Protector OATH SDK v5.4.0

Security Target – Android – Light Version

THALES

June 2020

Contents

This document describes the security target for first level security certification (CSPN) of product *Protector OATH* by the “Agence Nationale de la Sécurité des Systèmes d'Information” (ANSSI).

Document History

Version	Author	Description
1.3	Thales DIS – Hazrat Pradipta Ranjali	• Light version of Protector OATH SDK 5.4.0 security target.

References

Reference	Description
[PG]	Ezio Protector OATH Programmers’ Guide. On-line documentation available at https://thales-protector-oath-sdk.docs.stoplight.io/releases/5.4.0
[HOTP]	https://tools.ietf.org/html/rfc4226
[TOTP]	https://tools.ietf.org/html/rfc6238
[OATH]	https://tools.ietf.org/html/rfc6287
[RGS_B1]	Référentiel Général de Sécurité, Annexe B1 Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques.
[RGS_B2]	Référentiel Général de Sécurité, Annexe B2 Règles et recommandations concernant la gestion des clés utilisées dans les mécanismes cryptographiques.
[RGS_B3]	Référentiel Général de Sécurité, Annexe B3 Règles et recommandations concernant les mécanismes d’authentification.

Product identification

Organization	Thales Group DIS
Organization website	www.thalesgroup.com
Product Name	Protector OATH SDK
Evaluated Product Version	5.4.0
Product Category	Identification, Authentication and Access Control
Product Programmer's Guide	https://thales-protector-oath-sdk.docs.stoplight.io/releases/5.4.0

Product arguments

General product description

Protector OATH SDK is part of a solution for one time password (OTP) generation, secure storage and out of band messages exchange. The solution is composed by a library *Protector OATH SDK* for mobile application and many server components: *Mobile EPS* (Enrolment and Provisioning Server), *MSM* (Mobile Secure Messenger) and an Authentication Server. The solution enables developers to integrate a strong authentication layer for mobile users.

The library *Protector OATH SDK*, target of this certification, provides to mobile application developers an abstraction layer for security functions. The library provides mechanisms for provisioning and storage of secret keys involved in OTP generation. The library provides also a service for messages exchange and transaction verification as well as a secure storage mechanism.

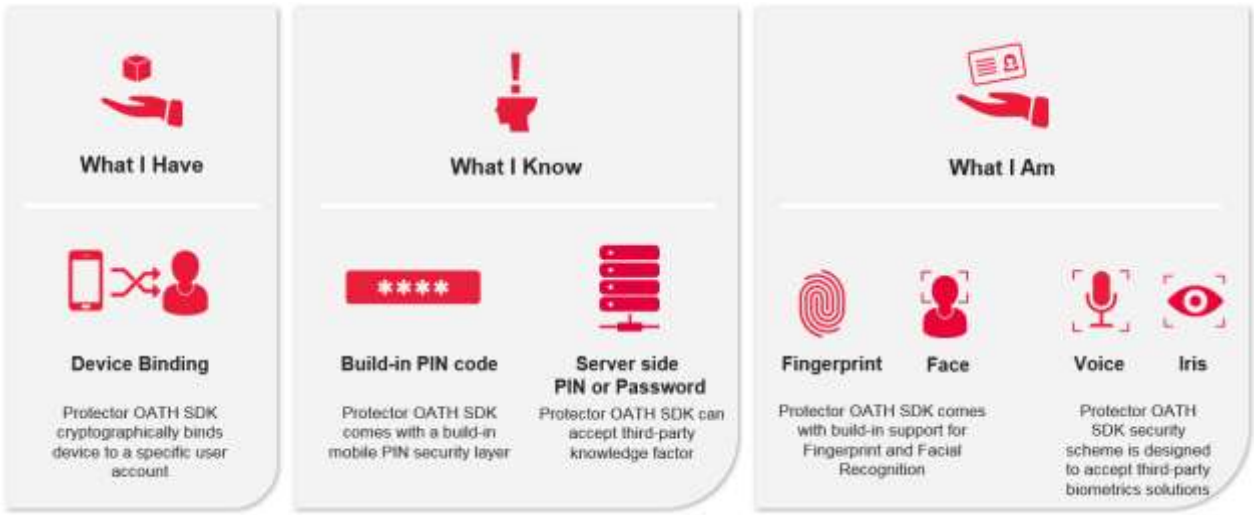
The evaluation perimeter contains the library *Protector OATH SDK*, the mobile equipment library, restricted to required services for “Identification, Authentication and Access Control” category.

