SECURITY TARGET FOR biocertiX

Common Criteria version 3.1 revision 5

Assurance Level EAL 2

CONTENTS

Terms

AC	Authentication Code/Keyboard Input
biocertiX	Business name of the TOE. The TOE consists of biocertiX software (signaturiX Core, signatiruX Admin, Document database, Licence and Configuration database) and biocertiX App accompanied by guidance documentation.
biocertiX server	Server (physical or virtual) where biocertiX software is installed.
ES	External System
Device	Samsung Tablet
HSM	Hardware Security Module
QR	Quick Response Code
QSCD	Qualified Signature Creation Device
Privileged User	The only privileged user is System Administrator
TOTP	Time-based One Time Password
TTP	Trusted Third Party
Vault	Secure Storage of Secrets

1. Introduction

1.1. ST Overview

This ST document defines the security objectives and requirements as well as the scope of the Common Criteria evaluation (according to the Common Criteria methodology) for the biocertiX

biocertiX allows to securely embed handwritten biometric signatures on PDF documents. The signatures are created with a S Pen using a Samsung Tablet (hereinafter referred to as Tablet) on which the biocertiX App mobile application is open. The signing of PDF documents is implemented in accordance with the standard: ETSI EN 319 142-1 V1.1.1 (2016-04) – PadES, signature level B-T [1].

The Target of Evaluation (TOE) is the combination of biocertiX software and the biocertiX App (mobile application). biocertiX software enables the External System to send PDF documents for signing, and users of the signaturiX Core module (the part of biocertiX software) to view the content of the documents and sign them using the biocertiX App (mobile application) open on a Tablet, whereby signing involves sampling the biometric data of the handwritten signature as it is created and embedding it in the PDF document being signed.

TOE is supported by the following software and hardware components to perform its tasks:

- ***External System (ES)*** - A system that allows authenticated users to send PDF documents to the signaturiX Core module for handwritten biometric signatures and receive PDF documents with an embedded biometric signature,
- ***Device (Samsung Tablet)*** - A set of hardware and software for sampling a biometric signature and its cryptographic protection, on which the biocertiX App mobile application is launched,
- ***Certum SimplySign*** - An external environment that enables electronic sealing and time stamping of PDF documents,
- ***Audit Database*** - Software that manages the audit of biocertiX system logs.
- ***Vault: Secret storage*** - A system for securely storing sensitive information (e.g., passwordto keystore with private key to decrypt biometrics provided with the biocertiX App, etc.).
- ***Samsung Knox Manage:*** A system that allows the management of Samsung Tablets (including (a) the configuration of the public key involved in encrypting the biometric data on the Tablet before sending it to signaturiX Core module, and (b) certificates used by TLS)

To ensure a secure working environment, the biocentriX Core software is delivered as a tamper-proof zip archive for which access (login and password) and the calculated SHA512 hash value is provided via email. The biocertiX software together with an External System and Tablet with the installed biocertiX App provide a handwritten biometric signature service on PDF documents.

1.2. ST Reference

This ST is identified by the following unique reference:

ST Title	SECURITY TARGET FOR biocertiX
ST Version	2.3-lite
ST Date	25.09.2023
ST Author	Asseco Data Systems

1.3. TOE Reference

This TOE is identified by the following unique reference:

TOE Name	biocertiX - handwritten biometric signatures on PDF documents
TOE Version[1]	1.1

[1] The versions of specific elements constituting TOE (version 1.1) are defined in section 1.5.1.

Evaluation Criteria	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology; CCMB-2017-04-004, version 3.1, revision 5, April 2017.
Evaluation Assurance Level	EAL 2
TOE Developer	Xtension Sp. z o.o.; Samsung Electronics Polska Sp. z o.o.; Asseco Data Systems S.A.
TOE Sponsor	Asseco Data Systems
Evaluation Facility	ITSEF NIT, Poland
Certification Authority	**NASK – National Research Institute, Standardisation and Certification Centre, Poland.**
Certification ID	2021-5

1.4. TOE Overview

biocertiX is a trustworthy system that offers a handwritten biometric signature service on PDF documents. biocertiX ensures that the biometric signature on the document was created by the BioSigner and that the signature is used for its intended purpose - to biometrically sign the document displayed to the BioSigner. The aim of the solution is to enable the expression, in a legally binding manner, of a declaration of intent in electronic form by persons who do not have the means to create an electronic signature or do not have the necessary skills to use such a signature.

biocertiX is a combination of web and mobile applications (biocertiX software and biocertiX App accordingly) for signing PDF documents (Figure 1).

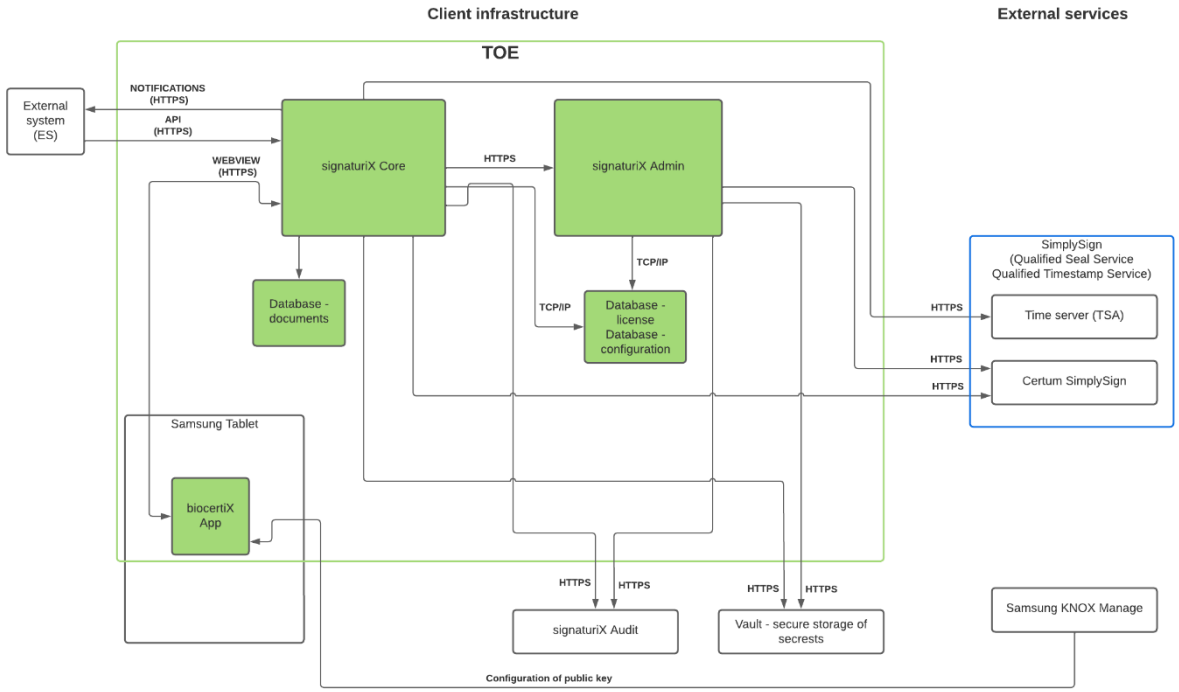

Figure 1: biocertiX – Secure Biosignature System

biocertiX software and biocertiX App are components of the TOE (biocertiX) that reside in a tamper-proof environment, providing the necessary functionality to protect the BioSigner attributes needed to securely create a handwritten biometric signature. Other elements are part of the system environment (elements outside the TOE, e.g. External System needed by the user to interact with the TOE, trusted third party services, etc.). Biometric

signatures require a biocertiX App (mobile application) installed on the Tablet with the ability to record the degree of S Pen pressure during the handwritten biometric signature creation.

The user interacts with the ES, which communicates with the TOE using encrypted HTTPS. The user is an individual who has at its disposal the Tablet equipped with S Pen. The ES using the signed digitally API of the signaturiX Core system sends the user a PDF document(s) to be displayed to BioSigner for signing on Tablet. The user and the BioSigner are not necessarily the same person. In response to the sent document(s), the signaturiX Core system generates and sends back to the ES a time-limited one-time QR/AC code. The ES displays this QR/AC code to the user. The user launches the biocertiX App on the Tablet and scans the QR code displayed on the ES or inputs AC code via the Tablet keyboard. This QR/AC code includes a unique authentication identifier of the PDF document(s) to be signed biometrically. The PDF document(s) is displayed on the Tablet in the biocertiX App and the BioSigner can review the content and sign it by providing a handwritten biometric signature on the Tablet.

The biometrics sample of the submitted signature is encrypted on the Tablet with a one-time symmetric key generated using a cryptographically strong random number generator [2, 3, 4] with hardware enhanced entropy provided by Samsung technology that complies with the statistical random number generator tests specified in NIST SP 800-90A [5]. The symmetric key is then encrypted with a public key configured on the Tablet during system initialization, and the corresponding private key is stored on the biocertiX server in a keystore file protected by a password stored in Vault. Each biocertiX instance has a pre-installation generated key pair, of which the public key is installed on the biocertiX App and private key is installed on signaturiX Core during initialization. The biometric signature secured in this way is sent to the signaturiX Core, where it is decrypted, converted to a standardized format and re-encrypted with a one-time symmetric key, which is encrypted with an HSM public key issued by a trusted third party – Certum SimplySign; The HSM public key is a 4096-bit RSA key generated for a given consumer by trusted third party (it is included in the TOE delivery in the license file). The corresponding RSA private key is held only by the TTP. Please note that the actual seal is performed by a TTP using a different pair of keys[2].

The user authentication process is provided by ES, therefore the TOE (signaturiX Core) does not store the credentials of individual users Through the signaturiX Core configuration service, a list of users (logins) who are authorised to use signaturiX Core is sent from the ES. When performing operations for service calls that have the login of a user in the request parameters, signaturiX Core checks whether this user is in the list of authorised users. It is the role of ES to ensure that only an authenticated user of the external system can send a request to signaturiX Core containing the login of this user.

