Strong Customer Authentication for Apple Pay on MacBook Air 2020 with M1 running macOS Big Sur 11.3.1

Security Target

Version 1.6
April 15, 2022

Apple
One Apple Park Way
Cupertino, CA 95014

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this document is to define the Target of Evaluation for meeting the requirements of Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market (PSD2) and the Commission Delegated Regulation (EU) 2018/389 of 27 November 2017, focusing on Strong Customer Authentication and Dynamic Linking for Apple Pay.

1.2. Abbreviations

Abbreviation	Meaning
AP	Application Processor
API	Application Programming Interface
AR	Authorization Random
CL	Contactless
CRS	Contactless Registry Service
CVV	Card Verification Value
HSM	Hardware Security Module
I/O	Input / Output
MAC	Message Authentication Code
NFC	Near Field Communication
OS	Operating System
PNO	Payment Network Operator
PSD2	Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015
SCA	Strong Customer Authentication
SIP	System Integrity Protection
SKS	Secure Key Store
SSE	An application in the Secure Enclave managing the pairing between the Secure Enclave and the Secure Element
TSM	Trusted Service Manager
TSP	Token Service Provider
UID	Unique Identifier

2. ST Introduction

2.1. Target of Evaluation Reference

This Security Target is identified with the following information:

Security Target identifiers	
Title	Strong Customer Authentication for Apple Pay, on MacBook Air 2020 with M1 running macOS Big Sur 11.3.1, Security Target
Version	1.6
Date	April 15, 2022
Developer	Apple Inc.

The TOE platform is a MacBook Air 2020 with M1 running macOS Big Sur 11.3.1, with the following platform identifiers:

Platform identifiers	
TOE	MacBook Air 2020 with M1 running macOS Big Sur 11.3.1
Device	Model: MacBook Air 2020
Operating System	macOS Big Sur 11.3.1 (Build 20E241)
Developer	Apple Inc.

The TOE consists of a range of hardware and software components as listed below. Note that all components are developed by Apple.

TOE Component	Version	Description
Apple Pay App (Wallet)	App part of macOS 11.3.1	Authentication policy on data and services In-app transaction data management
Application Processor (AP)	M1	Authentication policy on data and services Transaction data management
Biometric Sensor (Touch ID)	Gen 3	Sensor for fingerprint capture
Boot Loader	macOS Big Sur 11.3.1 (20E241)	Allows the device to start and boot the operating system

TOE Component	Version	Description
Secure Enclave	sepOS part of macOS Big Sur 11.3.1	Authentication Setup: • Enrollment of the authentication material • User authentication verification Authentication Prover: • Password verification • Biometrics matching • Authentication policy on data
macOS Platform		Device operating system platform (macOS Big Sur 11.3.1 (20E241)) with the following Apple Pay services that are included in the TOE:
• **Console**	macOS Big Sur 11.3.1 (20E241)	Provides the functionality for input/output (I/O)
• **Logind**	macOS Big Sur 11.3.1 (20E241)	Provides functionality for managing user logins and sessions
• **NFCd**	macOS Big Sur 11.3.1 (20E241)	Provides communication layer between the TOE and the Secure Element
• **Safari**	macOS Big Sur 11.3.1 (20E241)	Browser
• **System Preferences**	macOS Big Sur 11.3.1 (20E241)	Allows the user to indicate their preferred system preferences for the device, operating system and applications.

The Secure Element of the device is out of scope of this evaluation.

Note: *In the evaluated configuration the cryptographic modules are supplied by Apple as part of macOS and sepOS. Readers may draw some assurance from the conformance to FIPS 140-2 certified by the Cryptographic Module Validation Program for corecrypto for each major release (Apple corecrypto User Space Module, Apple corecrypto Kernel Space Module and the Apple Secure Key Store Cryptographic Module).*

Additionally, readers should note that the browser, Safari, is evaluated for each major macOS release using the collaborative PP (cPP); Protection Profile for Application Software Version 1.3.

2.2. TOE Guidance

References to the TOE guidance document:

Apple Pay Guidance		
Strong Customer Authentication for Apple Pay on MacBook Air 2020 with M1 running macOS Big Sur 11.3.1: Guidance	[AGD]	Guidance, version 1.3

2.3. TOE Overview

The TOE includes the components implementing Strong Customer Authentication and Dynamic Linking in Apple Pay. The TOE is the MacBook Air 2020 with M1 running macOS Big Sur 11.3.1 operating system.

The operating system, macOS Big Sur 11.3.1, manages the device hardware, provides Apple Pay functionalities, and provides the technologies required to enforce Strong Customer Authentication (SCA) and Dynamic Linking for Apple Pay e-commerce payments. Dynamic Linking for a transaction is the link between the authentication code generated upon successful SCA, with both the transaction's original specific amount and the identity of the payee.

macOS Big Sur 11.3.1 provides a consistent set of capabilities allowing the supervision of enrolled devices. This includes the preparation of devices for deployment, the subsequent management of the devices, and the termination of management.

The TOE platform protects itself by having its own code and data protected from unauthorized access (using hardware provided memory protection features), by securing user and TOE Security Functionality (TSF) data, by ensuring the integrity and authenticity of TSF updates and downloaded applications, and by locking the TOE upon user request or after a defined time of user inactivity.

The TOE provides protection of data at rest and access control mechanisms for use by applications. Access control for data and services, including Apple Pay, rely on the enforcement of user authentication.

User authentication on an enrolled device is provided by a user-defined password and user enrolled biometrics. The minimum length of the password, password rules, and the maximum number of consecutive failed authentication attempts is statically set by Apple for each macOS release. Biometrics are enrolled and managed by the user. Up to 5 fingerprints can be enrolled on a single TOE by the user.

For greater convenience when using multiple Apple devices, some devices can automatically unlock others in certain situations. For this security target, the considered usage is: *A Mac can be unlocked by an Apple Watch*. This can be enabled by the user with "Auto Unlock with Apple Watch" setting in the System Preferences.

When Auto Unlock with Apple Watch is enabled, a MacBook running macOS High Sierra or later can be unlocked using a paired Apple Watch Series 3 or later. The user can also use their Apple Watch to approve other requests to enter their administrator password.

The Secure Enclave is responsible for ensuring user authorization (the combination of user authentication and user intent) before a payment is made.

The TOE relies on its environment to perform Apple Pay transactions; the transaction data is always processed by the Secure Element, which is outside the TOE boundary. The Secure Element only allows a payment to be made after it receives authorization from the Secure Enclave. For each transaction, the Secure Element utilizes the payment applets to generate a payment cryptogram. This cryptogram and the Device Account Number form a transaction-specific dynamic security code, which is provided to the payment network and the card issuer, allowing them to verify each transaction.

2.4. Description of the Apple Pay Service

2.4.1. Card provisioning

When a user adds a credit, debit, or prepaid card (including store cards) to Apple Wallet, the device encrypts the card information and securely sends it, along with other information about the user's account and device, through Apple Pay servers to the card issuer or the card issuer's authorized service provider. Using this information, the card issuer (or their service provider) will determine whether to approve adding the card to Apple Wallet.

Apple Pay uses three server-side calls to send and receive communication with the card issuer or network as part of the card provisioning process: Required Fields, Check Card, and Link and Provision. The card issuer or network uses these calls to verify, approve, and add cards to Apple Wallet. These client-server sessions are protected for confidentiality and integrity using TLS protocol.

The full card numbers are never stored on the device or on Apple servers. Instead, a unique Device Account Number is created, encrypted, and then stored in the Secure Element. This unique Device Account Number is encrypted in such a way that Apple cannot access it. The Device Account Number is unique and different from 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 located in the Secure Element is isolated from the TOE, never stored on Apple servers, and never backed up to iCloud.

2.4.1.1. Adding a credit or debit card manually to Apple Pay

To add a card manually, the name, card number, expiration date, and card verification value (CVV) are used to facilitate the provisioning process. From within System Preferences, the Wallet app, or the Apple Watch app, users can enter that information by typing or using the camera on the device. When the user uses the camera to capture the card information, Apple Wallet attempts to populate the name, card number, and expiration date. The photo is never saved to the device nor stored in the photo library. When the camera is used, the user still needs to enter the CVV. After all the fields are filled in, the Check Card process verifies the fields other than the CVV. They are encrypted and sent to the Apple Pay Server. If a terms and conditions ID is returned with the Check Card process, Apple downloads and displays the terms and conditions of the card issuer to the user. If the user accepts the terms and conditions, Apple sends the ID of the terms that were accepted as well as the CVV to the Link and Provision process.

2.4.1.2. Adding credit or debit cards from an iTunes Store account to Apple Pay

For a credit or debit card on file with iTunes, the user may be required to re-enter their Apple ID password. The card number is retrieved from iTunes and the Check Card process is initiated. If the card is eligible for Apple Pay, the Apple Wallet application will download and display terms and conditions, then send along the term's ID and the card security code to the Link and Provision process. Additional verification may occur for iTunes account cards on file.

2.4.1.3. Adding credit or debit cards from a card issuer's website

Some card issuers may also support provisioning on their websites under certain circumstances. The user can use the "Sign in with Apple" function to select an eligible device associated with their Apple ID account to provision to. The provisioning flow is similar to that used for the other methods of adding cards.

2.4.1.4. Additional verification

A card issuer can decide whether a credit or debit card requires additional verification. Depending on what is offered by the card issuer, the user may be able to choose between different options for additional verification, such as a text message, email, or customer service call to complete the verification. For text messages or email, the user selects from contact information the issuer has on file. A code will be sent, which must be entered into System Preferences, the Wallet app, or the Apple Watch app. For customer service, the issuer performs their own communication process.

2.4.2. Payment authorization

The Secure Element will only allow a payment to be made after it receives authorization from the Secure Enclave. Authorization includes confirming that the user has authenticated with Touch ID or the device password and double-press the Touch ID sensor. Touch ID is the default method if available, but the

password can be used at any time. A password is automatically offered after three unsuccessful attempts to match a fingerprint; after five unsuccessful attempts, the password is required. A password is also required when Touch ID is not configured or not enabled for Apple Pay.

2.4.2.1. Using a shared pairing key

Communication between the Secure Enclave and the Secure Element takes place over a serial interface, with the Secure Element connected to the Near Field Communication (NFC) controller, which in turn is connected to the Application Processor. Although not directly connected, the Secure Enclave and the Secure Element can communicate securely using a shared pairing key that is provisioned during the manufacturing process. The pairing key is generated inside the Secure Enclave from the device's unique identifier (UID) key and the Secure Element unique identifier. The pairing key is then securely transferred from the Secure Enclave to a hardware security module (HSM) in the factory, which has the key material required to then inject the pairing key into the Secure Element. Authentication of the communication between the Secure Enclave and Secure Element is based on AES and utilizes a MAC-specific key derived from the shared pairing key. The Secure Enclave and Secure Element use this MAC-specific key to generate and verify the card cryptogram and host MAC to establish mutual authentication. The Secure Enclave initiates this process by sending a host challenge to the Secure Element. The Secure Element responds with the card challenge and card cryptogram, which the Secure Enclave uses to authenticate it. The Secure Enclave then includes a host MAC along with the transaction and authentication details as described below. The Secure Element uses the host MAC to authenticate the Secure Enclave before continuing with processing of the transaction.

2.4.2.2. Authorizing a secure transaction

When the user authorizes a transaction, the Secure Enclave sends signed data about the type of authentication and details about the type of transaction (contactless or e-Commerce) to the Secure Element, tied to an Authorization Random (AR) value. The AR is generated in the Secure Enclave when a user first provisions a credit card and persists while Apple Pay is enabled, protected by the Secure Enclave's encryption and anti-rollback mechanism. It is securely delivered to the Secure Element through the pairing key. On receipt of a new AR value, the Secure Element marks any previously added cards as deleted.