Authentication

The *Protector OATH* implements the following authentication factors:

- Category “What I have”:
    - A mobile equipment that is running a strong authentication application built on top of the product *Protector OATH SDK*. The library manages and protects the secret key needed to provide strong authentication service.
- Category “What I know”:
    - A Pin value known by the user and required in order to provide the strong authentication service.
- Category “What I am”:
    - A biometric (such as fingerprint recognition, face recognition, or iris scan), provided by the mobile equipment, optionally required to provide the strong authentication service.

In this document, mention of biometrics indicate either fingerprint, face or iris scan depending on the support in the mobile equipment.



The *Protector OATH SDK* is available for both iOS and Android mobile equipment. This evaluation perimeter covers Android platform.

Product usage description

The service provided by the product and in the evaluation perimeter is the OTP generation. In addition, the Password Manager to unlock features domains into the product and Secure PinPad to input a Pin are in the perimeter.

Other features of *Protector OATH SDK*, like the Out of Band messaging, Secure Storage and support functions are not in the perimeter.

One Time Password

The user, as a client of the remote service, downloads the strong authentication application that embeds the product *Protector OATH SDK*, from an application store.

Protector OATH SDK provides several mechanisms that need to be used by the application to setup and then use strong authentication service:

- Provisioning of a secret key from the back-end component into the mobile equipment. This is the process to inject the secret into user's device. The secret will be protected by several security layers and sealed with device information before it is stored.
- Generation of OTP values after user authentication. The previously injected secret can be used to generate OTP for authentication. It is not possible to generate a valid OTP without user authentication. The generation requires the factor "What I have" and at least one of the categories "What I know" or "What I Am".

Like all features domain in *Protector OATH*, the One Time Password feature is protected by a domain key whereby the key is managed by the Password Manager of the *Protector OATH*.

The *Protector OATH SDK* supports OTP generation algorithms: CAP, OATH ([HOTP], [TOTP] and [OCRA]), Thales OATH and Dynamic Signature (a Thales proprietary algorithm). OATH OTP implementation can be standard or based on a white-box cryptography library.

This security target perimeter includes only the OATH OTP generation algorithm and implemented in the standard form.

Secure PinPad

The strong authentication application can optionally invoke Secure Pin Pad when a Pin is required from the user. Secure Pin Pad provides a way to manage and manipulate the Pin input process in a secured, controlled and verified manner.

Password Manager

Password Manager provides mechanism to protect features of the *Protector OATH* with a password that can be set globally. The password manager is in charge of unlocking the different domains of *Protector OATH*. A domain is a set of features, and is protected by a key, there are three domains:

- OTP domain.
- Secure Storage domain (not in the evaluation perimeter).
- Out of Band domain (not in the evaluation perimeter).

A password can be used to protect the domain keys.

The application that wants to use one domain of *Protector OATH* needs first to unlock it by providing the correct password.