All interactions of the BioSigner with signaturiX Core via ES must be carried out using HTTPS. The TOE receives the document(s) to be signed from the ES using the API of signaturiX Core (SOAP and/or REST). Each document or package of documents sent to the TOE (signaturiX Core) has a unique 'document/package token' assigned to it by the ES. In response to the uploaded document(s), the signaturiX Core returns to the ES a unique, one-time credentials in the form of a QR/AC code, based on which the ES user accesses the document(s) in signaturiX Core by scanning the QR code or input AC code via the Tablet keyboard in the biocertiX App. The biocertiX App presents the selected document(s) to the BioSigner and allows him/her to sign it. Once the signature is affixed, the TOE (signaturiX Core) generates audit records/logs and transfers them to an external audit database to store and secure these records/logs. The content of each audit record/log is electronically signed with a dedicated private key separate from the key used to sign/seal the signed document itself, and the auditor can verify signatures of the audit records/logs. The audit database is protected (record/log integrity is ensured) in the TOE environment.

1.4.1. TOE Type

The TOE type is "none" (undefined), as the TOE is not of a readily available type[3].

Additional note:

The TOE allows to securely embed handwritten biometric signatures on PDF documents The TOE is the combination of web and mobile applications (biocertiX software and biocertiX App accordingly). SignaturiX Core part of biocertiX software communicates with the biocertiX App using HTTPS.

[2] The certificate of the public key corresponding to the Certum SimplySign private key used to create the qualified electronic seal is available according to Certification Policy of Certum SimplySign concerning QSCD.

[3] According to Annex A - Specification of Security Targets subsection A.4.2.2 [7] the TOE type may be defined as "none" if the TOE is not of a readily available type.

The TOE implements a protocol (described in Section 1.4) for fetching and embedding signature biometrics into a PDF document.

1.4.2. TOE Usage & Major Security Functions

signaturiX Core ensures that the signature operation must be authorized using the ES.

Usage of the TOE includes the following steps:

1) Secure receipt from ES of PDF documents for biometric signature,
2) Authentication of PDF document(s) using QR/AC code,
3) Secure embedding of biometric data (captured form external S_Pen) in a PDF document,
4) Binding of seal and time-stamp with PDF document with embedded encrypted biometric data (the actual seal and time-stamp is prepared in Simply Sign outside of the TOE),
5) A biometrically signed document is securely made available for download by the ES.

The main utility and security functions of the TOE are:

- TOE initialization

➢ TOE works in the client-server architecture. The client part of the TOE's located on the Tablet (biocertiX App). Its initialization is performed using Knox Manage software, which provides automatic configuration of the public key certificate, security policies and biocertiX App on managed tablets. During initialization of the server part of the TOE, the corresponding private key used by signaturiX Core is installed in keystore on biocertiX server and password to this keystore is stored in Vault. The public key used to create the cryptogram of biometrics decryption key is placed in the license file as part of the license. The Knox Manage system is the sole entity responsible for the secure initialization and management of keys and certificates (public key involved in encrypting the biometric data in biocertiX App the and certificates for TLS) used by biocertiX App and it is considered as trusted and secure. The process of initialization and management of keys used by biocertiX App is considered to be secured by Knox Manage system and it is out scope of the TOE evaluated configuration.

- TOE Administration (server part of the TOE)

➢ The System Administrator is allowed to manage users and to configure the system.

➢ signaturiX Admin - configuration. is created in the administration application and each System Administrator is identified by a login and password and TOTP. The administration application is inside the TOE

- Signing operation

➢ The user can indicate (using ES) PDF documents for signing in signaturiX Core. Indicated documents are transferred from ES to signaturiX Core with user login via API. Upon acceptance of a time-limited QR/AC code provided by the TOE, the user is given access to a document(s) on which he/she can provide handwritten signatures or submit an open document for handwritten signature to another person.

- Audit

➢ An audit trail is produced of all security relevant events within the TOE and stored externally. Access to the Audit Database requires prior authentication using a login and password and TOTP. Management access to audit trail is outside the scope of the TOE.

TOE shall manage the Data Assets as defined in Section 3.1.

1.4.3. Required non-TOE Hardware/Software/Firmware

The following non-TOE software-based elements (accompanied by necessary hardware for this software/services) are required for the operation of the TOE (they are excluded from the scope of the TOE delivery):

- ***External System (ES)***: A Web application that integrates with biocertiX using the API of biocertiX. The ES sends PDF documents to signaturiX Core and presents to the user the authentication QR/AC code received in response. The QR code is scanned or AC code is inputted via the Tablet keyboard on a Tablet where the user is presented with the document(s) (based on the QR/AC code, the view of the biocertiX App is redirected to the document transferred from ES to signaturiX Core). Once the documents are approved in signaturiX Core, the ES receives the signed document(s) from the signaturiX core.
  The TOE consumer is responsible to either development or acquire the ES.

- ***Certum SimplySign:*** Qualified electronic seal and qualified Timestamps service. As part of this service, SimplySign signs the sent document digest/hash. The digest/hash signed in this way together with the seal's public key certificate and timestamp is saved in the PDF document by the biocertiX system. The access to both of the SimplySign services provided by Certum is required for TOE operation.
  The TOE consumer is responsible for obtaining access to Certum SimplySign: Qualified electronic seal and qualified Timestamps services that are used by the TOE.
- ***Samsung Knox Manage:*** A system that allows the management of Samsung Tablets (including the configuration of the public key involved in encrypting the biometric data on the Tablet before sending it to signaturiX Core). The access to Samsung Knox Manage service provided by Samsung is required for TOE operation. Samsung Knox Manage is a part of Samsung Knox Suite
  The TOE consumer is responsible for acquiring license Samsung Knox Suite.

Additionally, the following non-TOE hardware-based elements are required for the operation of the TOE (all hardware elements are excluded from the scope of the TOE delivery):

- ***Samsung Tablet:*** A mobile device (S3 upwards) that allow to use a hardware enhanced entropy in a cryptographically strong random number generator.
- Server hosting the biocertiX software together with vault. Minimum requirement for that server:
  - ✓ CPU 4 Core 2.4 GHz Intel(R) Xeon(R) CPU E5-2680 v4 or equivalent
  - ✓ RAM 4 GB
  - ✓ HDD 30 GB
  - ✓ OS Debian 10
- Server hosting signaturiX Audit where the signaturiX Audit is installed (the signaturiX Audit itself is included in to scope of the TOE delivery). It must be a different server than the server hosting biocertiX software together with vault. Minimum requirements for that server:
  - ✓ 2 Core 2.4 GHz Intel(R) Xeon(R) CPU E5-2680 v4 or equivalent
  - ✓ RAM 2 GB
  - ✓ HDD 20 GB
  - ✓ OS Debian 10

1.5. TOE Description

The TOE is a software component connected to external services (signaturiX Audit, Vault, Qualified Seal Service, Qualified Timestamp Service and ES[4]) using HTTPS. The TOE is located in a tamper-proof environment. In regards to biometric data processing the TOE meets the requirements of ISO / IEC 19794-7 for biometrics and standards for PDF documents ISO 32000-1.

1.5.1. Physical Scope of the TOE

The TOE consists of the following elements (see Table 1 below):

- ***biocertiX App*** **–** mobile application (for devices) that responsible for sampling a biometric signature and its cryptographic protection. The biocertiX App. shall be acquired from Google Play Store.
- ***signaturiX Core***: software element that enables the embedding of biometric data (collected and encrypted handwritten biometric signature data using device) in PDF documents according to the protocol described in section 1.4.
- ***Document database***: A postgres database that stores documents in memory for the duration of their processing in signaturiX Core. This ensures that documents are not stored on the signaturiX Core server file system.
- ***Database (licenses and configuration)***: A postgres database that stores information about the logins of users who have been authorized by the API of biocertiX to access the biocertiX system and use its functionalities (including, for example, qualified seals). The configuration of the biocertiX appearance (colours, logos) and the current values of the biocertiX system parameters are also stored there. The logins of users authorized to use the biocertiX system are transmitted via the secure API of biocertiX.
- ***signaturiX Admin:*** An administration application that allows trusted System Administrators to configure the system parameters (tomcat 9 with the signaturix-admin web application).

The following guidance documentation are needed for compliant TOE setup:

[4] The Samsung Knox Manage, depicted in Figure 2, is not connected to the TOE during its operation. It is used solely utilized during the installation procedure (specifically during the TOE initialization).

- Installation, Configuration and Maintenance of TOE

The following non-TOE elements are provided together with the TOE:

- ***Vault: Secret storage*** - A system for securely storing sensitive information (e.g., password to keystore with private key to decrypt biometrics provided with the biocertiX App, etc.)
- ***signaturix-audit***: an auditing system with a database that stores audit records/logs (tomcat 9 with the signaturix-audit web application)
- ***signaturix-starter*** (alpine Linux with initialisation script)

The TOE (biocertiX - handwritten biometric signatures on PDF documents version 1.1 as defined in section 1.3) evaluated configuration consists of biocertiX Software, biocertiX App and guidance documentation as indicated in [table 1]. The TOE is delivered together with non-TOE elements indicated in [table 2].

biocertiX Software elements	**Version**
signaturiX Core	2.5.2
Document database	2.1
Database (licenses and configuration)	12.10
signaturiX Admin	2.5.2
biocertiX Application	**Version**
biocertiX App	1.008
Guidance documentation	**Version**
AGD_PRE	0.97
AGD_OPE	1.0

Table. 1. TOE evaluated configuration

Non TOE elements	**Version**
Vault: Secret storage	1.3.2
signaturix-audit	2.5.0
signaturix-starter	2.5.2

Table. 2. Non-TOE elements

1.5.1.1. Delivery of the TOE

The TOE comprises two components:

- The biocertiX Software is delivered in a tamper-protected file. Specifically, the biocertiX Software along with the guidance documentation and non-TOE elements defined in section 1.5.1 are placed in a zip-archive protected by SHA512 digest/hash.
  Link to this archive is sent to the Customer leading to the file distribution system (hosted by the Xtension provider). Access to the file is secured by a password, which is sent by SMS to a designated person (an employee of the Client).
- The biocertiX App. shall be acquired from Google Play Store.

