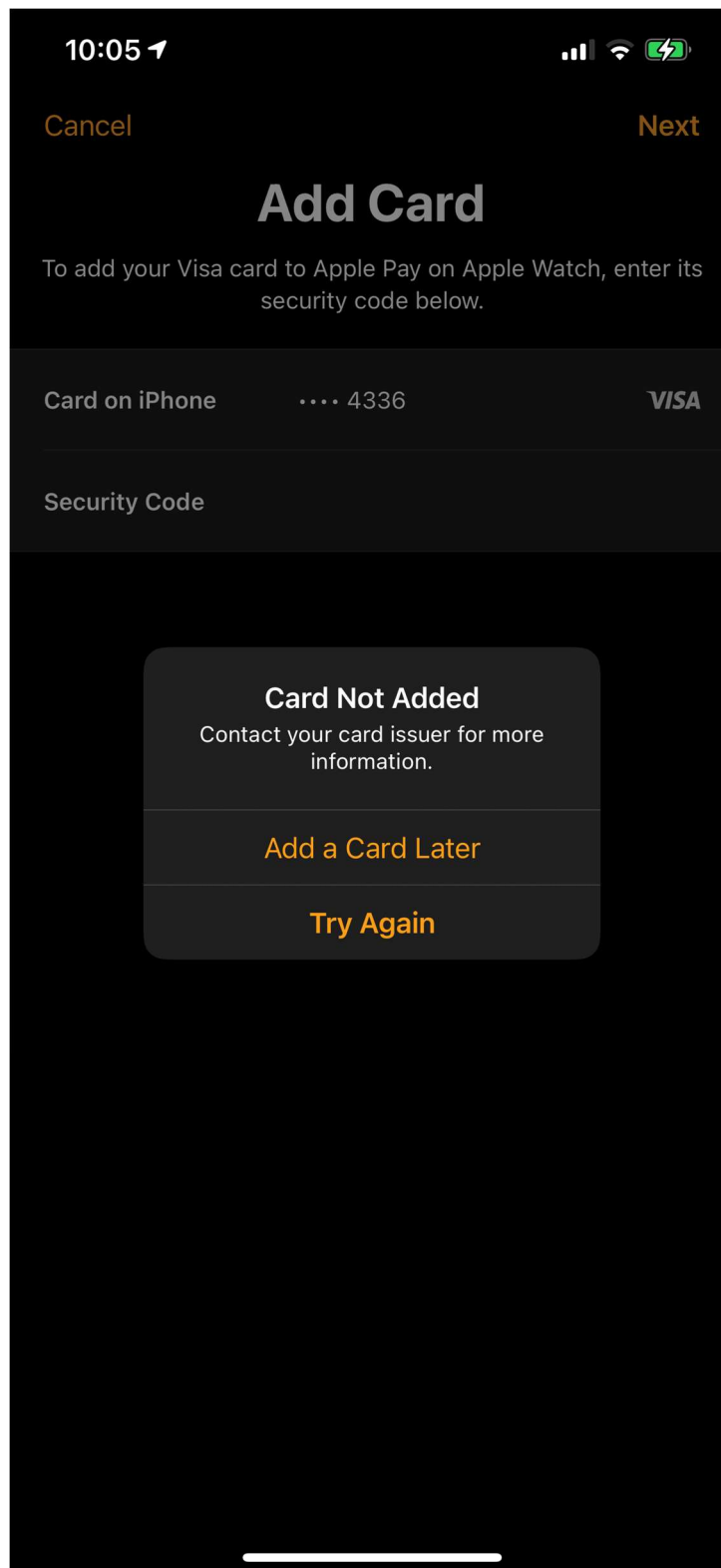
Apple decrypts the data, determines your card's payment network, and re-encrypts the data with a key that only your payment network (or any providers authorized by your card issuer for provisioning and token services) can unlock.

Information that you provide about your card, whether certain device settings are enabled, and device use patterns—such as the percent of time the device is in motion and the approximate number of calls you make per week—may be sent to Apple to determine your eligibility to enable Apple Pay. Information may also be provided by Apple to your card issuer, payment network, or any providers authorized by your card issuer to enable Apple Pay, to determine the eligibility of your card, to set up your card with Apple Pay, and to prevent fraud.