Using the pairing key and its copy of the current AR value, the Secure Element verifies the authorization received from the Secure Enclave before retrieving encrypted payment data from a payment applet for a contactless payment. This process also applies when retrieving encrypted payment data from a payment applet for e-Commerce transactions.

2.4.2.3. Using a payment cryptogram for dynamic security

Payment transactions originating from the payment applets include a payment cryptogram along with a Device Account Number. This cryptogram, a one-time code, is computed using a transaction counter and a key. The transaction counter is incremented for each new transaction. The key is provisioned in the payment applet during personalization and is known by the payment network and/or the card issuer. Depending on the payment scheme, other data may also be used in the calculation, including an Apple Pay server nonce.

These security codes are provided to the payment network and to the card issuer, which allows the issuer to verify each transaction. The length of these security codes may vary based on the type of transaction.

2.4.3. Paying with cards using Apple Pay

2.4.3.1. Paying with cards at websites

Apple Pay can be used to make payments at websites on iPhone, iPad, Apple Watch and Mac computers with Touch ID. Apple Pay transactions can also start on a Mac and be completed on an Apple Pay–enabled iPhone or Apple Watch using the same iCloud account.

Apple Pay on the web requires that all participating websites register with Apple. After the domain is registered, domain name validation is performed only after Apple issues a TLS client certificate. Websites supporting Apple Pay are required to serve their content over HTTPS. For each payment transaction, websites need to obtain a secure and unique merchant session with an Apple server using the Apple-issued TLS client certificate. Merchant session data is signed by Apple. After a merchant session signature is verified, a website may query whether the user has an Apple Pay–capable device and whether they have a credit, debit or prepaid card activated on the device. No other details are shared.

If the user doesn't want to share this information, they can disable Apple Pay queries in Safari privacy settings on iPhone, iPad and Mac devices.

After a merchant session is validated, all security and privacy measures are the same as when a user pays within an app.

If the user is transmitting payment-related information from a Mac to an iPhone or Apple Watch, Apple Pay Handoff uses the end-to-end encrypted Apple Identity Service (IDS) protocol to transmit payment-related information between the user's Mac and the authorizing device. The IDS client on the Mac uses the user's device keys to perform encryption so no other device can decrypt this information and the keys aren't available to Apple. Device discovery for Apple Pay Handoff contains the type and unique identifier of the user's credit cards along with some metadata. The device-specific account number of the user's card isn't shared and it continues to remain stored securely on the user's iPhone or Apple Watch. Apple also securely transfers the user's recently used contact, shipping and billing addresses over iCloud Keychain.

After the user authorizes payment using Touch ID, Face ID, a passcode or double-clicking the side button on Apple Watch, a payment token uniquely encrypted to each website's merchant certificate is securely transmitted from the user's iPhone or Apple Watch to their Mac and then delivered to the merchant's website.

Only devices in proximity to each other may request and complete payment. Proximity is determined through Bluetooth Low Energy (BLE) advertisements.

The Apple Pay Handoff feature is not in the scope of this Security Target. After the Handoff process, the transaction validation is ensured by the user's iPhone or Apple Watch. The transaction validation is covered by the Security Target of the iPhone or the Apple Watch.

2.4.4. Rendering cards unusable with Apple Pay

Credit, debit, and prepaid cards added to the Secure Element can only be used if the Secure Element is presented with authorization using the same pairing key and AR value from when the card was added. This allows macOS to instruct the Secure Enclave to render cards unusable by marking its copy of the AR as invalid under the following scenarios:

• When the password is disabled

• The user signs out of iCloud

- The user performs a disk erase

- The device is restored from Recovery mode

2.4.4.1. Suspending, removing, and erasing cards

Users can suspend Apple Pay on the device by placing their devices in Lost Mode using Find My. Users also have the ability to remove and erase their cards from Apple Pay using Find My, iCloud.com, or directly on their devices using the Wallet app. The ability to make payments using cards on the device will be suspended or removed from Apple Pay by the card issuer or respective payment network, even if the device is offline and not connected to a cellular or Wi-Fi network. Users can also call their card issuer to suspend or remove cards from Apple Pay.

Additionally, when a user erases the entire device - using disk erase, using Find My, or restoring their device - macOS will instruct the Secure Element to mark all cards as deleted. This has the effect of immediately changing the cards to an unusable state until the Apple Pay servers can be contacted to fully erase the cards from the Secure Element. Independently, the Secure Enclave marks the Authorization Random as invalid so that further payment authorizations for previously enrolled cards are not possible. When the device is online, it attempts to contact the Apple Pay servers to ensure that all cards in the Secure Element are erased.

2.4.5. Apple Card

2.4.5.1. Apple Card application in the Wallet app

Apple Card can be used with Apple Pay to make payments in stores, in apps and on the web. Apple Card is only available for qualifying applicants in the United States.

To apply for Apple Card, the user must be signed into their iCloud account on an Apple Pay– compatible iOS or iPadOS device and have two-factor authentication set up on the iCloud account. When the application is approved, Apple Card is available in the Wallet app or within Settings (or System Preferences)> Wallet & Apple Pay across any of the eligible devices the user has signed in with their Apple ID.

When a user applies for Apple Card, user identity information is securely verified by Apple's identity provider partners and then shared with Goldman Sachs Bank USA for the purposes of identity and credit evaluation.

Information such as the social security number or ID document image provided during the application is securely transmitted to Apple's identity provider partners and/or Goldman Sachs Bank USA encrypted with their respective keys. Apple cannot decrypt this data.

The income information provided during the application, and the bank account information used for bill payments, are securely transmitted to Goldman Sachs Bank USA encrypted with their key. The bank account information is saved in Keychain. Apple cannot decrypt this data.

When adding Apple Card to the Wallet app, the same information as when a user adds a credit or debit card may be shared with the Apple partner bank Goldman Sachs Bank USA and with Apple Payments Inc. This information is used only for troubleshooting, fraud prevention, and regulatory purposes.

A physical card can be ordered from Apple Card in the Wallet app. After the user receives the physical card, it's activated using the NFC tag present in the bifold envelope of the physical card. The tag is unique per card and can't be used to activate another user's card. Alternatively, the card can be manually activated in the Wallet settings. Additionally, the user can also choose to lock or unlock the physical card at any time from the Wallet app.

2.4.6. Unlock with Apple Watch

A MacBook running macOS High Sierra or later can be unlocked using a paired Apple Watch Series 3 or later. The user can also use their Apple Watch to approve other requests to enter their administrator password. This works anywhere the user needs to type their Mac password, such as when viewing pass-words in Safari preferences, unlocking a locked note, approving an app installation, or unlocking settings in System Preferences. The paired Apple Watch doesn't authenticate a user for use with Apple Pay.

2.5. TOE Use Cases

The TOE covers the following use cases:

Use Case	Description
UC.Device_Usage	Device usage The User can manage the device's authentication credentials, including enrolling new biometric templates, updating biometric templates, deleting biometric templates and changing the password.
UC.OS_Update	Device Software Update The User can perform an update of the software in the device to a new version. This use case requires that the user verifies the device's password. This use case ensures preservation of the User settings on the device: - No change to the User's authentication credentials (password or any biometrics) - No change to the User's data within the Secure Element unless specified by the data's issuer
UC.Apple_Pay_Install_Init	Apple Pay installation and initialization The User can provision a new card in Wallet app
UC.Apple_Pay_Usage	Apple Pay usage The User can perform Apple Pay transactions.
UC.End_Of_Service	Termination by User The User can end the Apple Pay mode of operation by performing a card removal in Wallet. The User can also end all current Apple Pay services by un-registering their iCloud account. Termination by Issuer The Issuer can perform a de-registration of an Apple Pay card that was provisioned on the User's device, following a card revocation or a user account termination.
UC.End_Of_Life	Termination of device The User can clear a device from all their settings and data by performing a disk full erase. The User could also end the life of their device by physically destroying it.

2.6.   TOE Architecture

The TOE platform includes some hardware elements of the device and all the components of the device Platform that are relevant to strong customer authentication (SCA) and Apple Pay: The Secure Enclave within the System on a Chip (SoC) of the device, and the Application Processor (AP) with the OS are the main components of this TOE. The Secure Element is outside of the TOE boundaries.

User authentication is managed by the Secure Enclave, which defines and enforces the user access policy to data and services. The AP allows the OS to process and the OS allows the Secure Enclave to get authentication material from the various sources (biometric sensors, and the console).

macOS also allows the Apple Pay services to operate: it holds the Wallet app and relays data between the Secure Enclave and the Secure Element.

The Secure Element is the secure component holding the Apple Pay secrets and processes the Apple Pay transactions.

This is highlighted in the following figure:
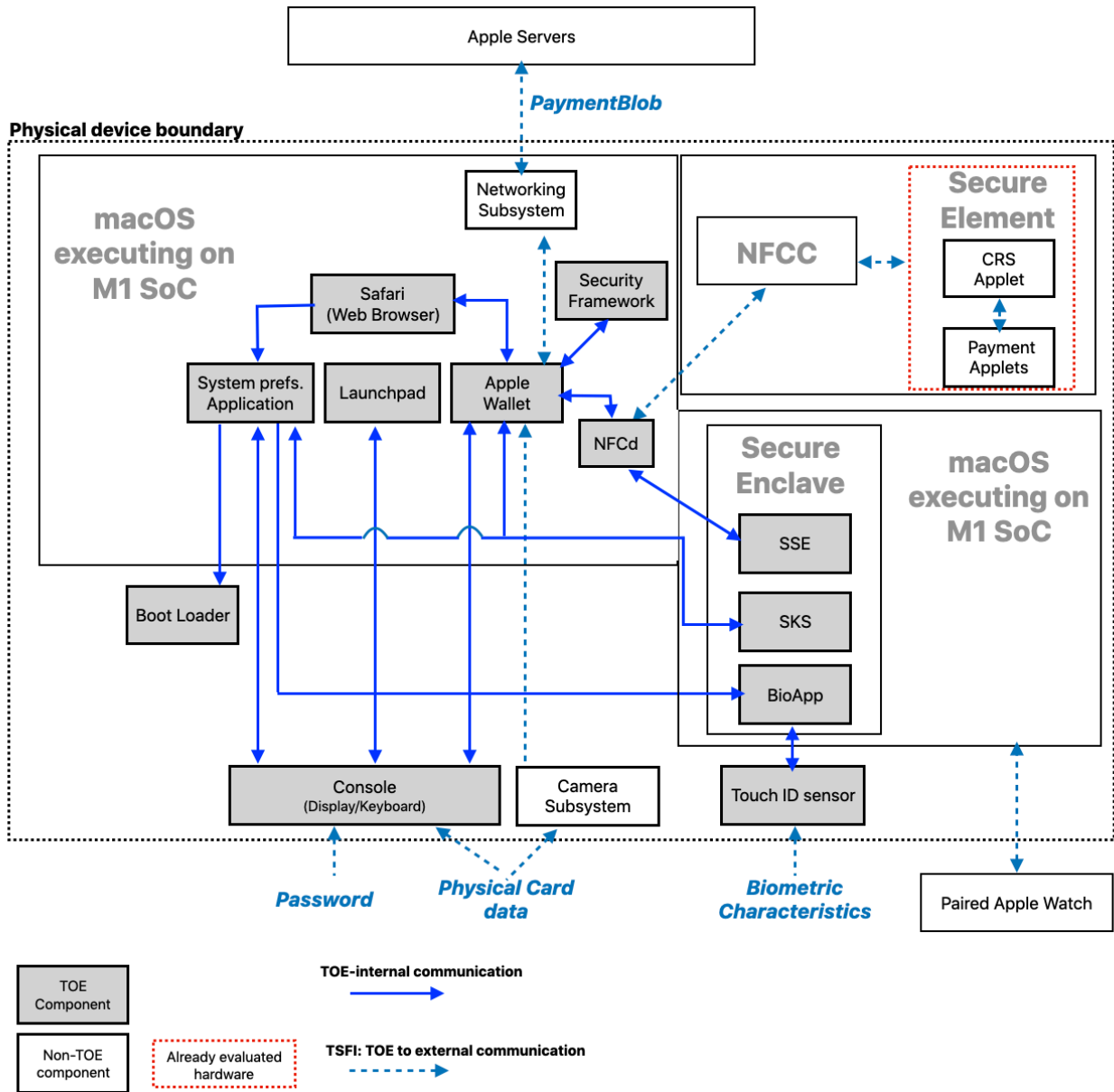
Figure 1: TOE Components and subsystems

2.6.1. TOE Environment

In this evaluation, the TOE environment includes the Secure Element (Hardware, Operating System, CRS applet and payment applets found in the Secure Element).