1.5.2. Logical Scope of the TOE

This chapter describes the logical security features offered by the TOE.

1.5.2.1. Secure initialization

When the system is installed, it is configured according to the 'administrator manual' documentation guidelines. The biocertiX App initialization is performed using Knox Manage software, which provides automatic configuration of the public key certificate, security policies and biocertiX App on managed tablets. The corresponding private key is installed on keystore on signaturiX Core during its initialization and password to this keystore is stored in Vault.

The Knox Manage system is the sole entity responsible for the secure initialization and management of keys and certificates (public key involved in encrypting the biometric data in biocertiX App the and certificates for TLS)

used by biocertiX App and it is considered as trusted and secure. The process of initialization and management of keys used by biocertiX App is considered to be secured by Knox Manage system and it is out scope of the TOE evaluated configuration.

The public key used to create the cryptogram of biometrics decryption key is installed in TOE during initialization (it is included in the license file as part of the license).

1.5.2.2. Roles & Available Functions

signaturiX System Administrator

They are users who administer the signaturiX Core software (the part of TOE)). They access through the signaturiX Admin application to perform various TOE-specific operations, e.g. making changes to the TOE configuration, etc. Trusted System Administrators are created in the signaturiX Admin application and each is identified by a login and password and TOTP.

biocertiX User

They are users who can indicate in the signaturiX Core module (using ES) PDF documents for signing. Users are identified by a user ID. In response to the uploaded document (s), the API returns a one-time, time-limited credentials (in the form of a QR/AC code). Using this QR/AC code, the user is given access to the document(s) via the biocertiX App, where he/she can provide handwritten signatures on the document or submit the open document to another person for signature.

1.5.2.3. Secure Administration

Once installed and initialized, the TOE can only be modified (e.g. changes to configuration files, biocertiX system parameters) by authorized System Administrator(s). The System Administrator should be authenticated before any change made to the system. All administrative activities (carried out in the TOE) are recorded by audit.

signaturiX System Administrator is authenticated with a login, a strong password and TOTP before is authorized to perform any actions in the signaturiX Admin application. The signaturiX Admin application must provide strict access control to all system components (System Administrator(s) are first identified and authenticated and then after successful authentication, access to system objects is controlled based on assigned activities according to the procedure guidelines).

1.5.2.4. Audit

All TOE security events are recorded in external database. This event log includes all changes to the TOE, including changes induced by System Administrators, which may affect its security. Inside the signaturiX Audit each entry is protected to prevent changes, entries are protected against deletion (their integrity is ensured). All audit records resulting from the actions of the TOE System Administrators and the execution of requests in the TOE are stored in the signautriX Audit database. The connection between the TOE and signaturiX Audit component is provided via TCP secure protocol TLS 1.3. Access to the Audit Database is possible only after prior authentication. The Audit records do not contain any data that allows the recovery or decryption of confidential biometrics data.

1.5.2.5. Trusted system communication

The TOE implements and enforces the following trusted communication methods and protocols:

- ***External System - TOE***: ES connects to the TOE using HTTPS. The ES sends to the TOE, via the TOE API, a document(s) to be signed. In response, the TOE returns to the ES a unique, one-time credentials in the form of a QR/AC code, based on which the ES user accesses the document(s) in the signaturiX Core module using the biocertiX App.
- ***TOE - Certum SimplySign:*** TOE connects to the SimplySing using HTTPS. A qualified electronic seal and qualified timestamps are provided by the Certum SimplySign.
- ***TOE - signaturiX Audit:*** TOE connects to the signaturiX Audit using HTTPS. A database that stores audit records/logs
- ***Biometrics protection:*** the 'primary biometric data' in a format appropriate for the biometric sampling device used is encrypted by the biocertiX App using a session AES-256 key generated by cryptographically strong random number generator with hardware enhanced entropy provided by Samsung technology that complies with the statistical random number generator tests specified in NIST SP 800-90A and sent in encrypted form to the signaturiX Core. The symmetric key is then encrypted with a public key configured on the Tablet during system initialization. The private key used for session AES key decryption is stored on the biocertiX server in a keystore file protected by a password stored in Vault.

- ***Document sealing and timestamping:*** the product of this encryption is referred to as the cryptogram of biometrics. The AES-256 key is encrypted with a public HSM key (RSA 4096-bit) and the result of this encryption is referred to as the cryptogram of biometrics decryption key. The cryptogram of biometrics and the cryptogram biometrics decryption key are embedded in the signed PDF document. The whole PDF document (with encrypted biometrics embedded) is hashed and the hash value as the data to be signed (DTBS/R) is sealed by using a Certum SimplySign (TTP). Additionally, the electronically sealed PDF document is time-stamped using the qualified timestamp service offered by Certum SimplySign (TTP). Sealing and Timestamping offered by Certum SimpleSign is a service located outside TOE.

Additionally, communication between

- signaturiX Core and biocertiX App, and
- signaturiX Core and ES

is provided using the HTTPS protocol with mutual authentication.

The evaluated TOE configuration covers only the following cipher suites:

- TLS 1.3 suites: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

- TLS 1.2 suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

Other TLS 1.2 cipher suites supported by the TOE are out of the scope of TOE evaluation.

2. Conformance Claims

2.1. CC Conformance Claim

This security target claims conformance to Common Criteria version 3.1 revision 5.

More precisely, this security target claims conformance to:

- Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017 [7].
- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017 [8].
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements; CCMB-2017-04-003, Version 3.1, Revision 5, April 2017 [9].

as follows:

- Part 2 extended; and
- Part 3 conformant.

The following must be considered:

- Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2017-04-004, Version 3.1, Revision 5, April 2017 [10]

The assurance requirement of this security target is **EAL2 Conformant**

2.2. PP Conformance Claim

This ST does not claim conformance to any PP for the TOE.

3. Security Problem Definition

3.1. Assets

The TOE has the following assets, which are to be protected in integrity and some of them in confidentiality as described below. The TOE must ensure that whenever the asset is persisted outside the TOE, the TOE has performed the necessary cryptographic operations to enforce confidentiality and integrity i.e., to detect if an asset has been modified. Access control to TOE assets outside the TOE are to be enforced by the environment.

R.Document - a PDF-formatted document sent by the S.User to the TOE. It is returned to the S.User with an embedded and encrypted scan of a biometric signature. It shall be protected in integrity and authenticity. The authenticity means that the document is processed in reliable manner in TOE, what is confirmed by the qualified electronic seal service. Non-repudiation in the case of S.BioSigner is not managed in TOE and its environment, because it is not the objective of TOE.

R.EmbeddedBioSignature: biometrics binaries scanned by S.BioSigner on the Tablet with the use of S Pen and converted to ISO 197 94-7:2014 compliant format and embedded in PDF document. It shall be protected in integrity and confidentiality.

R.Reference_User_Authentication_Data: the set of data used by TOE to authenticate the S.User. It contains login list used by the TOE to authenticate the S.User (who are authorised to use signaturiX Core). The R.Reference_User_Authentication_Data shall be protected in integrity and confidentiality.

Application Note 1

Any change of login list requires re-initialization of the TOE Core by S.Privileged_User.

R.Reference_Device_Authentication_Data: is the set of data used by the TOE to authenticate the Device (TLS certificate). It contains all the data used by the TOE to authenticate the Device. The R.Reference_Device_Authentication_Data shall be protected in integrity and confidentiality.

Application Note 2

The tablet is bound to the S.User only during the working session on the document(s) (from scanning the QR code or by input AC code via the Tablet keyboard to approving or rejecting the document(s)). It is not necessary to bind the tablet with the S.User longer than it is required to present and sign all the documents indicated by the S.User in S.ES.

R.TSF_Data: is the set of TOE data (configuration) used to operate the TOE. It shall be protected in integrity.

R.Privileged_User: is the set of data that uniquely identifies a S.Privileged_User (System Administrator (s)) within the TOE. It shall be protected in integrity.

R.Reference_Privileged_User_Authentication_Data: is the set of data used by the TOE to authenticate the S.Privileged_User. It shall be protected in integrity and confidentiality.

R.Audit: audit records containing logs of events requiring to be audited. The logs are produced by the TOE and stored externally. The R.Audit shall be protected in integrity and confidentiality.

R.Random random secrets, e.g. AES keys used by the TOE to encrypt biometric sample, QR/AC for Device authentication. It shall be protected in integrity and confidentiality.

3.2. Subjects

This following list of subjects[5] interact with the TOE:

- **S.User** is the individual person who uses the TOE and provides the documents for Signing by the S.BioSigner.
- **S.BioSigner** is the person who use S Pen to sign documents on the Tablet.
- **S.Privileged_User** is only the System Administrator who manages and implements changes to the TOE.
- **S.Attacker** is a human, or process acting on their behalf, located outside the TOE. A S.Attacker is a threat agent (a person with the aim of manipulating user data, or a process acting on their behalf) trying to undermine the TOE security functions, especially to change properties of the maintained assets.
- **S.ES** is a web application that integrates with biocertiX using the API of biocertiX.

3.3. Threats

The following threats are defined for the TOE. An attacker described in each of the threats is a subject that is not authorised for the relevant operation, but may present himself as an unknown user or as one of the other defined subjects.

T.BIOSIGNER_IMPERSONATION: S.Attacker impersonates a S.BioSigner and binds R.EmbeddedBioSignature created by the S.BioSigner with R.Document unaccepted by S.BioSigner. The assets R.Document and R.EmbeddedBioSignature are threatened.
This threat covers the following attacks:

- A S.Attacker may attempt to access to the R.EmbeddedBioSignature provided by a S.BioSigner, which can be replayed to impersonate the S.BioSigner (e.g. signing another document(s) on behalf of the S.BioSigner).
- A S.Attacker may try to record and imitate or generate the biometric characteristic of the S.BioSigner.
- A S.Attacker modifies R.EmbeddedBioSignature during or after creation before its embedding in R.Document.

T. USER_IMPERSONATION

A S.Attacker impersonates S.User. As examples, it could be:

- by transferring wrong R.Reference_User_Authentication_Data to TOE from S.ES.