After your card is approved, your bank, your bank's authorized service provider, or your card issuer creates a device-specific Device Account Number, encrypts it, and sends it along with other data (such as the key used to generate dynamic security codes that are unique to each transaction) to Apple. The Device Account Number can't be decrypted by Apple but is stored in the Secure Element—an industry-standard, certified chip designed to store your payment information safely—on your device. Unlike with usual credit or debit card numbers, the card issuer can prevent its use on a magnetic stripe card, over the phone, or on websites. The Device Account Number in the Secure Element is isolated from iOS, watchOS, and macOS, is never stored on Apple servers, and is never backed up to iCloud.

<https://support.apple.com/en-us/HT203027>





129. The module is executed by the secure element, the secure element configured to receive, from the plurality of applications, a user selection of a first application corresponding to

a first card, determine that the first application has a locked or unlocked status and is activated, and in response to said determining that the first application has an unlocked status and is activated, load the first application to the emulator device, along with corresponding first application data sets. For example, the iPhone X allow a user to select a card for payment, which selects the associated applet. On information and belief, the iPhone X will determine if the card (and therefore its applet) is unlocked and activated. On information and belief, the iPhone X will necessarily load the applet (such as the card applet) into the emulator along with the applet's data sets (such as the Device Account Number).

Pay with a different card instead of your default card

Here's how to switch cards:

- iPhone with Face ID: Double-click the side button. When your default card appears, tap it, then tap to choose another card. Glance at your iPhone to authenticate with Face ID, then hold the top of your device near the reader to pay.
- iPhone with Touch ID: Hold your device near the reader without resting your finger on Touch ID. When your default card appears, tap it, then tap to choose another card. Rest your finger on Touch ID to pay.
- Apple Watch: Double-click the side button. When your default card appears, swipe left or right to choose another card. Hold your watch near the reader to pay.

<https://support.apple.com/en-us/HT201239>

130. The module is executed by the secure element, the secure element configured to receive, from the plurality of applications, a user selection of a second application corresponding to a second card; determine that the second application has a locked or unlocked status and is activated; in response to said determining that the second application has an unlocked status and is activated, replace out of the emulator device, a portion of or in entirety, the first application, wherein said replacing out of the emulator device a portion of the first application further comprises retaining the portion of the corresponding first application data sets to be utilized by

the second application; and load the second application to the emulator device along with corresponding second application data sets. For example, the iPhone X allow a user to select a different card for payment, which selects the associated applet. On information and belief, the iPhone X will determine if the second card (and therefore its applet) is unlocked and activated. On information and belief, the iPhone X will necessarily replace the first applet with second the applet (such as the second card applet) into the emulator along with the applet's data sets (such as the second card's Device Account Number).

Pay with a different card instead of your default card

Here's how to switch cards:

- iPhone with Face ID: Double-click the side button. When your default card appears, tap it, then tap to choose another card. Glance at your iPhone to authenticate with Face ID, then hold the top of your device near the reader to pay.
- iPhone with Touch ID: Hold your device near the reader without resting your finger on Touch ID. When your default card appears, tap it, then tap to choose another card. Rest your finger on Touch ID to pay.
- Apple Watch: Double-click the side button. When your default card appears, swipe left or right to choose another card. Hold your watch near the reader to pay.

<https://support.apple.com/en-us/HT201239>

131. The module is executed by the secure element, the secure element configured to increment a counter for each successful application replacement, wherein the mobile device performs functions of the second card when the first application is replaced out of the emulator device and the second application is loaded in the emulator device. For example, the iPhone X maintains a counter pointing to the currently selected card, and the counter's state is viewable from the card selection screen:



<https://support.apple.com/en-us/HT201239> (annotations added)

132. When a different card is selected, the counter is incremented to point to the new card, and the card selection screen is updated to indicate the new card:

Pay with a different card instead of your default card

Here's how to switch cards:

- iPhone with Face ID: Double-click the side button. When your default card appears, tap it, then tap to choose another card. Glance at your iPhone to authenticate with Face ID, then hold the top of your device near the reader to pay.
- iPhone with Touch ID: Hold your device near the reader without resting your finger on Touch ID. When your default card appears, tap it, then tap to choose another card. Rest your finger on Touch ID to pay.
- Apple Watch: Double-click the side button. When your default card appears, swipe left or right to choose another card. Hold your watch near the reader to pay.

<https://support.apple.com/en-us/HT201239>

133. The emulator necessarily performs the functions of the second card (such as payment for transactions), when the first application is replaced out of the emulator device and the second application is loaded in the emulator device.