The Secure Element, included in the device, is already evaluated to Common Criteria. The Secure Element is contained in the same package as the NFCC. Payment Applets in the Secure Element process Apple Pay transactions. The NFCC is not connected to any antenna and is thus not usable for NFC transactions. It is used only for communications between the TOE and the Secure Element.

All other components and subsystems of macOS including user space and kernel software, hardware and subsystems included in the device including the camera, networking subsystems and Apple Watch are considered part of the TOE environment. The camera is used as an input device for card data. The networking subsystem provides connectivity to the Apple Servers responsible for managing Apple Pay transactions and to the Apple Watch.

The "Unlock with Apple Watch" feature relies on a secure pairing process and a secure unlock process.

2.6.2. Subsystems and Modules of the TOE

This section further breaks down the TOE components, providing more detail on the subsystems of the TOE and the modules that comprise them.

The subsystems of the TOE consist of:

- Secure Enclave: The software components of the TOE residing in the Secure Enclave. This subsystem includes several applications executing on the Secure Enclave operating system.

    - BioApp which provides functionality for processing biometric data and generating biometric templates.

    - The SKS (Secure Key Store) is a hardware cryptographic module. The module is embedded inside the Secure Enclave and packaged within the Application Processor.

    - The SSE (Secure Enclave-Secure Element) manages the pairing between the Secure Enclave and the Secure Element, allowing the Secure Element to process only genuine and authorized Apple Pay transactions. The SSE application maintains sensitive pairing material, allowing Secure Element and Secure Enclave to perform a mutual authentication before exchanging data.

- AW: The Wallet app subsystem handles the enrollment of applications and governs the payment operation.

- macOS components:

    - Launchpad: Provides the functionality for the macOS user interface

    - NFCd: This daemon facilitates the communication between Apple Wallet and the Secure Element

    - Safari browser: Provides the user interface to conduct payment transactions

    - Security Framework: Provides cryptographic support

    - System Preferences application

- Device components:

    - Touch ID sensor: This is the hardware component and associated drivers that allow fingerprints data to be captured and passed to BioApp for enrollment and matching

    - Boot-loader: This is the subsystem responsible for ensuring that the device boots using software with assured integrity

    - Console: Hardware and associated drivers that handles user input via the keyboard, and displays output via the screen

All other macOS components are outside of the TOE, including the Secure Element together with the NFCC hardware, and the XNU macOS kernel. The macOS subsystem components are individual applications.

2.6.3. TOE Lifecycle

The TOE lifecycle phases are as follows:

Lifecycle Phases			
Design	HW/FW design	SW design	Secure Element Applet design
Fabrication	HW fabrication	SW implementation	Applet development
Integration	MacBook Air integration *Assembly, Trust provisioning, FW integration, SW and applet loading*		
Issuance	Device delivery to User		
Initialization	User account creation Account setup: iCloud, Apple ID		
Enrollment/ Provisioning	User Authentication setup *Password setup, biometrics enrollment*		Apple Pay provisioning
Usage	Device Usage *Device unlock, User Authentication, macOS use, (optional) macOS update*		Apple Pay transaction
Termination	Physical destruction	macOS recovery mode	Apple Pay termination

The Design, Fabrication and Integration phases are entirely within the control of Apple Inc.

The Issuance phase is the physical delivery of the device to the User. This can be directly, in the case of an individual, or through a third party or an organization that is responsible for providing the device to the User.

Apple's model of the Initialization phase requires that the device is claimed by the User by associating it with the User's iCloud account and Apple ID. Before that point, Strong User Authentication and associated TSFs are not relevant, and Apple Pay is not accessible. Apple Pay services require that an iCloud account and user authentication credentials are setup (password and optionally biometrics).

The Usage phase describes the period when the Apple Pay service is activated and used by the associated User.

The Termination phase describes deactivation of the Apple Pay service to the User and may involve physical destruction of the device, complete erase and recovery of macOS (macOS recovery mode), or Apple Pay service termination by the User.

When the device is received, the model of the device should be verified to verify that the model number is one of those listed in Section 2.1. This can be accomplished using any of the following methods.

- Physically checking the underside of the device

- Once authenticated to the device the information is available to device users in the "About this Mac" overview, accessed from the Apple menu, or within the System Information app

3. Evaluation Assurance

3.1. Common Criteria Reference

This Security Target is based on the following Common Criteria™ (CC) publications:

Common Criteria	CC Version	Revision	Date
Part 1: Introduction and general model	V3.1	R5	April 2017
Part 2: Security functional requirements	V3.1	R5	April 2017
Part 3: Security assurance requirements	V3.1	R5	April 2017

3.2. CC Conformance claim

This Security Target is **conformant** to CC Part 2 and CC Part 3.

3.3. Protection Profile Conformance claim

This Security Target does not claim any conformance to an existing Protection Profile.

3.4. Assurance Level

The evaluation assurance level (EAL) for this work is EAL 2 augmented with ADV_FSP.3 and ALC_FLR.3:

Assurance Class	
ADV: **Development**	ADV_ARC.1 Security architecture description
	ADV_FSP.3 Functional specification with complete summary
	ADV_TDS.1 Basic design
AGD: **Guidance documents**	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: **Life-cycle support**	ALC_CMC.2 Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_FLR.3 Systematic flaw remediation
ASE: **Security Target evaluation**	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample

Assurance Class	
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

4. Security Problem Definition

4.1. Assets

The assets of this security target are:

Asset	Sen-sitiv-ity[*]	Type	Definition
D.User_Password	I, C	User	Password value setup by the User and used to wake up device after power loss, unlock the device and also allows the User to authorize payments (Apple Pay transactions).
D.User_Bio	I, C	User	Biometric data for the enrolled User's biometrics, used to unlock the device and to authorize payments (Apple Pay transactions).
D.Card_Data	I, C	User, TSF	Apple Pay card data, including credit/debit card number, User's name, expiration date, CVV, and transaction data (e.g. transaction history). Note: The card data is used as TSF data when the TOE sends it encrypted to the issuer, via Apple servers, in order to generate the Device Account Number stored in the SE.
D.User_Intent	I	User	State of the device resulting from a physical interaction of the User with the TOE, characterizing the intent of the User to perform payments (Apple Pay transactions)
D.Payment_Data	I	User	Apple Pay payment data being authorized by the User. For Apple Pay, critical elements are the amount (including the currency), the emitter, and the recipient (S.Merchant) which constitute the core of the Dynamic Linking.
D.OS	I	TSF	OS version currently installed on the device. This asset is not the OS code itself or the version number of that code. This asset is the set of elements that compose an OS version as prepared for that device, including elements verifiable by that device during boot (for instance the sepOS), and by the User through an identified version number.
D.User_Configuration	I	TSF	User configuration is a representation of the user personal configuration including Apple Pay, and Touch ID settings.
D.SEP_Configuration	I	TSF	Secure Enclave configuration is a representation of the state of the Secure Enclave in the device. It comprises (but is not limited to) the sepOS version and the state of the user authentication functions (password, enrolled biometrics, authenticated credentials, retry counters, and authentication based access control settings).
D.SEP_SE	I, C	TSF	Secret data that allows secure communication between the Secure Enclave and the SE during processing of an Apple Pay transaction.
D.SEP_Bio	I, C	TSF	Secret data that allows secure communication between the Secure Enclave and the biometric sensor during processing of biometric authentication.

Asset	Sen-sitiv-ity*	Type	Definition
D.SEP_Watch	I, C	TSF	Secret data that allows secure communication between the Secure Enclave on the TOE and the Secure Enclave on the Paired Watch, in the case the Paired Watch is used to unlock the TOE. The pairing secret allows the TOE and the Paired Watch to exchange sensitive data like the unlock secret.
D.Unlock_Secret	I,C	TSF	Secret data that is shared by the Secure Enclave on the TOE with the Paired Watch Secure Enclave to unlock the TOE. It is released by the Paired Watch to facilitate the TOE unlock. This asset is created by the TOE when the Unlock with Apple Watch feature is enabled by the User, and does not exist if there is no Paired Watch or if this feature is disabled.

*: I = Integrity, C = Confidentiality

4.2. Subjects

The subjects of this Security Target are:

Subject	Definition
S.User	User of Apple Pay on the device, able to: - Authenticate on the device (biometrics or password) - Manage authentication credentials (biometrics or password) - Manage device configuration (OS version, iCloud account) - Provision/enroll cards - Authorize Apple Pay transactions by providing consent for the transaction to proceed (biometrics/password and user intent) - Cancel the Apple Pay service
S.Apple_Servers	Apple servers in charge of: - Management of S.User iCloud account - Management of S.User provisioning/enrollment in Apple Pay - Management of OS releases, including the Wallet application - Device's interface for processing Apple Pay transactions (contact S.Issuer)
S.Issuers	The Issuer (or its service provider) is the third party in charge of: - Management of S.User data for Apple Pay services - Processing Apple Pay transactions
S.Merchant	The merchant is the third party accepting payment through an Apple Pay transaction.
S.SE	Certified Secure Element of the device including the TOE.
S.Apple_Watch	Apple Watch of S.User paired with the TOE (if applicable).

4.3. Assumptions

The assumptions for this Security Target are the following:

Assumption	Definition
A.DEVICE_AUTH	The User of the device owns all the Apple Pay activities that are performed on a device. All authentication credentials (biometrics or password) for the device that are enabled for use with Apple Pay are owned and protected by the User of the device.

Assumption	Definition
A.PERSO	The Apple Pay Issuers guarantee the correctness of the card data input in the device during provisioning/enrollment: Data shall uniquely identify a financial payment means and be linked to the account owned by the identified user.
A.WATCH_USER	If the User optionally pairs an Apple Watch to their MacBook Air, they own it and ensure the confidentiality of its authentication credentials.
A.NO_EXTERNAL_KEY-BOARD	The TOE is not paired with an external keyboard.
A.NO_EVIL_ROOT_USER	All Users of the device with root privileges are non-hostile.

4.4. Threat Agents

The threat agents of this TOE are the following:

Threat Agent	Definition
S.Attacker	A threat agent trying to interact with the Apple Pay system fraudulently, trying to modify the configuration and data of genuine Users' devices or forging data on their own device.