The assets R.Reference_User_Authentication_Data are threatened

T.EXCESS_AUTHORITY: A S.Attacker may be able to exercise S.Privileged_User authorities to inappropriately manage the TOE. The assets R.Privileged_User, R.Reference_Privileged_User_Authentication_Data and R.TSF_Data are threatened.
T.TSF_COMPROMISE: S.Privileged_User may cause R.TSF_Data (e.g. executable code) to be inappropriately accessed (viewed, modified, or deleted). R.TSF_Data is threatened.
T.UNAUTHORIZED_ACCESS: A S.Attacker may gain access to R.TSF_Data and/or user data for which they are not authorized. All the assets are threatened.
T.UNDETECTED_ACTIONS: A S.Attacker may gain unauthorised access to an unattended S.Privileged_User session, or is positioned on a communication channel or elsewhere in the network infrastructure, causing altered communication between the application software and other endpoints to compromise it. All the assets are threatened.
T.AUDIT: A S.Attacker may be able to cause the lost, destruction of R.Audit or may be able to tamper R.Audit or eavesdrop on R.Audit. The asset R.Audit is threatened.
T.CRYPTO: A S.Attacker can exploit weakness of crypto considering parameters values and known cryptanalysis attacks, thus compromise the cryptographic mechanisms and the data protected by those mechanism. All the assets requiring integrity and/or confidentiality and/or authenticity protection are threatened.

3.4. Relation Between Threats & Assets

This following table provides an overview of the relationships between asset, associated security properties and threats. For details consult the individual threats in the previous sections.

Asset	Security Dimensions	Threats
R.Document	Integrity	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
	Authenticity	T.BIOSIGNER_IMPERSONATION T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.EmbeddedBioSignature	Integrity	T.BIOSIGNER_IMPERSONATION T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
	Confidentiality	T.BIOSIGNER_IMPERSONATION T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.Reference_User_Authentication_ Data	Integrity	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO

		T.USER_IMPERSONATION
	Confidentiality	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO T.USER_IMPERSONATION
R.Reference_Device_Authentication_Data	Integrity	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
	Confidentiality	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.TSF_DATA	Integrity	T.EXCESS_AUTHORITY T.TSF_COMPROMISE T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.Privileged_User	Integrity	T.EXCESS_AUTHORITY T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.Reference_Privileged_Authentication_Data	Integrity	T.EXCESS_AUTHORITY T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
	Confidentiality	T.EXCESS_AUTHORITY T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
R.Audit	Integrity	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.AUDIT T.CRYPTO
	Confidentiality	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.AUDIT T.CRYPTO
R.Random	Integrity	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO
	Confidentiality	T.UNAUTHORIZED_ACCESS T.UNDETECTED_ACTIONS T.CRYPTO

Table 3-1 Relation between Assets, security properties & threats

3.5. Organisational Security Policies

TOE shall comply with the following Organizational Security Policies (OSP) as security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

OSP.ACCOUNTABILITY

The users of the TOE (S.User, S.Privileged_Users) shall be held accountable for security-relevant actions within the system.

OSP.CRYPTOGRAPHY

Approved cryptographic functions shall be used to perform cryptographic operations (e.g. meeting the FIPS or SOGIS requirements when appropriate).

3.6. Assumptions

A. PRIVILEGED_USER: It is assumed that all personnel administering the TOE (S.Privileged_Users) are trusted, competent and possesses the resources and skills required for his/her tasks and is trained to conduct the activities he/she is responsible for.

A.BIOSIGNER: It is assumed that the S.BioSigner is conscious of what he/she is signing and the responsibility resulting from it.

A.SAMPLING_BIOMETRIC_DATA: It is assumed that data sampled by S Pen are reliable and protected before it is transferred to TOE for encryption purposes

A.BIOSIGNER_DEVICE: It is assumed that the device used by the S.User and S.BioSigner to interact with TOE is under the S.User control for the signature operation, e.g. protected against malicious code, protected against physical interception by unauthorized entities. It is assumed that the process of initialization and management of keys and certificates (public key involved in encrypting the biometric data in biocertiX App and certificates for TLS) used by biocertiX App are secure (Knox). It is assumed that the TLS keys (in volatile memory) are secured and protected against any unauthorised access,

A.TRUSTED_USER: It is assumed that the S.User of the biocertiX system is not malicious, and exercises appropriate precautions.

A.ACCESS_PROTECTED: It is assumed that the signaturiX Core part of the TOE operates in a protected environment. Only S.Privileged_Users have access to TOE. The TOE software and hardware environment is installed, configured and managed by S.Privileged_Users in a secure state that mitigates against the specific risks applicable to the deployment environment. It is assumed that the TLS keys (in volatile memory of biocertiX server) are secured and protected against any unauthorised access,

A.AUDIT: It is assumed that any audit generated by the TOE are only handled by authorised personal. The personal that carries these activities should act under established practices.

A.TRUSTED_PKI: It is assumed that TTP (Certum SimplySign) service providers that exchange data with the TOE are trusted. It is assumed that Certum SimplySign supports and enforces at least one the following TLS cipher suites for all communication with the TOE:

- TLS 1.3 suites: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

- TLS 1.2 suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

A.TIME_STAMPS: It is assumed that reliable time stamps for audit logs are provided by biocertiX server operating system's clock configured in such a way that it is regularly synchronized with trusted server based on the NTP protocol.

Application Note 3

The timestamp mentioned above applies to the dating of events log.

A.EXTERNAL_SYSTEM: It is assumed that each S.User has to be authenticated in S.ES before using the biocertiX system and it is assumed that the S.ES provides TLS 1.3 for communication with the TOE.

4. Security Objectives

This section identifies and defines the security objectives for the TOE and its operational environment. These security objectives reflect the stated intent, counter the identified threats, and take into account the assumptions.

4.1. Security objectives for the TOE

The following security objectives describe security functions to be provided by the TOE.

OT.USER_MANAGEMENT

The TOE shall ensure that any modification to R.Reference_User_Authentication_Data is performed under control of the S.Privileged_User. Specifically, only the S.Privileged_User may switch TOE to configuration mode when the TOE accepts requests for modification of Reference_User_Authentication_Data. Moreover, only the S.Privileged_User may block (or unblock) the status of R.Reference_User_Authentication_Data.

OT.ES_AUTHENTICATION/API_SIGNING

The TOE verifies signatures on API (REST/SOAP) messages incoming from the S.ES system and sign API (REST/SOAP) response messages to the S.ES in order to authenticate the S.ES. Additionally, S.ES shall be authenticated according to the TLS 1.3 connection being established.

OT.BIOSIGNATURE_INTEGRITY_CONFIDENTIALITY

The TOE shall ensure that a R.EmbeddedBioSignature can't be modified when using by the TOE and that it is confidential during transmission between biocertiX App and signaturiX Core, between signaturiX Core and SimplySign and between signaturiX Core and S.ES. Before R.EmbeddedBioSignature is created by S.BioSigner

using S_Pen, R.Document shall be authenticated by the authenticated S.User using R.Reference_User_Authentication_Data and the Device shall be authenticated using R.Reference_Device_Authentication_Data.

OT.ROBUST_TOE_ACCESS: The TOE will provide secure mechanisms that control a user's (S.Privileged_User and S.User) logical access[6] to the TOE and explicitly denies access for unauthorized subjects (the TOE must ensure that only identified and authenticated users gain access to protected resources)

OT.SECURE_CHANNEL: The TSF shall communicate externally (with non-TOE entities: SimplySign, ES, Vault,) and internally (between separate subsystems of the TOE software) using a trusted channel that protects the confidentiality and integrity of user data being transmitted.

OT.INTEGRITY: The TOE will provide the capability to perform self-tests and to ensure that the integrity of critical functionality, software and data has been maintained.

OT.CRYPTOGRAPHIC_FUNCTIONS: The TOE shall provide cryptographic functions (i.e., encryption/decryption and digital signature operations) to maintain the confidentiality and allow for detection of modification of TSF data that is transmitted between physically separated portions of the TOE, or stored outside the TOE. The TOE must implement approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification and approved key generation techniques (e.g. meeting the FIPS or SOGIS requirements when appropriate).

OT.MANAGE: The TOE will provide System Administrator interface (for S.Privileged_User) enabling configuration of biocertiX system parameters (block/unblock S.Users, enable/disable config service, configure connection timeout for signing operation, etc.), and restrict remote access to this interface from unauthorized use.

OT.AUDIT: The TOE shall ensure that all users (S.User, S.Privileged_Users) can be held accountable for their security relevant actions by logging security relevant events. All these logs are securely transferred to the external signaturiX Audit (electronically signed with a dedicated private key separate from the key used to sign/seal the signed document).

OT.ROLES The TOE will support users (S.User) roles and separately System Administrator (S.Privileged_User) roles. Users not authorized as System Administrators are not allowed to perform administrative operations.

4.2. Security Objectives for the Operational Environment

OE.RELIABILITY: The authorized S.User is responsible for linking visible R.Document only with the person using S Pen at the moment. The authorized S.User is the sole entity to operate the Tablet.

OE.TIME_STAMPS: The biocertiX server operating system shall provide reliable time stamps. The operating system clock shall be configured in such a way that it is regularly synchronized with a trusted server based on the NTP protocol.

OE.PERSONNEL: Personnel working as TOE System Administrators (S.Privileged_Users) shall be carefully selected and possesses the resources and skills required for his tasks and trained for proper operation of the TOE.

OE.BIOSIGNER: S.BioSigner placing his/her signature in the area pointed on the Tablet screen, shall be conscious of what he/she is signing and the responsibility resulting from it.

OE.SAMPLING_BIOMETRIC_DATA: The data sampled by S Pen are reliable and protected before it is transferred to TOE for encryption purposes.

OE.BIOSIGNER_DEVICE: The device (tablet) containing the biocertiX App and which is used by the S.BioSigner to interact within the TOE shall be protected against malicious code and protected against physical interception by unauthorized entities. The device is considered trusted and it may be used to view the document(s) to be signed. The process of initialization and management of keys and certificates (public key involved in encrypting the biometric data in biocertiX App the and certificates for TLS) used by biocertiX App shall be secured by a trusted Knox Manage system.
The TLS keys (in volatile memory) shall be secured and protected against any unauthorised access.