Product environment description

Authentication

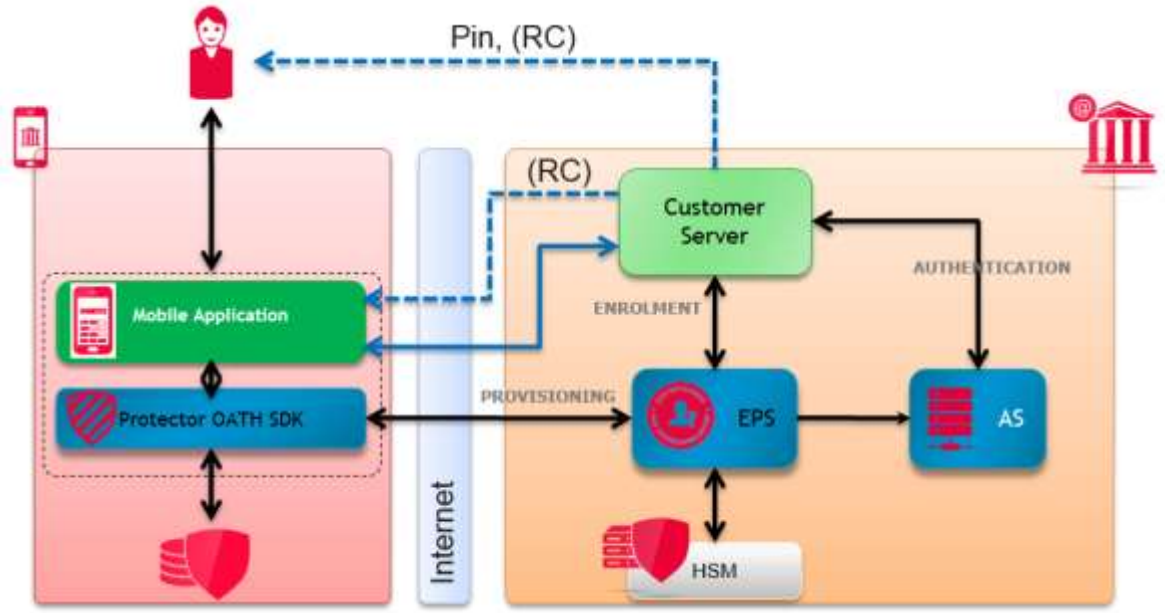


FIGURE 1 - SOLUTION FOR OTP GENERATION

The product environment is mainly related to provide a strong authentication to customer service.

During **enrolment** the customer back-end server request the *Mobile EPS* to create a new user. It implies at least the generation of the secret that will be injected (provisioning) into the end user mobile equipment in next steps, an initial Pin value and a Registration Code (RC) that uniquely identify the secret. Pin and Registration Code will be send to the user or to the application, depending on the deployment use case. The *Mobile EPS* will also push the secret into the Authentication Server deployed for the solution for further authentication of the user. Although enrolment is a mandatory step to create a new user into the solution, the *Protector OATH* is not involved into this first step.

During **provisioning** the *Protector OATH SDK* will securely retrieve the secret from the *Mobile EPS*, uniquely identified by the Registration Code.

During **nominal** usage (user authentication), the user can use the application to authenticate himself to the service. First the user needs to authenticate himself by one of the activated mechanism (Pin or Biometric in this evaluation perimeter), then the application will use *Protector OATH SDK* and this authentication mean to request an OTP generation to the library. Depending on the deployment use case, the OTP can be sent to the customer server by the application or displayed to the user for further input on a website for example.

When an OTP is received by the customer server, the service checks the OTP validity against the Authentication Server linked to the *Mobile EPS* to grant access or not to the user.

Environment hypothesis

- HM1: The user is managing is mobile equipment in a way to minimize security risks:
    - The mobile equipment operating system is up-to-date and the latest security patches available are applied.
    - Filesystem encryption is activated if available.
    - An operating system lock, considered as robust, is active.
    - The user doesn't record his Pin in the mobile equipment nor transmits it to a third party. The Pin is not used for any other usage than the strong authentication application.
- HM2: The mobile equipment enforces a first level of protection:
    - The root certificate authority of the mobile equipment is considered as trusted.
    - The mobile equipment has capacity to connect to the solution servers.
    - The random number generator has sufficient entropy.
    - The cryptography primitives provided by the operating system are resistant to state-of-the-art "basic" attacks.
- HM3: The strong authentication application using *Protector OATH SDK* follows guidelines defined into [PG].
- HM4: The biometric service provided by the operating system is designed to protect assets bound to the biometric and minimize false positive. It is compliant with the [RGS_B3] rules about User Authentication and Machine Authentication with a trusted local environment.
- HM5: The remote service must have an operational and trusted *Mobile EPS* and Authentication Server. The remote service usage by a client must enable the association of an OTP with the according user profile recorded in the Authentication Server.
- HM6: The platform's provided random number generator has sufficient quality to be used as source of entropy to a random number generator designed to be compliant with [RGS_B1] rules.

Dependency description

At least Android version 5 is required for the *Protector OATH SDK* to run.