4.5. Threats

The threats to assets of this TOE are the following:

Threat	Name	Definition	Assets
T.COR-RUPT	Corrupted Transaction or Transfer	An attacker attempts to corrupt an Apple Pay transaction. To gain from the attack, the attacker could be the emitter of the transaction or transfer, and attempt to reduce what it intends to pay (debit amount from attacker's account) from what it was supposed to pay (credit amount request or transfer amount agreed between attacker and victim). The attacker could also attempt to corrupt a payment when it is the recipient, and increase what it supposed to receive (credit transaction amount) from what it was supposed to receive (debit amount agreed). The attacker could also attempt to modify the recipient of a credit transaction by changing the payee in an Apple Pay transaction.	D.Payment_Data D.User_Configuration D.OS
T.PHYSI-CAL	Physical	The loss or theft of the device may give rise to loss of confidentiality of User data including credentials and TSF data. These physical access threats may involve attacks which attempt to access the device through external hardware ports, through its user interface, and also through direct and possibly destructive access to its storage media. The goal of such attacks is to access Apple Pay from a lost or stolen device which is not expected to return to its User. *Note: Defending against device re-use after physical compromise is out of scope.*	D.User_Password D.User_Bio D.Card_Data D.Unlock_Secret* D.SEP_Watch*
T.RE-COVER	Card Recovery	An attacker attempts to recover Apple Pay card data from an erased or blocked device and use it to perform a financial transaction or transfer. The attacker will target potential breaches in the process of Apple Pay cancellation, Apple Pay card revocation or macOS recovery mode.	D.User_Configuration D.Card_Data D.OS
T.REPLAY	Replay	An attacker attempts to replay an Apple Pay transaction.	D.Payment_Data

Threat	Name	Definition	Assets
T.SILENT	Silent Transaction	An attacker attempts to modify the behavior of the device, in order to perform silent Apple Pay transactions for some benefit. The attacker would have to perform the attack without knowledge of the device's rightful owner who will be the victim of the attack.	D.User_Intent D.User_Configuration D.SEP_Configuration D.SEP_SE D.OS
T.SKIM-MING	Authentication Bypass	An attacker attempts to perform a payment with Apple Pay, bypassing the required authentication step (biometrics data verification or password verification).	D.User_Intent D.Payment_Data D.SEP_Configuration D.User_Configuration D.OS
T.USURP	Card Ownership Usurpation	An attacker could attempt to authenticate on a device, with a goal of using any provisioned cards on that device. The attacker could focus on the card data during Apple Pay provisioning.	D.User_Bio D.User_Password D.User_Configuration D.SEP_Bio D.Card_Data D.OS D.Unlock_Secret* D.SEP_Watch

* Partial threat: By unlocking the device, the attacker could have a higher chance to access Apple Pay data.

**Partial threat: the device can be unlocked using the paired watch but it does not allow the User to use provisioned cards.

Assets & Threats mapping table:

Asset - Property - I = Integrity C = Confidentiality		T. C O R R U P T	T. P H Y S I C A L	T. R E C O V E R	T. R E P L A Y	T. S K I M M I N G	T. S I L E N T	T. U S U R P
D.Unlock_Secret	I,C		X					X
D.User_Bio	I,C		X					X
D.User_Password	I,C		X					X
D.User_Intent	I					X	X	
D.Payment_Data	I	X			X	X		
D.Card_Data	I,C		X	X				X
D.OS	I	X		X		X	X	X
D.User_Configuration	I	X		X		X	X	X
D.SEP_Configuration	I					X	X	
D.SEP_Watch	I,C		X					X
D.SEP_SE	I,C						X	
D.SEP_Bio	I,C							X

4.6. Organizational Security Policies

The organizations associated with the Apple Pay service shall comply with the following Organizational Security Policies as security rules, procedures, practices, or guidelines imposed by an organization upon its operations:

OSP	Definition
P.UPDATE	Apple ensures that only an authenticated user can update their device's operating system to a newly released macOS. Apple also informs the Apple Pay Issuers and PNOs of new applicable features of new releases.
P.DYN_LINK	Apple maintains the enforcement of Dynamic Linking for e-Commerce payments using Apple Pay cards, on device side and server side. This Organizational Security Policy guarantees that Apple preserves the following properties from design to feature release for Apple Pay e-Commerce payments: (a) The payer is made aware of the amount of the payment transaction and of the payee (b) The authentication code generated is specific to the amount of the payment transaction and the payee agreed to by the payer when initiating the transaction (c) The authentication code accepted by the payment service provider corresponds to the original specific amount of the payment transaction and to the identity of the payee agreed to by the payer (d) any change to the amount or the payee results in the invalidation of the authentication code.
P.WATCH	The User can enable auto unlock with Apple Watch. The User can also use their Apple Watch to approve other requests (related to the security functions of the TOE) to enter their administrator password. These requests do not include user authentication for Apple Pay transactions.

5. Security Objectives

5.1. Security Objectives for the TOE

Security Objectives	Definition

OT.User_auth	The TOE enforces the following authentication policy:
	• Password only:
	    ○ Add, update or delete biometrics
	    ○ Update password
	    ○ Modify authentication policy
	    ○ Update the OS to a new version signed by Apple
	• Password or biometric (if enrolled)
	    ○ Unlock of the device
	    ○ Payments confirmation
	• Paired Apple Watch (optional):
	    ○ Unlock of the device
	    ○ Approve other requests (related to the security functions of the TOE) to enter their administrator password. These requests do not include user authentication for Apple Pay transactions.

OT.Card_Data	The TOE enforces that sensitive card data: • Is encrypted before being sent to the Apple servers • Is not accessible after being sent to the Apple Server
OT.Password_Delete	The TOE enforces that removing the password: • Disables biometric authentication • Disables Apple Pay
OT.Card_Delete	The TOE securely triggers the delete of each individual Apple Pay card when: • A card is removed from Wallet • The TOE handles the revocation of a card by its Issuer • The use of Apple Pay is canceled (from iCloud, the System Preferences or password removal) • The iCloud account is no longer associated with the device When a card is removed, the TOE also instructs the Secure Element to mark it as deleted.
OT.Auth_SE	The TOE provides the Secure Element with password/biometric user authentication feature for Apple Pay payment approval.
OT.Payment	For e-commerce transactions, the TOE enforces that transaction details are displayed to the User (including the card to be used from the Wallet, the amount, and the payee) before the User shows their intent to pay and authenticates for payment validation. The TOE ensures that these details cannot be corrupted between the payment validation and the moment when details are sent to the SE. The TOE also requests explicit intent from the User for Apple Pay transactions.
OT.Bio_Delete	TOE Biometrics delete is a secure erase of the enrolled biometric data.
OT.Disk_Erase	TOE securely deletes all User data when the User sets the OS to recovery mode or performs a device erase locally or remotely from iCloud. This also initiates the disabling of Apple Pay and the destruction of Apple Pay cards.

Security Objectives	Definition
OT.Anti_Replay	The TOE ensures that each payment processed by an Apple Pay card holds the unique identifier.
OT.OS_Update	TOE enforces security measures ensuring preservation of User data when an OS update is installed in the TOE. This objective protects the user authentication credentials (biometrics and password), the Apple Pay card data, and more.
OT.Watch	The TOE allows the user to do the following operations using their Apple Watch through a secure channel: • unlock the TOE • approve other requests (related to the security functions of the TOE) to enter their administrator password. These requests do not include user authentication for Apple Pay transactions.

5.2. Security Objectives for the environment

Environment Security Objectives	Definition
OE.Card_Data	The Issuer is responsible for using the appropriate security measures to protect the confidentiality and the integrity of the sensitive card data and guaranteeing the authenticity of the card during enrolment. The Secure Element is responsible for securing the card validation exchanges with the Issuer's TSM and for ensuring confidentiality and integrity of each card's sensitive data during storage and use.
OE.Perso	The Issuer is responsible for verifying that the User is authorized to perform a transaction on the account of the card used as a reference, before allowing the card personalization. The Issuer also ensures the robustness of the personalization data, to prevent attacks like forgery, counterfeit or corruption.
OE.Card_Delete	The Issuers of all payment cards provisioned on a device are informed when the User removes a card from that device, removes that device from the iCloud account or performs a disk erase. The Issuers ensure these cards are removed from the User's account (i.e. the unlinking process of the DPAN from the FPAN, which is done by the Issuer or the corresponding TSP). The Secure Element is responsible for securely deleting the stored sensitive data (private/secret).
OE.Anti_Replay	The Apple Pay server verifies that each payment (e-Commerce Apple Pay transaction) is not replayed. The payment is invalidated if this verification fails.
OE.Transaction_Verification	For Apple Pay, the cryptogram released by the Secure Element for an Apple Pay transaction is verified by the Issuer (or its service provider). The cryptogram validation result allows the Issuer to approve or reject the transaction. The payment is invalidated if this verification fails.
OE.Dynamic_Linking	For eCommerce transactions, the Apple Pay server preserves and Issuer verifies the cryptographic based dynamic linking of the transaction data (including amount and payee). The payment is invalidated if this verification fails.
OE.Statement	The payment card Issuers ensure that the statement associated to the card (list of transactions) is fully accurate and includes, at a minimum, not restricted to, the amount and recipient of each transaction.

Environment Security Objectives	Definition
OE.Genuine_Wallet	The Wallet application is provided and signed by Apple.
OE.Watch	The S.Apple_Watch is responsible for ensuring the confidentiality of the unlock secret provided by the MacBook Air during all its lifetime: from in reception at enabling of the "Unlock with Apple Watch" feature, during its storage, during its release for unlocking the MacBook Air, and when it is deleted when the feature is disabled. The S.Apple_Watch is responsible for ensuring that it is protected by a password and the wrist detection feature is turned on in order to enable the feature "unlock Mac with Apple Watch".
OE.User	The S.User is responsible for ensuring that: • Other users of the device with root access o Are trusted and competent to prevent inadvertent malware installation • They own all the Apple Pay activities that are performed on a device • The password is robust and protected • Only their own biometrics credentials are enrolled (they do not enroll biometrics of someone else) • The TOE is not paired with an external keyboard • Only their own Apple Watch is paired with the TOE and the paired Apple Watch is protected. This includes abiding by the watchOS Software License Agreement and protecting the confidentiality of the paired Apple Watch's passcode

5.3. Rationale of the Security objectives for the security problem definition

The following table details the rationale for each element of the security problem definition. For all the objectives for the TOE, OT.OS_Update also ensures the authentication configuration is preserved during the OS update.

Element	Rationale
A.DEVICE_AUTH	OE.User and OE.Watch cover A.DEVICE_AUTH ensuring the user owns and protects all the authentication credentials.
A.PERSO	OE.Perso covers A.PERSO ensuring the provisioning process verifies the ownership of the provisioned cards.
A.WATCH_USER	OE.User and OE.Watch covers A.WATCH_USER ensuring the User of the device owns and protects their Apple Watch and related credentials.
A.NO_EXTERNAL_KEYBOARD	OE.User ensures the user of the TOE does not pair the TOE with an external keyboard.
A.NO_EVIL_ROOT_USER	OE.User ensures other users of the device with root access are trusted and competent to prevent inadvertent malware installation.