---

[6] The logical access to the TOE covers access to all TOE parts from outside of the operating system where these TOE parts reside. The local access to the TOE from the operating system where a given TOE part reside is not covered. Any user having access to the OS (where a given TOE part resides) is considered as trusted following the OE.RELIABILITY, OE.PERSONNEL, OE.BIOSIGNER_DEVICE and OE.ENVIRONMENT objectives.

OE.ENVIRONMENT: The signaturiX Core part of the TOE shall operate in a protected environment that limits physical access to the TOE to authorised S.Privileged_Users. The TOE software and hardware environment shall be installed and maintained by S.Privileged_Users in a secure state that mitigates against the specific risks applicable to the deployment environment. The TLS keys (in volatile memory of biocertiX server) shall be secured and protected against any unauthorised access.

OE.AUDIT: Any audit generated by the TOE are only handled by authenticated and authorised personnel protecting R.Audit confidentiality and integrity. The personnel that carries these activities should act under established practices. The signaturiX audit shall verify signatures on audit logs.

OE.TRUSTED_PKI: The TTP service provider (Certum SimplySign) for Qualified Seal and Qualified Timestamp shall be trusted. The Certum SimplySign shall support and enforce at least one the following TLS cipher suites for all communication with the TOE:

- TLS 1.3 suites: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

- TLS 1.2 suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

OE.EXTERNAL_SYSTEM: Before using the biocertiX system, each S.User must first be authenticated in S.ES. The S.ES must provide TLS 1.3 for all communication with the TOE.

4.3. Rationale for the Security Objectives

This section provides a rationale objective that covers each threat, organizational security policy and assumption.

4.3.1. Security Problem Definition & Security Objectives

The following tables map security objectives with the security problem definition.

	OT.USER_MANAGEMENT	OT. BIOSIGNATURE_INTEGRITY_CONFIDENTIALITY	OT.ROBUST_TOE_ACCESS	OT.SECURE_CHANNEL	OT.INTEGRITY	OT.CRYPTOGRAPHIC_FUNCTIONS	OT.MANAGE	OT. AUDIT	OT. ROLES	OT.ES_AUTHENTICATION/API_SIGNING
T.BIOSIGNER_IMPERSONATION		X				x		x		
T.USER_IMPERSONATION	x			x						x
T.EXCESS_AUTHORITY			x						x	
T.TSF_COMPROMISE							x			
T.UNAUTHORIZED_ACCESS			x							
T.UNDETECTED_ACTIONS			x	x	x			x		
T.AUDIT			x	x				x		
T.CRYPTO						x				
Organizational Security Policies										
OSP.ACCOUNTABILITY			x					x		

OSP.CRYPTOGRAPHY						x				

Table 4.1 TOE Security objectives & (threats, Organizational Security Policies)

	OE.RELIABILITY	OE.TIME_STAMPS	OE.PERSONNEL	OE.BIOSIGNER	OE.SAMPLING_BIOMETRIC_DATA	OE.BIOSIGNER_DEVICE	OE.ENVIRONMENT	OE.AUDIT	OE.TRUSTED_PKI	OE.EXTERNAL_SYSTEM
T.BIOSIGNER_IMPERSONATION	x									
T.USER_IMPERSONATION										x
T.EXCESS_AUTHORITY										
T.TSF_COMPROMISE			x				x			
T.UNAUTHORIZED_ACCESS										
T.UNDETECTED_ACTIONS										
T.AUDIT								x		
T.CRYPTO										
Organizational Security Policies										
OSP.ACCOUNTABILITY		x								
OSP.CRYPTOGRAPHY										

Table 4.2 TOE Security Objectives for the Operational Environment & (threats, Organizational Security Policies)

	OE.RELIABILITY	OE.TIME_STAMPS	OE.PERSONNEL	OE.BIOSIGNER	OE.SAMPLING_BIOMETRIC_DATA	OE.BIOSIGNER_DEVICE	OE.ENVIRONMENT	OE.AUDIT	OE.TRUSTED_PKI	OE.EXTERNAL_SYSTEM
A. PRIVILEGED_USER			x							
A.TIME_STAMPS		x								
A.TRUSTED_USER	x									
A.BIOSIGNER				x						
A.SAMPLING_BIOMETRIC_DATA					x					
A.BIOSIGNER_DEVICE						x				
A.ACCESS_PROTECTED							x			
A.AUDIT								x		
A.TRUSTED PKI									x	
A. EXTERNAL_SYSTEM										x

Table 4.3 TOE Assumptions and Security Objectives for the environment

4.3.2. Threats & Objectives

T.BIOSIGNER_IMPERSONATION is covered by **OT.AUDIT** requiring that audit records access attempts to TOE protected resources.

It is also covered by **OT.CRYPTOGRAPHIC_FUNCTIONS** requiring the usage of endorsed algorithms.

It is also covered by **OT.BIOSIGNATURE_INTEGRITY_CONFIDENTIALITY** requiring that the R.EmbeddedBioSignature is protected in integrity and confidentiality during transfer between the parts of the TOE (from biocertiX App to sygnaturiX Core), between the TOE and SimplySign and between sygnaturiX Core and S.ES. Before R.EmbeddedBioSignature is created by S.BioSigner using S_Pen, R.Document shall be authenticated by the authenticated S.User and the Device shall be authenticated using R.Reference_Device_Authentication_Data.

It is also covered by **OE. RELIABILITY** requiring that only the authorized S.User can operate Tablet.

T.USER_IMPERSONATION is covered by **OT.USER_MANAGEMENT** requiring that any modification to R.Reference_User_Authentication_Data is performed under control of the S.Privileged_User (any requested modification requires re-initialisation of the TOE).

It is also covered by **OT.ES_AUTHENTICATION/API_SIGNING** requiring that the S.ES must be authenticated in order for TOE to process the R.Reference_User_Authentication_Data received from the S.ES (TOE verifies signatures on API (REST/SOAP) messages incoming from the S.ES system and sign API (REST/SOAP) response messages to the S.ES, the S.ES is also authenticated with TLS 1.3).

It is also covered by **OT.SECURE_CHANNEL** requiring the protection of the integrity and confidentiality through the use of a trusted channel

It is also covered by **OE.EXTERNAL_SYSTEM** requiring that the S.User must first be authenticated in S.ES before allowing any actions in biocertiX system**.**

T.EXCESS_AUTHORITY is covered by **OT.ROBUST_TOE_ACCESS** requiring authentication of all TOE users (S.Privileged_User and S.User) and it is covered by **OT.ROLES** requiring that the TOE distinguishes administrative roles (S.Privileged_User) that are differentiated from users (S.Users) and authorization of system administrator (S.Privileged_User).

T.TSF_COMPROMISE is covered by **OT.MANAGE** requiring that the TOE will provide System Administrator interface (for S.Privileged_User) enabling configuration of biocertiX system parameters and restrict remote access to this interface from unauthorized use.

It is also covered by OE.ENVIRONMENT requiring that local access to the signaturiX Core part of the TOE is limited to the authorised S.Privileged_Users.

It is also covered by **OE.PERSONNEL** requiring that the **p**ersonnel working as TOE System Administrators (S.Privileged_Users) are carefully selected and possessed the resources and skills required to operate properly in the TOE.

T.UNAUTHORIZED_ACCESS is covered by **OT.ROBUST_TOE_ACCESS** requiring that the TOE provides a mechanism that explicitly denies access for unauthorized subjects .

T.UNDETECTED_ACTIONS is covered by **OT.ROBUST_TOE_ACCESS** requiring that the TOE provides secure mechanisms that control a user's (S.Privileged_User and S.User) logical access to the TOE and explicitly denies access for unauthorized subjects.

It is also covered by **OT.SECURE_CHANNEL** requiring the protection of the integrity and confidentiality through the use of a trusted channel

It is also covered by **OT.INTEGRITY** requiring the integrity of software that is installed onto the system from the network.

It is also covered by **OT.AUDIT** requiring that audit detect access attempts to TOE protected resources.

T.AUDIT is covered by **OT. AUDIT** requiring that audit detect access attempts to TOE protected resources.

It is also covered by **OT.ROBUST_TOE_ACCESS** requiring that the TOE must ensure that only identified and authenticated users (S.Users and S.Privileged_Users) gain access to protected resources

It is also covered by **OE.AUDIT** requiring that any audit generated by the TOE are only handled by authenticated and authorised personnel protecting R.Audit confidentiality and integrity. The personal that carries these activities should acts under established practices.

It is also covered by **OT.AUIDT** and **OE.AUDIT** requiring that TOE signs the logs and the signaturiX audit verifies these signatures. This prevents a S.Attacker from injecting logs to signaturiX audit.

It is also covered by **OT.SECURE_CHANNEL** requiring the protection of the integrity and confidentiality through the use of a trusted channel

T.CRYPTO is covered by **OT.CRYPTOGRAPHIC_FUNCTIONS** requiring that the TOE implements approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification and approved key generation techniques (e.g. meeting the FIPS or SOGIS requirements when appropriate).

4.3.3. Organizational Security Policies & Objectives

OSP.ACCOUNTABILITY is covered by **OT.ROBUST_TOE_ACCESS** requiring that the TOE provides secure mechanisms that control a user's (S.Privileged_User and S.User) logical access to the TOE and explicitly denies access for unauthorized subjects.

It is also covered by **OT. AUDIT** requiring that audit detect access attempts to TOE protected resources.

It is also covered by **OE.TIME_STAMPS** requiring the operational environment to provide a reliable timestamps (The operating system clock is configured in such a way that it is regularly synchronized with a trusted server based on the NTP protocol). The audit mechanism is required to include the current date and time in each audit record.

OSP.CRYPTOGRAPHY is covered by **OT.CRYPTOGRAPHIC_FUNCTIONS** requiring that the TOE implements approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification and approved key generation techniques (e.g. meeting the FIPS or SOGIS requirements when appropriate).