To activate and use the biometric authentication, the minimal version number of Android is 6 and the mobile equipment must have a hardware sensor for fingerprint. To support other biometric type (for example Face or Iris), Android 9 (or later) is required.

Users and typical roles

- MU1: the user has access to the strong authentication application developed with the *Protector OATH SDK* and is able to authenticate himself by one of the activated mean (Pin or biometric). The user is

typically a client of the service that need to authenticate himself to the remote service. The user is not considered malicious or excessively careless (HM1).

Product evaluation perimeter

The evaluation perimeter is the *Protector OATH* product, restricted to the following features, linked to the authentication service, in particular the various authentication factors and the authentication code generation (the proof of identity):

Authentication

- OTP OATH generation (OCRA included).
- OATH secret provisioning from the *Mobile EPS* (restricted to the Provisioning Protocol v3).
- OATH secret storage into the mobile equipment.
- Secure PinPad to input securely the Pin.
- User authentication by Pin.
- User authentication by biometric (Fingerprint, Face or Iris, depending of platform support).

Used by Authentication

- Password Manager.

The following elements, present into the product, are not considered within the evaluation perimeter:

- OTP CAP: computation and provisioning.
- OTP OATH in white-box cryptography format.
- EMV QR.
- MSP (Mobile Signing Protocol).
- DCVV (Dynamic CVV).
- Provisioning Protocols other than v3 (in particular PPv5 required for WBC).
- Dynamic Signature (proprietary protocol conforming CAP).
- Offline provisioning, seed importation inheritance (migration from *Protector OATH* 1.x to 2.x).
- OTP VIC verification.
- “Dual seed” support for OATH.
- Out Of Band service.

- Secure Storage.

Technical environment for product usage

The *Protector OATH SDK* is developed to be run on standard Android mobile equipment.

Hardware compatibility

The technical environment to use the strong authentication application, developed with the *Protector OATH SDK*, requires a physical handset (smartphone or tablet) supporting mobile application execution environment and IP networking (through SIM card with data subscription or through WIFI).

The optional user authentication mechanism with biometric requires a hardware sensor on the physical handset.

Operating system

The strong authentication application developed with the *Protector OATH* SDK under evaluation requires mobile equipment running Android version 5.0 up to version 10 on architectures `armeabi-v7a` or `arm64-v8a`.

The optional user authentication mechanism with biometric requires Android version 6 at a minimum.

Assets to be protected by the product

The sensitive assets to be protected are ones involved into the authentication (via Pin or Biometric), the secret key generated by the *Mobile EPS* that is associated to a unique user and assets managed by the Password Manager.

The Property Persistent Storage is an internal component (not exposed to the application makers) used to store some of the assets managed by the product and as such is in the evaluation perimeter. It is also used to provide a Secure Storage service to the application, but this wrapper (Secure Storage) around this component is not in the evaluation perimeter.

While provisioning of the secret key on the mobile equipment, some local assets are generated to ensure the secret key confidentiality and provide security services. All these assets are used in further steps to generate an OTP, enabling user authentication on a remote service.

The following sensitive assets are identified for the **Authentication** perimeter:

- B1: Pin, this asset is volatile, provided by the user.
- B2: the secret key used to generate OTP values, stored in persistent memory on the mobile, protected by some other assets:
    - B2.1: the storage key. This asset is volatile, derived from asset B3 and is unique to each secret key.
    - B2.3: the environment key. This asset is volatile.
    - B2.3: the enciphering key. This asset is volatile, derived from user authentication.

  In case the secret key B2 has been upgraded to support multi-authentication mode the secure scheme is updated and new assets are introduced:
    - B2.4: the authentication wrapper key. This asset is stored encrypted in persistent memory.
    - B2.5: the asymmetric biometric key. This asset is persistent. The product uses the fingerprint match on device as authentication factor.
    - B2.6: the biometric fingerprint data. This asset is persistent, stored encrypted.
    - B2.7: the cache key. This asset is persistent.
    - B2.8: the biometric fingerprint enciphering key. This asset is volatile.

During the provisioning the secret key is transferred from the server to the mobile equipment.