Element	Rationale
T.SKIMMING	OT.User_Auth, OT.Auth_SE and OT.Payment ensure that an Apple Pay transaction is always authenticated and authorized by the User. OT.User_Auth and OE.Genuine_Wallet ensure that the Apple release of a new OS or Apple Wallet application implementing the authentication functions and payment functions are controlled, and that it preserves the confidentiality and integrity of the authentication and payment data. OT.Password_Delete ensures that if the password is turned off, the Apple Pay and biometric authentication are not available anymore.
T.USURP	OT.Card_Data OT.OS_Update and OE.Card_Data ensure that the card data is kept confidential and not altered from an attacker during storage and use in the Secure Element. OT.User_Auth, OT.Auth_SE and OT.Payment prevent the attacker from attempting to perform Apple Pay transactions, enforcing user authentication with set credentials (which are not known or owned by the attacker according to OE_User), and preventing the attacker from replacing a User's password or biometrics with their own. OT.User_Auth and OE.Genuine_Wallet additionally ensure that the installation of a new Apple released OS or Apple Wallet application preserves the confidentiality and integrity of the payment data. OT.Password_Delete ensures that if password login is disabled, the Apple Pay and biometric authentication are not available anymore.
T.RECOVER	OT.Bio_Delete ensures that a removed password or biometric credential cannot be recovered. OT.Card_Data, OT.OS_Update and OE.Card_Data ensure that the confidential card data is only stored by the Secure Element which protects them from disclosure. OT.Disk_Erase covers the physical erase of the content and settings of the device where the TOE will trigger secure erase all provisioned card data. OE.Card_Delete ensures that the Issuers of provisioned cards are securely removing the deleted cards from the User's account so that no transaction can further proceed.
T.REPLAY	OE.Anti_Replay ensures that each Apple Pay transaction is not replayed. OE.Anti_Replay and OT.Anti_Replay ensures that each Apple Pay transaction cannot be replayed.
T.CORRUPT	OT.Payment enforces the dynamic linking of the Apple Pay transaction data, ensuring that the critical content (such as emitter, recipient, amount) cannot be changed after the transaction was processed using a provisioned card. OE.Dynamic_Linking ensures that the Apple Pay server is verifying the integrity of the Apple Pay transaction data Dynamic Linking. OE.Statement provides an additional verification point for the account holder as the Issuer ensures that all processed Apple Pay transactions appear on the statement of the account associated to the card. OT.User_Auth and OE_Genuine_Wallet ensure that the Apple release of a new OS or Apple Wallet application implementing the payment functions are controlled.

Element	Rationale
T.SILENT	OT.User_Auth and OT.Payment ensure that an Apple Pay transaction is always authorized by the User. OE.Statement ensures that the card Issuers provide account holder verification material (in the form of transaction statements) allowing them to identify any fraudulent activity on their account. OT.User_Auth and OE.Genuine_Wallet ensure that the Apple release of a new OS or Apple Wallet application implementing the authentication functions and payment functions is controlled. OT.Password_Delete ensures that if password is turned off, the Apple Pay and biometric authentication are not available anymore.
T.PHYSICAL	The User credentials maintained by the TOE are secured by OT.User_Auth enforcing the secure verification process of biometrics or password. The TOE lifecycle is secure through the management of the device within the Apple iCloud environment where the User is able to remove the device from its account (OT.Card_Delete) and erase the disk content (OT.Disk_Erase). This binding ensures that the User's critical data is safe in case device is lost or stolen. OT.Card_Data, OT.OS_Update and OE.Card_Data ensure that the provisioned card data is kept confidential in the Secure Element and cannot be extracted.
P.UPDATE	OT.User_Auth and OE.Genuine_Wallet ensure that the Apple release of a new OS or Apple Wallet application implementing the authentication functions and payment functions are controlled, and that it preserves the confidentiality and integrity of the authentication and payment data.
P.DYN_LINK **(Dynamic Linking)**	P.DYN_LINK is covered by OT.Payment, OT.User_Auth, OT.Anti_Replay, OE.Anti_Replay, OE.Transaction_Verification, and OE.Dynamic_Linking, which all participate in the enforcement of the Dynamic Linking requirements on e-Commerce payments. The mapping is as follows: (a) OT.Payment ensures that there is a step, part of the user intent confirmation phase, when the User (payer) is made aware of the amount of the payment transaction and payee (b) OT.User_Auth ensures that the user authentication was performed to ensure agreement by the payer to authorize the Apple Pay transaction data (specific to the payer, amount and payee), OE.Anti_Replay ensures that each payment is uniquely identified, and OT.Payment ensures that the payment data is integrity protected (c) OE.Anti_Replay, OE.Transaction_Verification and OE.Dynamic_Linking ensure that the received Apple Pay transaction data correspond to what was agreed to by the payer: The unique identifier to prevent replay is verified, the cryptogram for the data is verified, and the dynamic linking of the user authentication and the payment data integrity is verified (d) OE.Anti_Replay, OE_Transaction_Verification and OE.Dynamic_Linking ensure that any change to the amount or the payee results in the invalidation of the payment and its unique identifier so no replay is attempted.
P.WATCH	P.WATCH is covered by OT_User_Auth and OT.Watch, which allow the Apple Watch dedicated features.

Security Objectives mapping table:

	T.CORRUPT	T.PHYSICAL	T.RECOVER	T.REPLAY	T.SILENT	T.SKIMMING	T.USURP	P.DYN_LINK	P.UPDATE	P.WATCH	A.PERSO	A.DEVICE_AUTH	A.WATCH_USER	A.NO_EXTERNAL_KEYBOARD	A.NO_EVIL_ROOT_USER
OT.Anti_Replay				x				x							
OT.Card_Data		x	x				x								
OT.Payment	x				x	x	x	x							
OT.Card_Delete		x													
OT.OS_Update		x	x				x								
OT.Auth_SE						x	x								
OT.User_Auth	x	x			x	x	x	x	x	x					
OT.Watch										x					
OT.Password_Delete						x									
OT.Bio_Delete		x													
OT.Disk_Erase		x	x												
OE_Anti_Replay				x				x							
OE.Card_Data		x	x				x								
OE.Perso											x				
OE.Dynamic_Linking	x							x							
OE.Statement	x			x											
OE.Transaction_Verification								x							
OE.Watch												x	x	x	
OE.User							x					x	x	x	x
OE.Genuine_Wallet	x			x			x		x						
OE.Card_Delete			x												

6. **Security Functional Requirements**

6.1. SFR supporting definitions

6.1.1. Security Functional Policies (SFP)

Access Control SFPs are given in the table below:

Authentication_SFP	Authentication policy enforcing authentication, re-authentication and authorization rules as defined by OT.User_Auth. This SFP includes information flows between the Secure Enclave and the biometric sensor (for biometric authentication) and between the TOE and S.Apple_Watch.
Payment_SFP	Security policy enforcing that processing a payment requires the User to confirm the intent to pay and being re-authenticated (password, or, if configured, biometrics) to allow processing of the related data and its exportation.
Card_Perso_SFP	Security policy enforcing that: - Importing D.Card_Data is only allowed if: a password is configured - Confidential parts of the card data are protected before being sent to the Apple Servers - Confidential parts of the card data are not imported in the Wallet

6.1.2. Subjects and Objects

Objects are the Assets identified in the Section 4.1.
Subjects are listed in Sections 4.2 and 4.4

6.1.3. Security Attributes

Security Attribute		
Card Data Confidential parts	Parts of the Card Data that should stay confidential and not been stored in the Wallet	- Secret parts of the card number - CVV - …
User Authorization	Part of D.Payment_Data specifying the explicit authorization of the S.User	- "yes" - "no"
BioAuth Unlock	Part of D.User_Configuration, authorization to use biometric authentication for unlocking a locked device, "selected" or "not selected" by the User.	- "selected" - "not selected" (default)
BioAuth AP	Part of D.User_Configuration, authorization to use biometric authentication for Apple Pay operations, "selected" or "not selected" by the User.	- "selected" (default) - "not selected"
Watch Unlock	Authorization to use the paired Apple Watch to unlock the TOE.	- "enabled" - "disabled" (default)
Apple OS public key	The public key used to check the authenticity of a new version of D.OS.	- Part of installed D.OS
OS_signature	Part of the OS file that is checked by the TOE before updating D.OS	- Part of the update file

Security Attribute		
Password_off	Part of D.SEP_Configuration, configuration of the OS allowing to use the device without any authentication. When enabled, the TSF should disable biometric authentication and Apple Pay.	- "disabled" (default) - "enable"

6.1.4. Writing conventions for the SFR operations

Iterations are identified by a slash character "/" followed by the name of the iteration.
Assignments and selections are done with *italicized text*.
Refinements are identified with the prefix "Refinement:".

6.2.   Identification and authentication

6.2.1. User authentication

FIA_UID.2 User identification before any action

FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.2 User authentication before any action

FIA_UAU.2.1	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Note: this gives S.User the role of "Authenticated User". According to this requirement, S.User is authenticated by the TSF before performing any of the operations listed in the following requirements.

FIA_UAU.5 Multiple authentication mechanisms

FIA_UAU.5.1	The TSF shall provide *password authentication, biometric authentication (fingerprint), Watch unlock (paired Apple Watch)* to support user authentication.
FIA_UAU.5.2	The TSF shall authenticate any user's claimed identity according to the: • *Password authentication as default authentication* • *Biometric authentication for:* - *Unlock if selected in the System Preferences (BioAuth Unlock = selected)* - *Transaction authorization if selected in the System Preferences (BioAuth AP = selected)* • *Watch unlock (if Watch Unlock = enabled) for:* - *unlock the TOE* • *Rules defined in FIA_UAU.6 Re-authenticating and FIA_AFL.1 Authentication failure handling.*

Note: The TOE allows other authorizations not related to Apple Pay to be performed from optional paired Apple Watch.

FIA_AFL.1/Biometric Authentication failure handling

FIA_AFL.1.1	The TSF shall detect when 5 *(five)* unsuccessful authentication attempts occur related to b*iometric validation*.
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall *require password validation, blocking further biometric validation attempts*.

FIA_AFL.1/Recovery Authentication failure handling

FIA_AFL.1.1 /Recovery	The TSF shall detect when 10 *(ten)* unsuccessful authentication attempts occur related to *first password validation after boot or reboot*.
FIA_AFL.1.2 /Recovery	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall *force booting into recoveryOS*

FIA_AFL.1/Delay Authentication failure handling

FIA_AFL.1.1 /Delay	The TSF shall detect when 5 *(five)* unsuccessful authentication attempts occur related to *first password validation after boot or reboot*.
FIA_AFL.1.2 /Delay	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall *start delaying further password validation attempts and require password validation*.

FIA_UAU.6 Re-authenticating

FIA_UAU.6.1	The TSF shall re-authenticate the user under the conditions *that the user requests:* - *OS update (change of D.OS)* - *Erasing the device* - *Unlocking the Security & Privacy pane in System Preferences (allowing to change "BioAuth Unlock" and "BioAuth AP", password and biometric patterns)* - *Unlocking the Users & Groups pane in System Preferences* - *Transaction validation (export of D.Transaction_Data), the re-authentication should be done during the 60 second after the transaction validation request.*

6.2.2. Data Authentication

FDP_DAU.1 Basic Data Authentication

FDP_DAU.1.1	The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of *D.Payment_Data (including S.Merchant and S.User data)*.
FDP_DAU.1.2	The TSF shall provide *S.SE* with the ability to verify evidence of the validity of the indicated information.

6.2.3. User attribute definition

FIA_ATD.1 User attribute definition

FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: *D.User_Password, D.User_Bio, D.Card_Data, BioAuth Unlock, BioAuth AP, Watch Unlock*.
Refinement: The update of D.OS shall not modify these user attributes.	

6.2.4. Specification of secrets

FIA_SOS.2 TSF Generation of secrets

FIA_SOS.2.1	The TSF shall provide a mechanism to generate secrets that meet *FIPS 140-2 validated cryptographic module*.
FIA_SOS.2.2	The TSF shall be able to enforce the use of TSF generated secrets for *creation of the D.Unlock_Secret during the Unlock with Apple Watch Setup*.

6.3.   Access/Flow Control SFRs

6.3.1. Authentication _SFP

FDP_ACC.2/Authentication_SFP Complete access control

FDP_ACC.2.1/ Authentication_SFP	The TSF shall enforce the *Authentication_SFP* on:	
	Sub-jects:	*S.User, S.Apple_Watch*
	Objects:	*the TSF*
	and all operations among subjects and objects covered by the SFP.	
FDP_ACC.2.2/ Authentication_SFP	The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.	