4.3.4. Assumptions & Objectives

A.PRIVILEGED_USER is covered by **OE.PERSONNEL** requiring that the TOE System Administrators (S.Privileged_Users) shall be carefully selected and well trained and non-hostile.

A.BIOSIGNER is covered by **OE.BIOSIGNER** requiring that the S.BioSigner, placing his/her signature in the area pointed on the Tablet screen, is conscious of what he/she is signing and the responsibility resulting from it

A.SAMPLING_BIOMETRIC_DATA is covered by OE. SAMPLING_BIOMETRIC_DATA requiring the data sampled by S Pen are reliable and protected before it is transferred to TOE for encryption purposes.

A.BIOSIGNER_DEVICE is covered by **OE.BIOSIGNER_DEVICE** requiring

- the Signer's device to be protected against malicious code and protected against physical interception by unauthorized entities,
- the process of initialization and management of keys and certificates (public key involved in encrypting the biometric data in biocertiX App the and certificates for TLS) used by biocertiX App to be secured by a trusteed Knox Manage system,
- the TLS keys (in volatile memory) to be secured and protected against any unauthorised access.

A.TRUSTED_USER is covered by **OE.RELIABILITY** requiring that only the authorized S.User can operate the Tablet.

A.ACCESS_PROTECTED is covered by **OE.ENVIRONMENT** requiring the following:

- the TOE be operated in an environment with physical access controls,
- the TOE software and hardware environment to be installed and maintained by S.Privileged_Users in a secure state that mitigates against the specific risks applicable to the deployment environment,
- the TLS keys (in volatile memory of biocertiX server) to be secured and protected against any unauthorised access.

A.AUDIT is covered by **OE.AUDIT** requiring:

- any audit generated by the TOE are only handled by authenticated and authorised personnel protecting R.Audit confidentiality and integrity.
- the personnel that carries these activities should act under established practices.

A.TRUSTED PKI is covered by **OE.TRUSTED_PKI** requiring that the TOE exchanges data with trusted PKI service providers.

A.TIME_STAMPS is covered by **OE.TIME_STAMPS** requiring that the biocertiX server operating system provides reliable time and that the operating system time is regularly synchronized with a trusted NTP server.

A.EXTERNAL_SYSTEM is covered by **OE.EXTERNAL_SYSTEM** requiring that the S.User must first be authenticated in S.ES before allowing any actions in biocertiX system and that the S.ES must provide TLS 1.3 for all communication with the TOE.

5. Extended Components Definition

5.1. Class FCS: Cryptographic Support

5.1.1. Generation of random numbers (FCS_RNG)

Family behaviour

This family defines quality requirements for the generation of random numbers which are intended to be use for cryptographic purposes.

Components levelling and description

Figure 3 shows the component levelling for this family.

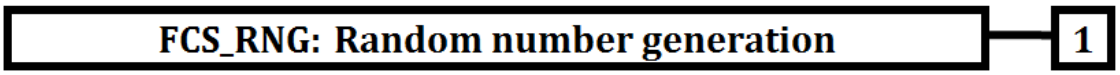

Figure 3 — FCS_RNG: Component levelling

FCS_RNG.1 Random number generation requires that random numbers meet a defined quality metric.

Management of FCS_RNG.1

The following actions can be considered for the management functions in FCS_RNG.1:

a) there are no management activities foreseen.

Audit of FCS_RNG.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP, PP-Module, functional package or ST:

a) there are no actions defined to be auditable.

FCS_RNG.1 Random number generation

Component relationships

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1

The TSF shall provide a [selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic] random number generator that implements: [assignment: list of security capabilities].

FCS_RNG.1.2

The TSF shall provide [selection: bits, octets of bits, numbers [assignment: format of the numbers]] that meet [assignment: a defined quality metric].

6. Security Requirements

6.1. Typographical Conventions

The following conventions are used in the definitions of the SFRs:

- Selections made in this ST are written in **bold text and double underlined**, and the original text is indicated in a footnote.
- Assignments made in this ST are written in **bold italics and underlined**, and the original text is indicated in a footnote.
- Iterations are denoted by a slash "/" and the iteration indicator after the component identifier.

6.2. Subjects, Objects and Operations

This section describes the subjects, objects and operations support by the TOE.

Subject	Description
S.ES	A Web application that integrates with biocertiX using the API of biocertiX. The S.ES sends PDF documents to signaturiX Core and presents to the S.User for signing.
S.User	The individual person who use the TOE through the secure protocol (HTTPS) where they provide the documents for Signing by the S.BioSigner.
S.BioSigner	A person who use S Pen to sign documents on the Tablet
S.Privileged_User	A privileged user (System Administrator) that can administrate the TOE.

Table 6.1. Subjects and description

Object	Description
R.Document	Represents a PDF-formatted document.
R.EmbeddedBioSignature	Biometrics binaries scanned on the Tablet with the use of S Pen and converted to ISO 197 94-7:2014 compliant format and embedded in PDF document
R.Privileged_User	Represents within the TOE a data used to identify a S.Privileged_User (System Administrator) that can administrate the TOE. R.Privileged_User is a login ofS.Privileged_User.
R.Reference_User_Authentication_Data	Data used by the TOE to authenticate a S.User.
R.Reference_User_Status	Status of a S.User in a TOE (blocked or unblocked)

R.Reference_Privileged_User_Authentication_Data	Data used by the TOE to authenticate a Privileged_User
R.Reference_Device_Authentication_Data	Data used by the TOE to authenticate a Device (TLS certificate).
R.TSF_DATA	TOE Core Part (Signatruix Core) Configuration Data. It is the set of TOE data (configuration) used to operate the TOE. The R.TSF_DATA covers among others the following - config service status[7] - connection timeout for signing operation (user session) and for system administrator session.

Table 6.2. Objects and description

Subject	**Operation**	**Object**	**Description**
S.Privileged_User	Block/unblock_User	R.Reference_User_Status	S.Privileged_User, after logging in to SX Admin, can block and unblock using SX Core the list of users (previously uploaded from S.ES).
S.ES	User_maintenance	R.Reference_User_ Authentication_Data	The S.ES updates the R.Reference_User_Authentication _Data in the TOE.
S.Privileged_User	TOE_maintenance	R.TSF_DATA	The signaturiX Core configuration (R.TSF_DATA) can be changed by a S.Privileged_User.
S.User, S.BioSigner, S.ES	Signing	R.Document, R.EmbeddedBioSignature, R.Reference_User_Authentication_Data, R.Reference_Device_Authentication_Data,	The S.User instructs the TOE to perform a signature operation, see FDP_ACC.1.1/Signing for details.

[7] The system administrator must enable the config service in signaturiX Core to enable reception of a list of users from the ES. The system administrator disables the config service once a list of users has been updated by the ES.

The TOE accepts the updates of R.Reference_User_Authentication_Data (user_maintenance operation) only when config service is enabled.

The TOE accepts the signing request (signing operation) from S.User only when config service is disabled.

		R.Reference_User_Status	

Table 6.3. Subject, object, operation and description

6.3. Security Functional Requirements

The individual security functional requirements are specified in the sections below.

6.3.1. Security Audit (FAU)

FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1: The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions

b) All auditable events for the **not specified**[8] level of audit,

c) And:

- ✓ ***Privileged User (S.Privileged_User) management;***
- ✓ ***Privileged User (S.Privileged_User) authentication;***
- ✓ ***User (S.User) management;***
- ✓ ***User (S.User) authentication;***
- ✓ ***Biometrics binaries generation[9];***
- ✓ ***Signing document [10];***
- ✓ ***change of TOE configuration;***
- ✓ ***results of tests related to FPT_TST.1***[11].

Application Note 4

Management of S.Privileged_User shall include all events, which create, modify or delete the R.Privileged_User and R.Reference_Privileged_User_Authentication_Data.

Management of S.User shall include all events, which

- create, modify or delete the R.Reference_User_Authentication_Data,

- change a S.User status - R.Reference_User_Status (i.e. block and unblock a S.User)

TOE configuration shall include all events, which modifies R.TSF_DATA.

Application Note 5

The audit log for the signing operation contain the R.EmbeddedBioSignature in encrypted form.

FAU_GEN.1.2: The TSF shall record within each audit record at least the following information:

a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

[8] [selection, choose one of: minimum, basic, detailed, not specified]

[9] The biometrics binaries are generated in SignaturiX Core in accordance with the requirements of ISO/IEC 19794-7:2021 — Information technology — Biometric data interchange formats — Part 7: Signature/sign time series data.

[10] Document signing relates to action of embedding qualified seal and qualified timestamp into R.Document.

[11] [assignment: other specifically defined auditable events]

b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, ***none***[12]**.**

Application Note 6

Audit trail shall not include any data which allow to retrieve biometric data like (biometrics binary data etc.).

6.3.2. Cryptographic Support (FCS)

FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data - Cryptographic key generation

FCS_CKM.1.1/Generation_of_encryption_keys_for_biometric_data: The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm ***defined in Table6.4***[13] and specified cryptographic key sizes ***defined in Table6.4*** [14] that meet the following: ***defined in Table6.4*** [15].

Table 6.4. Key Generation Table

Key generation algorithm	Key size(s)	Standard
NativePRNG[16]	256 bits	none

Application Note 7

This SFR covers the generation of AES encryption keys for biometric data encryption in biometriX App and signaturiX Core. For this purpose:

- An AES key is generated in biometriX App using a cryptographically strong random number generator [2, 3, 4] with hardware enhanced entropy provided by Samsung technology that complies with the statistical random number generator tests specified in NIST SP 800-90A [5].
- An AES key is generated in signaturiX Core using SecureRandom instance (https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/security/SecureRandom.html)

FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys - Cryptographic key generation

FCS_CKM.1.1/Generation_of_the_client_and_server_encryption_keys: The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm ***defined in Table*****6.5**[17] and specified cryptographic key sizes ***defined in Table*** **6.5**[18] that meet the following: ***defined in Table*** **6.5**[19].