- B2.9: the session key for confidentiality. This asset is volatile.
- B2.10: the session key for authentication. This asset is volatile.
- B2.11: the *Mobile EPS* public key.

Those assets usage depends on the activation or not of authentication modes. Application that uses *Protector OATH SDK* needs first to upgrade it to support multi-authentication mode and then activate the biometric fingerprint if needed.

The following sensitive assets are identified for the **Authentication** and **Password Manager** and **Property Persistent Storage** perimeters:

- B3: the master key. This asset is persistent.
  - This asset is stored in the mobile, managed by the Password Manager and protected by the Property Persistent Storage using the asset B4.2.
  - This asset is used by the OTP domain, provided by the Password Manager, to derive the volatile asset B2.1.

The following assets are identified for the **Password Manager** and the **Property Persistent Storage** perimeter:

- B4.1: Password. This asset is volatile.
- B4.2: the application key. This asset is stored in the mobile equipment.

The following assets are identified for the **Property Persistent Storage** perimeter:

- B5.1: A password or a key. This asset is volatile.
- B5.2: The property storage secret key. This asset is volatile.
- B5.3: The secret Key layer 1. This asset is stored in the mobile.
- B5.4: The encryption secret Key L1. This asset is volatile and derived from asset B5.3.
- B5.5: The authentication secret key L1. This asset is volatile and derived from asset B5.3.
- B5.6: The encryption secret Key L2. This asset is volatile.
- B5.7: The encryption secret Key L3. This asset is volatile.
- B5.8: The data securely stored by the Property Persistent Storage.

The Property Persistent Storage is used by the Password Manager.

Threat description

The security model of the *Protector OATH* has been designed to counter-act the following attack vectors:

- Attack during secret key exchange between the mobile equipment and the EPS server;
- Attack on secret key stored on the mobile handset (offline attack);
- Attack on user authentication;
- Attack on data protected by the Secure Storage service on mobile equipment (offline attack);
- Attack on keys protected by the Password Manager;
- Attack during cryptographic computation (for the various domains, OTP computation and Secure Storage)

The following threat scenarios are identified:

- M1 - mobile equipment theft: a malicious user can try to impersonate the legitimate user or try to access application's secret (physical access to the mobile equipment).
- M2 - Brute force on a secret or an authentication factor:
    - Pin.
    - Biometric.
    - Secret keys.
- M3 - Pin or password or secret key access in real time during the cryptographic operations.
- M4 - Threats on the authentication code:
    - OTP replay.
    - Forge a new OTP from a valid one.
    - Reverse engineering of authentication factor from a valid OTP.
- M5 - Secret key interception during provisioning.
- M6 - Pin and/or password theft;
- M7 - Secret key cloning;

Security function description

FS1 - Pin management

User Pin is one factor that can be used in the authentication scheme.

Secure Pin Pad

Secure Pin Pad is a single visual view that provides security for data entries. The purpose of Secure Pin Pad is to ensure that the management and manipulation of the Pin code are conducted in a secured, controlled, and verified way. It ensures that the protection mechanisms are in place to defeat or mitigate key logger, over shoulder attacks, memory dump and screen capture. Secure Pin Pad helps a software solution to prevent targeted attacks more efficiently.

The Secure Pin Pad feature uses internally a Thales DIS library to handle Secure Key Pad. You can configure elements of the Secure Pin Pad using the initialization parameters or through APIs offered by the product.

Pin blockage

The secret key is encrypted with the Pin on the mobile. The security relies on the property that a bad Pin must generate a bad OTP that is not distinguishable from a good OTP.

The attacker cannot validate an OTP without submitting it to the authentication server for checking. The authentication server enforces a threshold policy to limit the number of wrong OTP and disable the user service if required.

There isn't any mechanism in *Protector OATH SDK* that permits to know the entered Pin is the good one or not and no way to block locally the Pin in the *Protector OATH SDK*.

Pin change

The Pin can be changed in the strong authentication application, developed with the *Mobile SDK SDK*. This operation involves a modification of the encryption layer of the secret key that integrates the new Pin.

FS2 - Biometric management

Once the secret to generate OTP is provisioned into the mobile equipment, it is possible to activate the authentication with biometric (such as face recognition, fingerprint recognition, or iris scan), if hardware and operating system requirements match the minimum required ones.