FDP_ACF.1/Authentication_SFP Security attribute-based access control

FDP_ACF.1.1/ Authentication_SFP	The TSF shall enforce the *Authentication_SFP* to objects based on the following:	
	Subjects:	*S.User, S.Apple_Watch*
	Objects:	*the TSF*
	Security attributes:	*"Watch Unlock"*
FDP_ACF.1.2/ Authentication_SFP	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *S.User is authenticated according to FIA_UAU.5*	
FDP_ACF.1.3/ Authentication_SFP	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *If "Watch Unlock" set to "enabled", S.Apple_Watch gives S.User the capability to:* - *Unlock the TOE* - *Approve other requests (related to the security functions of the TOE) to enter their administrator password.*	
FDP_ACF.1.4/ Authentication_SFP	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *"Password_off" is enabled*.	

FDP_ITT.1 Basic internal transfer protection

FDP_ITT.1.1	The TSF shall enforce the *Authentication_SFP* to prevent the *modification and disclosure* of user data when it is transmitted between physically-separated parts of the TOE.

Note: This requirement concerns the protection of biometric data sent by the biometric sensor to the Secure Enclave. Protection against modification includes also protection against replay.

6.3.2. Payment_SFP

FDP_ETC.2/Transaction Export of user data with security attributes

FDP_ETC.2.1 /Transaction	The TSF shall enforce the *Payment_SFP* when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2 /Transaction	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3 /Transaction	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4 /Transaction	The TSF shall enforce the following rules when user data is exported from the TOE: - *exported Payment_SFP details (the card to be used from the Wallet, the amount and the payee in the case of e-commerce payments) are those displayed to S.User during re-authentication request (FIA_UAU.6 Re-authenticating)* - *D.Payment_Data includes a unique identifier (An Apple Pay server nonce).*

FDP_ACC.2/Payment_SFP Complete access control

FDP_ACC.2.1/ Payment_SFP	The TSF shall enforce the *Payment_SFP* on*:* <table><tr><td>Subjects:</td><td>S.User</td></tr><tr><td>Objects:</td><td>D.Payment_Data (including S.User and S.Merchant Data)</td></tr></table>*, and all operations among subjects and objects covered by the SFP.*
FDP_ACC.2.2/ Payment_SFP	The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

Note: the only possible operation on D.Payment_Data is to export it as a transaction order to the Secure Element.

FDP_ACF.1/Payment_SFP Security attribute based access control

FDP_ACF.1.1/ Payment_SFP	The TSF shall enforce the *Payment_SFP* to objects based on the following: <table><tr><td>Subjects:</td><td>S.User</td></tr><tr><td>Objects:</td><td>D.Payment_Data (including S.User and S.Merchant Data), the SE (through a trusted channel)</td></tr><tr><td>Security attributes:</td><td>"User Authorization", "BioAuth AP", "Password_off"</td></tr></table>
FDP_ACF.1.2/ Payment_SFP	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: - *Export of D.Payment_Data with "User Authorization" set to "yes" to the SE is allowed if:* - *S.User shows their intent to pay (using the gesture of activating the Touch ID sensor combined with successfully matching the user's fingerprint or double pressing of touch ID sensor when user doesn't have touch ID set up)* - *S.User had been successfully re-authenticated for transaction validation (FIA_UAU.6).*
FDP_ACF.1.3/ Payment_SFP	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*.
FDP_ACF.1.4/ Payment_SFP	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *"Password_off" is "enabled"*.

6.3.3. Card_Perso_SFP

FDP_ACC.2/Card_Perso_SFP Complete access control

FDP_ACC.2.1 /Card_Perso_SFP	The TSF shall enforce the *Card_Perso_SFP* on	
	Subjects:	*S.User, S.Apple_Servers*
	Objects:	*D.Card_Data*
	and all operations among subjects and objects covered by the SFP.	
FDP_ACC.2.2 /Card_Perso_SFP	The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.	

FDP_ACF.1/Card_Perso_SFP Security attribute based access control

FDP_ACF.1.1 /Card_Perso_SFP	The TSF shall enforce the *Card_Perso_SFP* to objects based on the following:	
	Subjects:	*S.User, S.Apple_Servers*
	Objects:	*D.Card_Data*
	Security attributes:	*Password_off*
FDP_ACF.1.2 /Card_Perso_SFP	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: - *S.User is connected to its iCloud account, and is authenticated on the TOE*.	
FDP_ACF.1.3 /Card_Perso_SFP	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*.	
FDP_ACF.1.4 /Card_Perso_SFP	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: "*Password_off*" is enabled.	

FDP_ETC.2/Card_Perso_SFP Export of user data with security attributes

FDP_ETC.2.1 /Card_Perso_SFP	The TSF shall enforce the *Card_Perso_SFP* when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2 /Card_Perso_SFP	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3 /Card_Perso_SFP	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4 /Card_Perso_SFP	The TSF shall enforce the following rules when user data is exported from the TOE: - *D.Card_Data is encrypted before being exported to S.Apple_Server* - *"Card Data Confidential parts" are not kept on the TOE after being exported*.

FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1	The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product from unauthorised disclosure during transmission.

FDP_ITC.1/Card_Perso_SFP Import of user data without security attributes

FDP_ITC.1.1 /Card_Perso_SFP	The TSF shall enforce the *Card_Perso_SFP* when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.1.2 /Card_Perso_SFP	The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.
FDP_ITC.1.3 /Card_Perso_SFP	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *None*.

Note: security attributes of the Card Data are "Card Data Confidential parts" in section 6.1.3. These requirements specify that the confidential part of the cards data is used for card enrollment (FDP_ETC.2/Card_Perso_SFP) but are not stored on the TOE.

6.4.   Secure Enclave/Secure Element Trusted Channel

FTP_ITC.1/SE Inter-TSF trusted channel

FTP_ITC.1.1/SE	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2/SE	The TSF shall permit *the TSF* to initiate communication via the trusted channel.
FTP_ITC.1.3/SE	The TSF shall initiate communication via the trusted channel for *Payment initiation and transmission of D.Payment_Data*.

FDP_UCT.1/SE Basic data exchange confidentiality

FDP_UCT.1.1/SE	The TSF shall enforce the *Payment_SFP,* to *transmit* user data in a manner protected from unauthorised disclosure.

FDP_UIT.1/SE Data exchange integrity

FDP_UIT.1.1/SE	The TSF shall enforce the *Payment_SFP,* to *transmit and receive* user data in a manner protected from *modification, insertion and replay* errors.
FDP_UIT.1.2/SE	The TSF shall be able to determine on receipt of user data, whether *modification, insertion or replay* has occurred.

FPT_RPL.1/SE Replay detection

FPT_RPL.1.1/SE	The TSF shall detect replay for the following entities: *S.SE*.
FPT_RPL.1.2/SE	The TSF shall perform *reject data* when replay is detected.

6.5.   Secure Enclave/Apple Watch Trusted Channel

FTP_ITC.1/Watch Inter-TSF trusted channel

FTP_ITC.1.1 /Watch	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2 /Watch	The TSF shall permit *the TSF* to initiate communication via the trusted channel.
FTP_ITC.1.3 /Watch	The TSF shall initiate communication via the trusted channel for: *unlock the TOE*.

FDP_UCT.1/Watch Basic data exchange confidentiality

FDP_UCT.1.1 /Watch	The TSF shall enforce the *Authentication_SFP*, to *transmit and receive* user data in a manner protected from unauthorised disclosure.

FDP_UIT.1/Watch Data exchange integrity

FDP_UIT.1.1 /Watch	The TSF shall enforce the *Authentication_SFP* to *transmit and receive* user data in a manner protected from *modification, insertion and replay* errors.
FDP_UIT.1.2 /Watch	The TSF shall be able to determine on receipt of user data, whether *modification, insertion or replay* has occurred.

FPT_RPL.1/Watch Replay detection

FPT_RPL.1.1/Watch	The TSF shall detect replay for the following entities: *S.Apple_Watch*.
FPT_RPL.1.2/Watch	The TSF shall perform *reject data* when replay is detected.

6.6.   Local data protection

FPR_UNO.1 Unobservability

FPR_UNO.1.1	The TSF shall ensure that *S.Attacker* are unable to observe the operation *Password set, Password check, Password Update, Password removal, Biometrics set, Biometrics check, iometrics update, Biometrics delete, Apple Pay provisioning, Apple Watch Unlock activation*, on *D.User_Bio, D.User_Password, D.Card_Data, D.SEP_Watch, D.Unlock_Secret* by *S.User*.

FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the *deallocation of the resource from* the following objects: *D.User_Bio, D.User_Password, "Card Data Confidential parts" and iCloud Account (in D.User_Configuration)*.
Refinement: - The removal of the iCloud Account by S.User triggers the deallocation of all the Apple Pay data/configuration - The revocation of individual card by its issuer triggers the deallocation of the related card data - The procedure of erase on the device triggers the deallocation of all security attributes except the D.OS - The removal of "Card Data Confidential parts" is done by instructing the Secure Element to mark the card as deleted - The removal of the Password by S.User disables the actual Password; and triggers the deallocation of all the Apple Pay data	

FDP_SDI.1 Stored data integrity monitoring

FDP_SDI.1.1	The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors* on all objects, based on the following attributes: *D.User_Bio, D.Card_Data, D.OS*.

6.7.   TSF management

6.7.1. Roles and Management Functions

FMT_SMR.1 Security roles

FMT_SMR.1.1	The TSF shall maintain the roles *Authenticated User*.
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: *management of security attributes (see FMT_MSA.1).*

6.7.2. Management of security attributes

FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1	The TSF shall enforce the *Payment_SFP, Card_Perso_SFP* to provide	
	"BioAuth Unlock"	restrictive (disabled)
	"BioAuth AP"	permissive (selected)
	"Password_off"	restrictive (disabled)
	"Watch Unlock"	restrictive (disabled)
	default values for security attributes that are used to enforce the SFP.	
FMT_MSA.3.2	The TSF shall allow *nobody* to specify alternative initial values to override the default values when an object or information is created.	

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1	The TSF shall enforce the *Authentication_SFP, Card_Perso_SFP* to restrict the ability to *modify* the security attributes ", "BioAuth Unlock", "BioAuth AP", "Password_off" to "Authenticated User", "Watch Unlock".*
Refinement:	
–	When "Password_off" is enabled, the TSF removes the D.Card_Data according to FDP_RIP.1.

6.7.3. Management of TSF_Data

FMT_MTD.1 Management of TSF data

FMT_MTD.1.1	The TSF shall restrict the ability to *update* the *D.OS* to *"Authenticated user".*

FMT_MTD.3 Secure TSF data

FMT_MTD.3.1	The TSF shall ensure that only secure values are accepted for *D.OS*.
Refinement: secure value is defined by "OS_signature" is valid and signed by "Apple OS public key".	