Table 6.5. Key Generation Table

Key generation algorithm	Key size(s)	Standard
EC Diffie-Hellman (ECDHE)	128 bits, 256 bits	RFC8446[11], RFC8447[22]

Application Note 8

This SFR covers the generation of the client and server symmetric encryption keys and client and server MAC keys for the TLS 1.3 protocol (according one of the following cipher suites: TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256 and TLS_AES_128_GCM_SHA256) and TLS 1.2 protocol (according one of the following cipher suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 and TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384). These keys are generated during handshake protocol of TLS 1.2 and TLS 1.3 according EC Diffie Hellman (ECDHE) algorithm.

[12] [assignment: other audit relevant information]
[13] [assignment: cryptographic key generation algorithm]
[14] [assignment: cryptographic key sizes]
[15] [assignment: list of standards]
[16] AES key generation for biometric data encryption uses secure generator from JAVA 11 software „java.security.SecureRandom (https://docs.oracle.com/en/java/javase/11/docs/specs/security/standard-names.html#securerandom-number-generation-algorithms)
[17] [assignment: cryptographic key generation algorithm]
[18] [assignment: cryptographic key sizes]
[19] [assignment: list of standards]

FCS_CKM.2/Symmetric_key_transport_protocol - Cryptographic key distribution

FCS_CKM.2.1/Symmetric_key_transport_protocol: The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method ***defined in Table* 6.6**[20] that meets the following: ***defined in Table* 6.6**[21]**.**

Table 6.6. Key Distribution Table

Cryptographic Key distribution	Standard
Encryption of data encryption key with RSA 4096 bits (key wrapping)	PKCS#1v2.2 [12], RSAES-OAEP [13]

Application Note 9

This SFR covers the symmetric key transport protocol from biocertiX App to signaturiX Core, i.e. Encryption of AES-256 biometrics decryption key. It also covers the symmetric key transport protocol from signaturiX Core to external word, i.e. encryption of AES-256 standardized biometrics decryptions key.

FCS_CKM.2/Key_establishment_of_session_keys - Cryptographic key distribution

FCS_CKM.2.1/Key_establishment_of_session_keys: The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method ***defined in Table 6.7***[22] that meets the following: ***defined in Table 6.7***[23]**.**

Table 6.7. Key Establishment Table

Cryptographic key establishment method	Standard
EC – Diffie-Hellman	RFC8446 [11], RFC8447[22]

Application Note 10

This SFR covers the key establishment of session keys in the TLS protocol.

FCS_CKM.2/Exchange_of_X.509_certificates - Cryptographic key distribution

FCS_CKM.2.1/Exchange_of_X.509_certificates: The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method ***defined in Table 6.8***[24] that meets the following: ***defined in Table 6.8***[25]**.**

Table 6.8. Certificate exchange method Table

Cryptographic certificate exchange method	Standard
Exchange of X509 v3 certificates	RFC8446 [11], RFC8447[22] and RFC5280 [14]

Application Note 11

This SFR covers the exchange of client and server X.509 v3 certificates in the TLS protocol.

FCS_CKM.4 Cryptographic key destruction

[20] [assignment: cryptographic key distribution method]
[21] [assignment: list of standards]
[22] [assignment: cryptographic key distribution method]
[23] [assignment: list of standards]
[24] [assignment: cryptographic key distribution method]
[25] [assignment: list of standards]

FCS_CKM.4.1: The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method ***Secure erasing-zeroization***[26] that meets the following: ***no standards***[27]

Application Note 12

Symmetric keys used for biometric data encryption and decryption are zeroized automatically when they are no longer needed, ie. after the end of biosigning process. Zeroization is done by overwriting once with zeroes.

FCS_COP.1/Biometric_data_encryption_in_App - Cryptographic operation

FCS_COP.1.1/Biometric_data_encryption_in_App: The TSF shall perform ***the operation defined in table 6.97***[28] in accordance with a specified cryptographic algorithm ***defined in table 6.9***[29] and cryptographic key sizes ***defined in table 6.9***[30] that meet the following: ***defined in table6.9***[31].

Table 6.9. Cryptographic operations for biometric data protection

Cryptographic operations	Algorithm	Key size(s)	Standards
Symmetric encryption and decryption	AES/CBC/PKCS5Padding	256 bits	FIPS197 [15]. SP800-38A [16]

Application Note 13

This SFR covers biometric data encryption in biometriX App and decryption in signaturiX Core.

FCS_COP.1/Biometric_data_encryption_in_Core- Cryptographic operation

FCS_COP.1.1/Biometric_data_encryption_in_Core: The TSF shall perform ***the operation defined in table 6.10***[32] in accordance with a specified cryptographic algorithm ***defined in table 6.10***[33] and cryptographic key sizes ***defined in table 6.10***[34] that meet the following: ***defined in table6.10***[35].

Table 6.10. Cryptographic operations for biometric data protection

Cryptographic operations	Algorithm	Key size(s)	Standards
Symmetric encryption	AES/CTR/NoPadding	256 bits	FIPS197 [15], SP800-38A [16]

Application Note 14

This SFR covers biometric data encryption in signaturiX Core.

FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations - Cryptographic operation

FCS_COP.1.1/TLS_protocol_AES_cryptographic_symmetric_operations: The TSF shall perform ***the operation defined in table 6.11***[36] in accordance with a specified cryptographic algorithm ***defined in table 6.11***[37] and cryptographic key sizes ***defined in table 6.11***[38] that meet the following: ***defined in table 6.11***[39].

[26] [assignment: cryptographic key destruction method]
[27] [assignment: list of standards]
[28] [assignment: list of cryptographic operations]
[29] [assignment: cryptographic algorithm]
[30] [assignment: cryptographic key sizes]
[31] [assignment: list of standards]
[32] [assignment: list of cryptographic operations]
[33] [assignment: cryptographic algorithm]
[34] [assignment: cryptographic key sizes]
[35] [assignment: list of standards]
[36] [assignment: list of cryptographic operations]
[37] [assignment: cryptographic algorithm]
[38] [assignment: cryptographic key sizes]
[39] [assignment: list of standards]

Table 6.11. Symmetric cryptographic operations in the TLS protocol

Cryptographic operations	Algorithm	Key size(s)	Standards
Symmetric encryption and decryption, MAC calculation	AES in GCM mode	128 bits, 256 bits	FIPS197[15] and SP800-38D [17]

FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations - Cryptographic operation

FCS_COP.1.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations: The TSF shall perform ***the operation defined in table 6.12***[40] in accordance with a specified cryptographic algorithm ***defined in table 6.11***[41] and cryptographic key sizes ***defined in table 6.11***[42] that meet the following: ***defined in table 6.11***[43].

Table 6.11. Symmetric cryptographic operations in the TLS protocol

Cryptographic operations	Algorithm	Key size(s)	Standards
Symmetric encryption and decryption, MAC calculation	CHACHA20_POLY1305	256 bits	RFC8439 [19]

FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations - Cryptographic operation

FCS_COP.1.1/TLS_protocol_cryptographic_asymmetric_operations: The TSF shall perform ***the operation defined in table 6.12***[44] in accordance with a specified cryptographic algorithm ***defined in table 6.12***[45] and cryptographic key sizes ***defined in table 6.12***[46] that meet the following: ***defined in table 6.12***[47].

Table 6.12. Asymmetric cryptographic operations in the TLS protocol

Cryptographic operations	Algorithm	Key size(s)	Standards
Asymmetric encryption and decryption	RSA	4096 bits	PKCS#1 v2.2 (RSAES-OAEP) RFC8446 [11]

Application Note 15

These SFRs covers asymmetric encryption and decryption operations used by the TLS protocol.

FCS_COP.1/Digital_signature_RSA - Cryptographic operation

FCS_COP.1.1/Digital_signature_RSA: The TSF shall perform ***the operation defined in table 6.13***[48] in accordance with a specified cryptographic algorithm ***defined in table 6.13***[49] and cryptographic key sizes ***defined in table 6.13***[50] that meet the following: ***defined in table 6.13***[51].

[40] [assignment: list of cryptographic operations]
[41] [assignment: cryptographic algorithm]
[42] [assignment: cryptographic key sizes]
[43] [assignment: list of standards]
[44] [assignment: list of cryptographic operations]
[45] [assignment: cryptographic algorithm]
[46] [assignment: cryptographic key sizes]
[47] [assignment: list of standards]
[48] [assignment: list of cryptographic operations]
[49] [assignment: cryptographic algorithm]
[50] [assignment: cryptographic key sizes]
[51] [assignment: list of standards]

Table 6.13 Cryptographic signatures for REST and SOAP API and API of signaturiX Audit and TLS

Cryptographic operations	Algorithm	Key size(s)	Standards
Digital signature generation and verification	RSA + SHA-256	4096 bits	PKCS#1 v2.2 (RSASSA-PSS) RFC8446 [11]

Application Note 16

This SFR covers cryptographic operations used by signing of REST & SOAP API. The API is secured with asymmetric cryptography in such a way that requests are signed with a dedicated private key and verified with a dedicated public key (independently of the encryption provided by HTTPS). This SFR also covers cryptographic operations used by signing audit records that are send to signaturiX Audit using the REST API of signaturiX Audit. The API of signaturiX Audit is secured with asymmetric cryptography in such a way that requests are signed with a dedicated private key and verified with a dedicated public key (independently of the encryption provided by HTTPS).

This SFR also covers digital signature generation and verification used by the TLS 1.2 and TLS 1.3 protocols.

FCS_COP.1/ Digital_signature_ECDSA - Cryptographic operation

FCS_COP.1.1/Digital_signature_ECDSA: The TSF shall perform ***the operation defined in table 6.14***[52] in accordance with a specified cryptographic algorithm ***defined in table 6.14***[53] and cryptographic key sizes ***defined in table 6.14***[54] that meet the following: ***defined in table 6.14***[55].

Table 6.14 Cryptographic signatures for TLS 1.2

Cryptographic operations	Algorithm	Key size(s)	Standards
Digital signature generation and verification	ECDSA	256 bits	FIPS 186-4[23], RFC5246 [24], RFC8447[22]

Application Note 17

This SFR covers digital signature generation and verification used by the TLS 1.2 protocol.