Multi-Authentication mode

In order to allow usage of other authentication factors the secure scheme used to protect the secret needs first to be upgraded.

During upgrade the user needs to authenticate himself by using his Pin. The secure scheme is then upgraded to allow both Pin and biometric to unlock the secret. This upgrade step is needed only once per secret and cannot be reversed.

As there isn't any way to know locally the Pin is correct, providing a wrong Pin during secure scheme upgrade will lead to an invalid state that cannot be reversed, the secret will not be able to generate valid OTP anymore. It is then strongly recommended to validate an OTP with the authentication server with the provided Pin before starting the upgrade. This is the only possible way to validate the Pin was correct.

Biometric activation

Biometric can be activated after the secure scheme has been upgraded. It can be activated and used per secret to generate OTP if the platform support it and if user has preciously enrolled himself into the mobile equipment biometric system. The user needs to provide his Pin in order to activate the new authentication factor.

Any secret that has been upgraded to the multi-authentication mode secure scheme can support the biometric authentication factors.

Biometric deactivation

At any time the application can request to deactivate the biometric support for any of the secret.

FS3 - Confidentiality protection of secret key during provisioning

The secret key provisioning on mobile handset from the *Mobile EPS* involves two security layers, a TLS connection that is used to transport messages of a proprietary protocol. The following steps are involved into the proprietary protocol, to protect the secret key during every exchange operations:

1. The strong authentication application starts and detects that the secret key is missing.
   This state triggers a process to get a secret key for OTP generation from the back-end, identified by a registration code. Depending on the deployment scenario this registration code can be entered by the user or retrieved by the application from a back-end server.
   The registration code, bound to the user, uniquely identify the secret key on the back-end side.
2. The *Protector OATH SDK* generates assets to protect the transfer.
3. Assets and the registration code are wrapped with the public key of the *Mobile EPS* and send to it.
4. On the *Mobile EPS*, the security module unwrap assets and the registration code.
5. On the *Mobile EPS*, the secret key corresponding to the registration code is selected.

6. The security module encrypts the secret. The *Mobile EPS* gets the message and sent it to the mobile application.
7. On reception in the mobile equipment the message is authenticated and then decrypted.

The secret key confidentiality is ensured during all the steps since an encryption layer with the Pin is enforced during all the process. The Pin is not needed during the provisioning, and the secret key is never in plaintext. Moreover a mutual authentication between the mobile application and the *Mobile EPS* is enforced:

- The mobile application is authenticated when the registration code is submitted for validation. Hence the secret key is delivered to an authorized user.
- The registration code is encrypted with the public key of the server targeted. This ensures that the registration code is used only by the authorized server.

The *Protector OATH SDK* rejects any plaintext communication (HTTP), self-signed certificate, host mismatch and enforce the server certificate to be signed by a root-CA trusted in the mobile. Those security checks cannot be deactivated.

The *Protector OATH SDK* apply then several security layers to provide anti-cloning and platform security features before storing the secret into the persistent memory.

FS4 - Confidentiality protection of secret key stored for OTP

The secret key is received encrypted into the mobile.

The confidentiality protection of the keys stored on the mobile equipment follow these principles:

1. The enciphered secret key is encrypted using a volatile environment key.
2. The double enciphered secret key is then encrypted with a key derived from the master key.
3. The data is natively protected by the segregation of mobile execution environment managed by the operating system that prevents an application to access to the data of another application.

In case of multi-authentication mode upgrade, the first security layer provided by the Pin-derived key is replaced by a wrap key. The wrap key is stored in the persistent memory encrypted.

FS5 - Confidentiality protection of keys during OTP computation

The OTP computation requires secrets from 3 different environments:

- The operating system: the master key.
- The execution environment.
- The user authentication.

Finally the secret for OTP computation is unwrapped with the latest key obtained from the user authentication layer. When the secret key is accessible in memory, the OTP generation is achieved.

Integrity protection of sensitive data

The sensitive assets for OTP generation are not protected in integrity by construction. Indeed, according to the fundamental principle used in the mobile solution, a bad OTP must not be distinguished from a good OTP. The sensitive assets might be altered by an attacker without breaking the security model, any attempt to alter assets will lead to invalid OTP generation and invalid authentication will be detected on the authentication server.