6.8.   Security Requirements Rationale

6.8.1. Security Functional Requirements (SFR) Dependencies

TOE SFR	Required dependencies	Covered by
FIA_UAU.2	FIA_UID.1	FIA_UID.2
FIA_AFL.1(Biometric/Recovery/Delay)	FIA_UAU.1	FIA_UAU.2
FDP_ACC.2/Authentication_SFP	FDP_ACF.1	FDP_ACF.1/Authentication_SFP
FDP_ACF.1/Authentication_SFP	FDP_ACC.1 FMT_MSA.3	FDP_ACC.2/Authentication_SFP FMT_MSA.3
FDP_ACC.2/Payment_SFP	FDP_ACF.1	FDP_ACF.1/Payment_SFP
FDP_ACF.1/Payment_SFP	FDP_ACC.1 FMT_MSA.3	FDP_ACC.2/Payment_SFP FMT_MSA.3
FDP_ITT.1	FDP_ACC.1, or FDP_IFC.1	FDP_ACC.2/Authentication_SFP
FDP_ETC.2/Transaction	FDP_ACC.1, or FDP_IFC.1	FDP_ACC.2/Payment_SFP
FDP_ACC.2/Card_Perso_SFP	FDP_ACF.1	FDP_ACF.1/Card_Perso_SFP
FDP_ACF.1/Card_Perso_SFP	FDP_ACC.1 FMT_MSA.3	FDP_ACC.2/Card_Perso_SFP FMT_MSA.3
FDP_ITC.1/Card_Perso_SFP	FDP_ACC.1, or FDP_IFC.1 FMT_MSA.3	FDP_ACC.2/Card_Perso_SFP FMT_MSA.3
FDP_ETC.2/Card_Perso_SFP	FDP_ACC.1, or FDP_IFC.1	FDP_ACC.2/Card_Perso_SFP
FDP_UCT.1/SE	FDP_ACC.1, or FDP_IFC.1 FTP_ITC.1, or FTP_TRP.1	FDP_ACF.1/Payment_SFP, and FDP_ACF.1/Card_Perso_SFP FTP_ITC.1/SE
FDP_UIT.1/SE	FDP_ACC.1, or FDP_IFC.1 FTP_ITC.1, or FTP_TRP.1	FDP_ACF.1/Payment_SFP, and FDP_ACF.1/Card_Perso_SFP FTP_ITC.1/SE
FDP_UCT.1/Watch	FDP_ACC.1, or FDP_IFC.1 FTP_ITC.1, or FTP_TRP.1	FDP_ACF.1/Authentication_SFP FTP_ITC.1/Watch
FDP_UIT.1/Watch	FDP_ACC.1, or FDP_IFC.1 FTP_ITC.1, or FTP_TRP.1	FDP_ACF.1/Authentication_SFP FTP_ITC.1/Watch
FMT_SMR.1	FIA_UID.1	FIA_UID.2
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1	FMT_MSA.1, FMT_SMR.1
FMT_MSA.1	FDP_ACC.1, or FDP_IFC.1 FMT_SMR.1, SMT_SMF.1	FDP_ACC.2/Authentication_SFP, FDP_ACC.2/Authentication_SFP, and FDP_ACC.2/Card_Perso_SFP FMT_SMR.1, FMT_SMF.1
FMT_MTD.1	FMT_SMR.1, SMT_SMF.1	FMT_SMR.1, SMT_SMF.1
FMT_MTD.3	FMT_MTD.1	FMT_MTD.1

Requirements without dependency: FIA_UID.2, FIA_SOS.2, FIA_UAU.6, FIA_UAU.5, FDP_DAU.1, FIA_ATD.1, FPT_ITC.1, FTP_ITC.1/SE, FPT_RPL.1/SE, FTP_ITC.1/Watch, FPT_RPL.1/Watch, FPR_UNO.1, FDP_RIP.1, FMT_SMF.1 and FDP_SDI.1

6.8.2. Rationale SFR/Security Objectives for the TOE

Objective reference	Rationale
OT.User_Auth	FIA_UID.2 and FIA_UAU.2 enforce each User of the device is authenticated before being able to do any action in the user interface.
	FIA_UAU.5 specifies different authentication methods.
	FIA_UAU.6 enforces a re-authentication for the OS update, for changing the authentication configuration, including biometric data management and biometric authentication policy or for a transaction validation.
	FMT_SMR.1 and FMT_SMF.1 SF ensure that the TSF shall maintain the roles Authenticated User and be capable of performing the management functions.
	FIA_ATD.1, FMT_MSA.3, FMT_MSA.1 and FMT_MTD.1 specify the management of related information.
	FMT_MTD.3 specifies the signature verification on the OS update.
	FIA_AFL.1 (Biometric/Erase/Delay) specify the failure handling in case of wrong biometric or password authentication.
	FPR_UNO.1 ensures the non-observability of the password.
	FDP_ITT.1 also support this objective ensuring the biometric data is not modified or disclosed during internal transfer.
OT.Card_Data	The SFP Card_Perso_SFP and the associated SFRs (FDP_ACC.2/Card_Perso_SFP, FDP_ACF.1/Card_Perso_SFP, FDP_ITC.1/Card_Perso_SFP and FDP_ETC.2/Card_Perso_SFP, FPT_ITC.1) enforce card data stored in the wallet does not include sensitive card data as this one is only sent to the Secure Element.
	Also, FTP_ITC.1/SE, FDP_UCT.1/SE and FDP_UIT.1/SE enforce that all the data exchanged between the TSF and S.SE is protected (with their corresponding security property) by a trusted channel.
	FPR_UNO.1 ensures the non-observability of secret card data.
	FDP_SDI.1 ensures card data stored in containers controlled by the TSF are monitored for integrity errors.
OT.Password_Delete	FDP_ACC.2/Card_Perso_SFP and FDP_ACF.1.4/Card_Perso_SFP enforce the card enrollment is not possible if Password_off is enabled.
	FIA_UAU.6 enforces a re-authentication for changing the Password_off configuration.
	FPR_UNO.1 ensures the non-observability of the password.
OT.Card_Delete	FDP_RIP.1 ensures the confidential parts of the card data are securely removed by the Secure Element.
	Refinement of FMT_MSA.1.1 ensures the secure delete in case of password turning off.
	FPR_UNO.1 ensures the non-observability of sensitive card data.
OT.Auth_SE	FIA_UID.2, FIA_UAU.2, FIA_UAU.5 specify the base of the authentication feature.
	FIA_AFL.1/Biometric enhance the biometric security limiting the authentication attempts before requiring password authentication.
	Other FIA_AFL.1 protect the TSF and the user data against brute force attacks.
	FIA_UAU.6 specifies the re-authentication for some functions of the TOE.

Objective reference	Rationale
OT.Payment	FDP_ETC.2.4/Transaction ensures the transaction details are displayed before a transaction is validated by the User.
	FIA_UAU.6 ensures each transaction is validated by a re-authentication.
	FDP_DAU.1 provides evidences of the validity of Apple Pay Transaction Data, verifiable by the Issuer.
	FDP_ACC.2/Payment_SFP and FDP_ACF.2/Payment_SFP ensure user authorization is done before each transaction.
	FPT_RPL.1/SE ensures transaction replay detection.
OT.Bio_Delete	FDP_RIP.1 ensures biometric data delete is a secure erase.
OT.Disk_Erase	FDP_RIP.1 ensures all sensitive data is securely removed during a disk erase
OT.Anti_Replay	FDP_ETC.2.4/Transaction ensures a unique identifier provided by the Apple server is included in the transaction data, avoiding any replay attack.
OT.OS_Update	FIA_UAU.6.1 ensures S.User is re-authenticated before the OS update proceeds.
	The refinement of FIA_ATD.1.1 ensures that all the user data with impact on the TSF behavior are not modified during the OS update process.
	FDP_SDI.1 ensures D.OS stored in containers controlled by the TSF is monitored for integrity errors.
OT.Watch	FTP_ITC.1/Watch, FDP_UCT.1/Watch, FDP_UIT.1/Watch, FIA_SOS.2, FPR_UNO.1 and FPT_RPL.1/Watch enforce that all the data exchanged between the TSF and S.Apple_Watch is protected by a trusted channel regarding the confidentiality and the integrity.
	The protected operations are specified by FDP_ACC.2/Authentication_SFP and FDP_ACF.1/Authentication_SFP.

6.8.3. SAR Dependencies

TOE SAR	Dependencies
ADV_ARC.1	ADV_FSP.1, ADV_TDS.1
ADV_FSP.3	ADV_TDS.1
ADV_TDS.1	ADV_FSP.2
AGD_OPE.1	ADV_FSP.1
AGD_PRE.1	No dependencies
ALC_CMC.2	ALC_CMS.1
ALC_CMS.2	No dependencies
ALC_DEL.1	No dependencies
ALC_FLR.3	No dependencies
ASE_CCL.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.1
ASE_ECD.1	No dependencies
ASE_INT.1	No dependencies
ASE_OBJ.2	ASE_SPD.1
ASE_REQ.2	ASE_OBJ.2, ASE_ECD.1

TOE SAR	Dependencies
ASE_SPD.1	No dependencies
ASE_TSS.1	ASE_INT.1, ASE_REQ.1, ADV_FSP.1
ATE_COV.1	ADV_FSP.2, ATE_FUN.1
ATE_FUN.1	ATE_COV.1
ATE_IND.2	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1
AVA_VAN.2	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1

All the dependencies are covered.

6.8.4. SAR Rationale

For this evaluation, the SARs of EAL2 have been chosen as they provide assurance by a full security target and an analysis of the SFRs in that ST, using a functional and interface specification, guidance documentation and a basic description of the architecture of the TOE, to understand the security behavior. It was established that this level is appropriate for the model of attack of Apple Pay.

ADV_FSP.3 augmentation has been chosen to enhance the level of information provided related to SFR-enforcing TSFIs and provide a complete summary of the TOE.

ALC_FLR.3 augmentation has been chosen to guarantee the security of the security functions in the scope of this security target during the maintenance of the TOE.

7.   TOE Summary Specification

This section describes the security functions of the TOE covering the SFR of the previous chapter.

7.1.   SF User authentication and management

When using the TOE, before being able to use Touch ID as a biometric authentication method, the device must be set up so that a password is required to unlock it.

Touch ID is Apple's fingerprint authentication system that enables the secure access to a MacBook Air equipped with the Touch ID sensor. This technology is based on the Touch ID sensor and the Secure Enclave.

When Touch ID detects a successful match, the device unlocks without asking for the device password. Touch ID does not replace the password but provides easy access to the device within defined boundaries and time constraints. To use Touch ID, the device must be set up so that a password is required to unlock it.

The password can always be used instead of biometrics, and the password is required under the following circumstances:

- The device has just been turned on or restarted
- The User has not unlocked their device for more than 48 hours
- The User has not used their password to unlock their device for 156 hours (six and a half days), and the User has not used a biometric to unlock their device in 4 hours
- The device has received a remote lock command
- There were five unsuccessful biometric match attempts (though for usability, the device might offer entering a password instead of using biometrics after a smaller number of failures)

These features implement the requirements listed in §6.2 and in §6.3.1 for the authentication, in §6.6 for local data protection and in FMT_SMF.1 and FMT_MSA.1 for management.

Unlock with a paired Apple (requirements in §6.5 for secure communication with Apple Watch) Watch gives the user the capability to use their optional paired Apple Watch in order to:

- Unlock the TOE
- Approve other requests (related to the security functions of the TOE) to enter their administrator password. These requests do not include user authentication for Apple Pay transactions.

The following table summarizes the functions supporting User authentication.

Function	Description
Password authentication	Password_Setup: Setup of the device's password.
	Password_Update: Update of the device's password.
	Password_Verify: Authentication of the User using password.
	Password_Delete: Removal of the password
Biometric authentication	Bio_Enroll: Enrollment of biometrics credential (first, or additional).
	Bio_Update: Update of biometrics.
	Bio_Verify: Authentication of the User using biometrics.
	Bio_Delete: Delete an enrolled biometrics credential (one, or all).
	Unlock_With_Watch_Setup: Enable the feature « Unlock with Apple Watch ».

Unlock with Apple Watch	Unlock_With_Watch_Check: Secure process to import the Unlock Secret to the TOE's Secure Enclave for unlocking the device and approvals prompts.

7.1.1. Password_Setup

The TOE offers a configuration setting where the User can set up a password that will be used to perform authentication in order to access restricted services on the device including: first unlock after power-on or reboot, Apple Pay transactions, Apple Pay Transfers, and more. The TOE enforces strong access controls in order to prevent access to these restricted services without authentication (FIA_UID.2).

Note: *The User has the possibility to use password or any other activated (with enrolled templates) authentication method to perform unlock (beyond first unlock) or Apple Pay transactions*.

The TOE ensures the non-observability of the password during the setting process within the Secure Enclave (FPR_UNO.1).