134. Apple has had knowledge and notice of the '724 Patent since it issued on May 25, 2021.

135. Apple has had knowledge of how its products infringe the '724 Patent since at least September 7, 2021. Despite that knowledge, Apple continues to infringe the '724 Patent both directly and indirectly.

136. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '724 Patent, as provided by 35 U.S.C. § 271(b), by inducing infringement by others, such as Apple's customers and end-users, in this District and elsewhere in the United States. For example, Apple's customers and end-users directly infringe, either literally or under the doctrine of equivalents, through their use of the inventions claimed in the '724 Patent. Apple induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting that

they use the Accused Products in an infringing manner, including technical support, marketing, product manuals, advertisements, and online documentation. Because of Apple's inducement, Apple's customers and end-users use Accused Products in a way Apple intends and directly infringe the '724 Patent. Apple performs these affirmative acts with knowledge of the '724 Patent and with the intent, or willful blindness, that the induced acts directly infringe the '724 Patent.

137. Apple has indirectly infringed and continues to indirectly infringe one or more claims of the '724 Patent, as provided by 35 U.S.C. § 271(c), by contributing to direct infringement by others, such as customers and end-users, in this District and elsewhere in the United States. Apple's affirmative acts of selling and offering to sell the Accused Products in this District and elsewhere in the United States and causing the Accused Products to be manufactured, used, sold and offered for sale contributes to others' use and manufacture of the Accused Products such that the '724 Patent is directly infringed by others. The accused components within the Accused Products are material to the invention of the '724 Patent, are not staple articles or commodities of commerce, have no substantial non-infringing uses, and are known by Apple to be especially made or adapted for use in the infringement of the '724 Patent. Apple performs these affirmative acts with knowledge of the '724 Patent and with intent, or willful blindness, that they cause the direct infringement of the '724 Patent.

138. Because of Apple's direct and indirect infringement of the '724 Patent, RFCyber has suffered, and will continue to suffer, damages in an amount to be proved at trial.

139. Because of Apple's direct and indirect infringement of the '724 Patent, RFCyber has suffered, and will continue to suffer, irreparable harm for which there is no adequate remedy at law, unless Apple's infringement is enjoined by this Court.

DEMAND FOR JURY TRIAL

140. Plaintiff hereby demands a jury for all issues so triable.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff prays for relief against Defendant as follows:

- a. Entry of judgment declaring that Defendant infringes one or more claims of each of the Patents-in-Suit;
- b. Entry of judgment declaring that Defendant's infringement of the Patents-in-Suit is willful;
- c. An order awarding damages sufficient to compensate Plaintiff for Defendant's infringement of the Patents-in-Suit, but in no event less than a reasonable royalty, including supplemental damages post-verdict, together with pre-judgment and post-judgment interest and costs;
- d. Enhanced damages pursuant to 35 U.S.C. § 284;
- e. Entry of judgment declaring that this case is exceptional and awarding Plaintiff its costs and reasonable attorney fees pursuant to 35 U.S.C. § 285;
- f. An accounting for acts of infringement;
- g. Such other equitable relief which may be requested and to which the Plaintiff is entitled; and
- h. Such other and further relief as the Court deems just and proper.

Dated: December 2, 2021

Respectfully submitted,

/s/ Raymond W. Mort, III

Raymond W. Mort, III
Texas State Bar No. 00791308
Email: raymort@austinlaw.com
THE MORT LAW FIRM, PLLC
100 Congress Avenue, Suite 2000
Austin, Texas 78701
Tel/Fax: 512-865-7950

OF COUNSEL:

Alfred R. Fabricant (*pro hac vice* to be filed)
NY Bar No. 2219392
Email: ffabricant@fabricantllp.com
Peter Lambrianakos (*pro hac vice* to be filed)
NY Bar No. 2894392
Email: plambrianakos@fabricantllp.com
Vincent J. Rubino, III (*pro hac vice* to be filed)
NY Bar No. 4557435
Email: vrubino@fabricantllp.com
Richard M. Cowell (*pro hac vice* to be filed)
NY Bar No. 4617759
Email: rcowell@fabricantllp.com
FABRICANT LLP
411 Theodore Fremd Road, Suite 206 South
Rye, New York 10580
Telephone: (212) 257-5797
Facsimile: (212) 257-5796

***ATTORNEYS FOR PLAINTIFF RFCYBER
CORP.***