FS6 - Confidentiality protection of keys in Password Manager

Password manager uses the internal mechanism, Property Persistent Storage, to provide confidentiality and authentication. This service is not exposed directly to the application.

Confidentiality and authentication in Property Persistent Storage

The confidentiality and authentication protection of the data stored on the mobile equipment follow these principles:

1. An asset is generated, stored and protected by the Property Persistent Storage.
2. The data, provided by the application to the Property Persistent Storage is protected with three different properties :
    a. Encryption and authentication is provided from asset protected by the platform.
    b. Anti-cloning mechanism is applied from platform.
    c. Anti-cloning mechanism is applied from application.

Confidentiality and authentication of keys in Password Manager

The Password Manager uses several instances of Property Persistent Storage. In order to unlock the storage the following principles are used:

1. An asset provided by the application is used to unlock a first Property Persistent Storage instance. This instance contains Application Key.
2. The Application Key is then used to unlock Property Persistent Storage for the usage domain.

FS7 – One Time Password algorithm

Protector OATH SDK, restricted to this evaluation perimeter, uses standard OTP algorithms [HOTP], [TOTP] and [OATH].

Threat coverage by product security functions

	M1: Mobile equipment theft	**M2:** Brute force	**M3:** Access to asset during cryptographic	**M4:** Threat on authentication code	**M5:** Secret Interception during provisioning	**M6:** Pin or password theft	**M7:** Secret key cloning
FS1 – Pin Management	X	X				X	
FS2 – Biometric management	X	X					
FS3 – Confidentiality of secret key during provisioning			X		X		
FS4 – Confidentiality of secret key stored for OTP	X	X	X				X
FS5 – Confidentiality of keys during OTP computation	X	X	X				
FS6 – Confidentiality and integrity of keys in Password Manager	X	X	X				X
FS7 – One Time Password algorithm			X	X			

Protection rationale

Threat	Protection
M1: mobile handset theft	The secret key is protected by the Pin or the Biometric. • The Pin is not known by the attacker. No data derived from the Pin is stored in persistent memory, mitigating any brute force attack on it even with full mobile equipment reverse and analysis. Pin can only be validated by submitting the generated OTP to the authentication server, the server enforce strict retry counter management in a

	secure environment, blocking the account after a defined number of false OTP. • Biometric verification is provided by the mobile equipment manufacturer. Access control to keys to unlock the secret to generate OTP is enforced by the platform into hardware-backed keystore. • Secret key for OTP as well as application's data are sotred encrypted into the mobile equipment.
M2: Brute force	• No data derived from the Pin is stored in persistent memory, mitigating any brute force attack on it even with full mobile equipment reverse and analysis. Pin can only be validated by submitting the generated OTP to the authentication server, the server enforce strict retry counter management in a secure environment, blocking the account after a defined number of false OTP. • The product relies on mobile equipment for biometric • Cryptographic secret uses state-of-the art recommendation on key strength and cannot be brute forced.
M3: Access to asset during cryptographic operation	This attack requires a rooted mobile handset and a dedicated malicious application installed on the mobile handset. The *Ezio Protector OATH* enforce a proper wiping of all assets at the end of usage. Secret key for OTP computation are managed in the native part of the product (not in Java layers). Support security functions make attacks on the field more complex (root detection, hook detection, tamper detection, debugger detection, emulator detection, overlay detection).
M4: Threats on the authentication code	*Ezio Protector OATH* uses standard OTP algorithms, reviewed by experts. Replay is not possible and is enforced in Authentication Server.
M5: Secret key interception during provisioning	All the communications between the *Protector OATH* SDK and the *Mobile EPS* are encrypted with MPP (the Thales proprietary Mobile Provisioning Protocol) and TLS protocols.
M6: Pin and/or password theft	Useless without the secret key stored on the mobile equipment.
M7: Secret key cloning	A copy of the secret key stored on the mobile equipment cannot be used on another one since the volatile environment key is unique by mobile. A copy of the secret key without Pin is useless and keys involves into the Biometric are protected into hardware-backed keystore.