7.1.2. Password_Verify

The TOE offers password entry to the User as part of the device unlock procedure, or during the User authorization step of an Apple Pay transaction. The User might have selected the default method as being biometrics, but password verification is always possible (as a fallback or by User choice). The TSFs ensure that password verification preserves the secrecy of the password value set by the User, which is expected in order to prevent an attacker from guessing the code by observing the verification process (FPR_UNO.1). The TOE ensures that the verification process would not validate a code that does not match the password set by the User and detect alterations to the configured value (FDP_SDI.1), preventing use of the User account (and related data) and forcing a device panic. To discourage brute-force password attacks, the TOE authentication failure policy is escalating time delays after the entry of an invalid password (FIA_AFL.1/Delay) or switching the boot into recoveryOS after 10 attempts (FIA_AFL.1/Recovery).

7.1.3. Password_Update

The User can update the password value through the device's settings menu. This functions as a verify operation, as knowledge of the previous password value is required to proceed to the setup step where the User types a new value and effectively updates the value in the Secure Enclave (FIA_UAU.2, FDP_ACC.2/Authentication_SFP and FDP_ACF.1/Authentication_SFP). The TOE ensures the non-observability of the old password during the verification as well as of the new password during the setting process (FPR_UNO.1). The TOE also enforces password alteration detection, preventing a corrupted password from being used to reset the authentication function (FDP_SDI.1).

7.1.4. Password_Delete

If the User wants to fully remove the use of the password on the device, because this would not allow an adequate security level for the processing of the TSF, the actual password is disabled and the iCloud Account information and all the Apple Pay data is deallocated. (FDP_RIP.1). To prevent misuse of the password deletion function, the User is required to first verify the current password before proceeding (FIA_UAU.2, FDP_ACC.2/Authentication_SFP and FDP_ACF.1/Authentication_SFP), and the TOE protects the password secrecy during the verification process in case an attacker was to use this path for attempting an observation-based attack (FPR_UNO.1).

The Secure Element, in the TOE Environment, is responsible for securely deleting the Apple Pay card data during this process.

7.1.5. Bio_Enroll

The TOE offers biometric authentication through the Touch ID service. To use Touch ID, the User must set up the device so that a password is required to unlock it, and password verification is required to be able to perform any modifications to the biometrics setting. Access to the fingerprint enrollment and the setting for enabling biometrics for Apple Pay, is gated by this password verification (FIA_UAU.2, FDP_ACC.2/Authentication_SFP and FDP_ACF.1/Authentication_SFP). The Secure Enclave inside the TOE ensures the non-observability of the new biometrics during the enrollment process (FPR_UNO.1). A User can enroll 5 fingerprints for Touch ID.

7.1.6. Bio_Update

Once a fingerprint is enrolled on a device, the User can update it within the settings of the Touch ID feature, for which access is gated by the password (FIA_UAU.2, FDP_ACC.2/ Authentication_SFP and FDP_ACF.1/ Authentication_SFP). After a successful password verification, the User can reset Touch ID, deleting the associated templates, and re-enroll their fingerprint later. This procedure is effectively enforcing the same security principles as a deletion (Bio_Delete), and an enrollment (Bio_Enroll): the Secure Enclave ensures that biometric data integrity is preserved, its manipulation is protected from observation, and the deallocation prevents attackers from finding any residual information (FPR_UNO.1, FDP_SDI.1, and FDP_RIP.1).

7.1.7. SF.Bio_Verify

After the disk is erased and password is entered to allow the device's normal mode of operation, the biometric authentication function is offered to the User as the default means for authentication means, if it is enabled, for device unlock, Apple Pay transactions. The User will present their biometric features to the device's biometric sensor, and the Secure Enclave will securely perform the verification of the submitted template to the enrolled biometric template(s). The Secure Enclave ensures the non-observability and integrity monitoring of the biometric templates during the verification process (FPR_UNO.1 and FDP_SDI.1). In case of verification failure, which means the TOE was not able to find a successful match, an authentication failure policy is enforced and password verification is required before the biometric authentication function is enabled again (FIA_AFL.1/Biometric).

7.1.8. Bio_Delete

The User has the capability to delete all enrolled biometrics from the Touch ID settings on the device using the Reset Touch ID option. This function deletes the associated template of all the fingerprints enrolled. This function, like the others, is gated by the password (FIA_UAU.2, FDP_ACC.2/ Authentication_SFP and FDP_ACF.1/ Authentication SFP). The deletion process ensures that no residual information of the deallocated data is left behind or leaked to a potential observer (FDP_RIP.1, and FPR_UNO.1).

7.1.9. Unlock_With_Watch_Setup

Optionally, the User can decide to enable a feature called "Unlock with Apple Watch" so that the MacBook Air can be unlocked by the Watch. Enabling this feature requires the User to enter the password on the MacBook Air (FDP_ACC.2/Authentication_SFP, FDP_ACF.1/ Authentication _SFP and FMT_MSA.1) before the setup process is started.

During this process, the TOE Secure Enclave creates a non-predictable Unlock Secret (FIA_SOS.2 and FPR_UNO.1).

The TOE's Secure Enclave securely exports the Unlock Secret to the paired Watch's Secure Enclave applying security measures to the sensitive asset with Secure Enclave/Watch Pairing Secret (FTP_ITC.1/Watch, FDP_UCT.1/Watch, FDP_UIT.1/Watch, FPT_RPL.1).

Until the feature is disabled, the TOE maintains the security attributes and assets of the Unlock with Apple Watch process, including the Unlock Secret (FIA_ATD.1).

7.1.10. Unlock_With_Watch_Check

Optionally, if the User enabled the feature called "Unlock with Apple Watch", the MacBook Air user can use a paired watch to:

- Unlock the device
- Approve other requests (related to the security functions of the TOE) to enter their administrator password

During this process, the paired Watch's Secure Enclave securely imports the Unlock Secret to the TOE's Secure Enclave, applying security measures to the sensitive asset with Secure Enclave/Watch Pairing Secret (FTP_ITC.1/Watch, FDP_UCT.1/Watch, FDP_UIT.1/Watch, FPT_RPL.1).

7.2.   SF Biometric/Secure Enclave secure channel

The TSF protects the data exchanged between the biometric sensor and the Secure Enclave against disclosure and modification (FDP_ITT.1)

7.3.   SF Secure Enclave/Secure Element secure channel

The TSF is able to initialize a secure channel with the Secure Element (FTP_ITC.1/SE). This secure channel protects the exchanged data with its corresponding security property: against disclosure (FDP_UCT.1/SE), modification (FDP_UIT.1/SE) and replay (FPT_RPL.1/SE).

7.4.   SF Secure Enclave/Watch secure channel

The TSF is able to initialize a secure channel to S.Watch (FTP_ITC.1/Watch). This secure channel protects the exchanged data against disclosure (FDP_UCT.1/Watch), modification (FDP_UIT.1/Watch) and replay (FPT_RPL.1/Watch).

7.5.   SF Card Data management

The following table summarizes the functions supporting Card Data management.

Function	Description
Apple Pay card data management	AP_Card_Provisioning: Provisioning of a new Card for Apple Pay.
	AP_Cancellation: User removes a card from Apple Wallet.
	AP_Revocation: Issuer initiated suspend/unlink of an Apple Pay Card in Wallet.

7.5.1. AP_Card_Provisioning

On the TOE, there are different ways to add a card into Apple Wallet:

- Adding a card manually
- Adding cards on file from an iTunes Store account to Apple Pay
- Adding cards from a card issuer's website

The first three modes are only available to an authenticated User on the device, with the password enabled (FIA_UAU.2, FDP_ACC.2/Card_Perso_SFP, and FDP_ACF.1/Card_Perso_SFP).

When a User adds a card to Wallet, the TSF encrypts card data (FPT_ITC.1) and sends it to Apple servers (FDP_ETC.2/Card_Perso_SFP). Full card numbers are not stored on the device (FDP_ITC.1/Card_Perso_SFP) or on Apple servers.

Integrity protection is in place to prevent alteration of the enrolled Apple Pay card data (FDP_SDI.1).

7.5.2. AP_Cancellation

When the User decides to remove an Apple Pay card from the Apple Wallet, the TSF orders the Secure Element to securely invalidate and remove the Device Account Number (FDP_RIP.1).

7.5.3. AP_Revocation

When the Issuer decides to suspend or unlink an Apple Pay card from the Apple Wallet of a TOE User, the TSF orders the Secure Element to securely invalidate and remove the Device Account Number (FDP_RIP.1).

7.6.   SF Payment management

When a device initiates an Apple Pay transaction, the Secure Element, in the TOE Environment, only allows a payment to be made after it receives authorization from the Secure Enclave. This involves confirming the User has provided intent and has authenticated with biometric authentication, or using the device password (FDP_ACC.2/Payment_SFP, FDP_ACF.1/Payment_SFP, and FMT_SMR.1). Biometric authentication is the default method if available, but the password can be used at any time. A password is automatically offered after three unsuccessful attempts to match biometrics; after five unsuccessful attempts, the password is required. A password is also required when biometric authentication is not configured or not enabled for Apple Pay (FMT_SMF.1, FMT_MSA.1).

Apple Pay includes an anti-replay mechanism that prevents transactions from being repeated by including in D.Payment_Data (An Apple Pay server nonce).

The processing of the Apple Pay transaction happens in the TOE Environment, on the Secure Element, using the secret card data, the Transaction Token and producing a payment cryptogram. The TOE ensures that the payment evidence transmitted back to the Issuer for processing (through a Terminal or network) was authorized by the User (FIA_UAU.6 Re-authenticating). In the case of Apple Pay online transactions, which are processed through the Apple Pay server, this integrity protection ensures the Dynamic Linking of the transaction data by a cryptographic based Authentication Code as exposed in the PSD2 regulation. The Apple Pay server ensures the integrity of the Dynamic Linking, and the Issuer verifies that it corresponds to a valid transaction, containing the right Transaction Token and produced by a genuine Apple Pay card (FDP_DAU.1).

The exported user data (transaction data displayed to S.User) is controlled by FDP_ETC.2/Transaction.

The Secure Element, in the TOE Environment, is responsible for ensuring the confidentiality of the Apple Pay Card data during the transaction processing.

The following table summarizes the function supporting Payment management.

TSF	Description
AP_Transaction	Processing of an Apple Pay transaction.

7.7.   SF OS Update

An OS update can be offered at any time by Apple to the User. The User is required to authenticate through a password verification, or the update cannot be installed (FIA_UAU.6, FIA_UAU.2, FMT_MTD.3).

The OS update preserves the user attributes, especially the card data, the password, the enrolled biometric patterns, and other authentication parameters (FIA_ATD.1).

The ability to update the D.OS is restricted to Authenticated user (FMT_MTD.1).

7.8.   SF iCloud logout & disk erase

An iCloud logout is performed when the User unlinks a device from an iCloud account. When this happens, the TOE ensures that the iCloud Account related data is securely deallocated, and that no residual information is left behind (FDP_RIP.1).

The most destructive security function available to the User on a macOS device is the disk erase, as it erases of all the User data. When this is performed, the TOE ensures that all the sensitive data deallocated as part of a disk erase is protected from leaving residual information. This covers the iCloud Account information, all the Apple Pay Card Data, and the User authentication data like password and biometrics (FDP_RIP.1).

Change History

Date	Version	Author	Comments
2021-09-10	1.0	Apple	Initialization of the Security Target
2021-10-15	1.1	Apple	Minor updates
2021-11-19	1.2	Apple	Updates according to internal reviews
2021-12-17	1.3	Apple	Minor fixes
2022-01-17	1.4	Apple	Minor fixes
2022-02-15	1.5	Apple	Minor fixes
2022-04-15	1.6	Apple	Updates according to internal reviews