FCS_COP.1/hashing_operations - Cryptographic operation

FCS_COP.1.1/hashing_operations: The TSF shall perform ***the operation defined in table 6.15***[56] in accordance with a specified cryptographic algorithm ***defined in table 6.15***[57] and cryptographic key sizes ***defined in table 6.15***[58] that meet the following: ***defined in table 6.15***[59].

Table 6.15. Cryptographic hashing

Cryptographic operations	Algorithm	Key size(s)	Standards
Hashing for the purpose of integrity control	SHS (SHA-256, SHA-384, SHA-512)	n/a	FIPS PUB 180-4 [18]

Application Note 18

[52] [assignment: list of cryptographic operations]
[53] [assignment: cryptographic algorithm]
[54] [assignment: cryptographic key sizes]
[55] [assignment: list of standards]
[56] [assignment: list of cryptographic operations]
[57] [assignment: cryptographic algorithm]
[58] [assignment: cryptographic key sizes]
[59] [assignment: list of standards]

This SFR covers cryptographic hashing operations. SHA-256 is used for the purpose of DTBS/R preparation for electronic sealing of PDF document by Certum SimplySign. For the purpose of time stamping by Certum SimplySign all mentioned in Table 6.15 variants of SHS are used. Inside TOE SHA-256 is used for integrity checks. In TLS protocol SHA-384 and SHA-256 are used as specified in cipher suites for TLS 1.2 and TLS 1.3.

FCS_RNG.1 Random number generation

FCS_RNG.1.1 The TSF shall provide a **deterministic**[60] random number generator that implements ***forward and backward secrecy***[61].

FCS_RNG.1.2 The TSF shall provide **bits**[62] that meet ***Recommendation for Random Number Generation Using Deterministic Random Bit Generators NIST SP 800-90 Rev.1 [20] and A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications NIST SP 800-22 Rev.1a [21]***[63].

Application Note 19

This SFR covers generation of QR/AC code. The RNG/PRNG should be resistant to manipulation or analysis of its sources, or any attempts to predictably influence its states by applying checksums over the sources.

6.3.3. User Data Protection (FDP)

FDP_ACC.1/User_Maintenance- Subset access control

FDP_ACC.1.1/User_Maintenance: The TSF shall enforce the ***User Maintenance SFP***[64] on:

Subjects: S.ES

Objects: The security attributes R.Reference_User_Authentication_Data of S.User

Operations: User_Maintenance:

The S.ES updates R.Reference_User_Authentication_Data of S.User in the TOE. [65]

FDP_ACF.1/User_Maintenance - Security attribute based access control

FDP_ACF.1.1/User_Maintenance: The TSF shall enforce the ***User_Maintenance SFP***[66] to objects based on the following:

1) ***Whether the subject is an authenticated S.ES***[67].

FDP_ACF.1.2/User_Maintenance: The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

1) ***None*** [68].

FDP_ACF.1.3/User_Maintenance: The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

1) ***None***[69].

[60] [selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic]
[61] [assignment: list of security capabilities]
[62] [selection: bits, octets of bits, numbers [assignment: format of the numbers]]
[63] [assignment: a defined quality metric]
[64] [assignment: access control SFP]
[65] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[66] [assignment: access control SFP]
[67] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[68] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[69] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]

FDP_ACF.1.4/User_Maintenance: The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

1) ***The config mode is disabled in the TOE.***[70]

FDP_ACC.1/TOE_Maintenance - Subset access control

FDP_ACC.1.1/TOE_Maintenance: The TSF shall enforce the ***TOE Maintenance SFP***[71] on:

Subjects: S.Privileged_User

Objects: The security attribute R.TSF_Data of the TOE

Operations: TOE_Maintenance:

The Privilegded User manages R.TSF_DATA[72]

FDP_ACF.1/TOE_Maintenance - Security attribute based access control

FDP_ACF.1.1/TOE_Maintenance: The TSF shall enforce the ***TOE maintenance SFP***[73] to objects based on the following:

2) Whether the subject is an authenticated[74] ***S.Privileged_User***[75].

FDP_ACF.1.2/TOE_Maintenance: The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

2) None[76].

FDP_ACF.1.3/TOE_Maintenance: The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

2) ***None***[77].

FDP_ACF.1.4/TOE_Maintenance: The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

2) ***None***[78].

FDP_ACC.1/Block-Unblock_User - Subset access control

FDP_ACC.1.1/Block/Unblock_User: The TSF shall enforce the ***Block-Unblock User SFP***[79] on:

Subjects: S.Privileged_User

Objects: The security attribute R.Reference_User_Status of User

Operations: **Block/Unblock_User:**

---

[70] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[71] [assignment: access control SFP]
[72] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[73] [assignment: access control SFP]
[74] The authentication performed by the TSF applies only remote access of S.Privileged_User to the TOE (from outside of biocertiX server OS). See OT.ROBUST_TOE_ACCESS for more information.
[75] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[76] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[77] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]
[78] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[79] [assignment: access control SFP]

The Privilegded User changes the status of User (from blocked to unblocked or from unblocked to blocked). [80]

FDP_ACF.1/Block-Unblock_User - Security attribute based access control

FDP_ACF.1.1/Block-Unblock_User: The TSF shall enforce the ***Block-Unblock User SFP***[81] to objects based on the following:

3) ***Whether the subject is an authenticated S.Privileged_User***[82].

FDP_ACF.1.2/Block-Unblock_User: The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

3) ***None***[83].

FDP_ACF.1.3/Block-Unblock_User: The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

3) ***None***[84].

FDP_ACF.1.4/Block-Unblock_User: The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

3) ***None***[85].

FDP_ACC.1/Signing - Subset access control

FDP_ACC.1.1/Signing: The TSF shall enforce the ***Signing SFP***[86] on:

Subjects: S.User, S.BioSigner, S.ES

Objects: R.Document, R.EmbeddedBioSignature,
R.Reference_User_Authentication_Data,
R.Reference_Device_Authentication_Data
R.Reference_User_Status

Operations: Signing:

The S.User instructs the TOE to perform a signature operation containing the following steps:

S.User sends R.Document to TOE via authenticated S.ES

TOE verifies R.Reference_User_Authentication_Data and R.Reference_User_Status

S.User authenticates the R.document by inputing (QR/AC code) on the device. The device itself is authenticated using the R.Reference_Device_Authentication_Data

The TOE displays R.Document

S.BioSigner uses S pen to sign document.

TOE gets biometrics binaries and performs all necessary steps to embede qualified seal an qualified timestamp into R.Document resulting in R.Document with R.EmbeddedBioSignature

[80] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[81] [assignment: access control SFP]
[82] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[83] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[84] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]
[85] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[86] [assignment: access control SFP]

The S.User accepts or rejects R.Document with R.EmbeddedBioSignature

The TOE deactivates[87] the R.Reference_Device_Authentication_Data when the signature operation is completed[88].

FDP_ACF.1/Signing - Security attribute based access control

FDP_ACF.1.1/ Signing: The TSF shall enforce the ***Signing SFP***[89] to objects based on the following:

1) ***Whether the S.User is authorized to create a signature (TOE must positively verify R.Reference_User_Authentication_Data)***[90].

FDP_ACF.1.2/Signing: The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

1) ***None***[91].

FDP_ACF.1.3/Signing: The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

1) ***The device and document must be positively authenticated using R.Reference_Device_Authentication_Data (QR/AC code)***[92].

FDP_ACF.1.4/Signing: The TSF shall explicitly deny access of subjects to objects based on the following additional rule:

1) ***The config mode is enabled in the TOE,***
2) ***The R.Reference_User_Status is blocked*** [93]

6.3.4. Identification and Authentication (FIA)

FIA_ATD.1 User attribute definition

FIA_ATD.1.1: The TSF shall maintain the following list of security attributes belonging to individual users: ***the security attribute as defined in FIA_USB.1/User and FIA_USB.1.1/Privileged_User*** [94].

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.

Application Note 20

- The S.User is authenticated using R.Reference_User_Authentication_Data which is assumed to be confidential.
- The S.Privileged_User (System Administrator) is authenticated in accordance with FIA_UAU.5/Privileged_User.
- The S.ES (External System) is authenticated in accordance with FIA_UAU.5/ES.

FIA_UAU.5/Privileged_User - Multiple authentication mechanisms

[87] Deactivating R.Reference_Device_Authentication_Data implies termination of the S.User's session working with the document package in question. That is, the deletion of the S.User's session cookies on the biocertiX App side and the cancellation of their session on the signaturiX Core side.
[88] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[89] [assignment: access control SFP]
[90] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[91] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[92] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]
[93] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[94] [assignment: list of security attributes]

FIA_UAU.5.1/Privileged_User: The TSF shall provide ***login, password and TOTP verification***[95] to support user authentication.

FIA_UAU.5.2/Privileged_User: The TSF shall authenticate any user's claimed identity according to the:

1. ***S.Privileged_User enters login and password that are verified by the TOE***
2. ***If the login and password are correct, S.Privileged_User enters "Authenticator" type TOTP code that is verified by the TOE*** [96].

Application Note 21

When a S.Privileged_User logs into SX Admin for the first time (enters his or her username and password), he or she is shown a QR code which he or she must scan with an 'Authenticator' type application, such as Google Authenticator - from then on, this application generates TOTP codes for him or her every time wants to authenticate again.

This SFR refers to the Privileged User role. This SFR applies to remote access to SX Admin (that is from outside the biocertiX servers OS).

FIA_UAU.5/ES - Multiple authentication mechanisms

FIA_UAU.5.1/ES: The TSF shall provide ***TLS certificate and API message verification***[97] to support user authentication.

FIA_UAU.5.2/ES: The TSF shall authenticate any user's claimed identity according to the:

1. ***ES is authenticated using TLS certificate,***
2. ***If the TLS certificate is correct, each API message signature is verified***[98].

Application Note 22

S.ES (External System) is authenticated at the TLS level with a certificate, in addition, each API message with is authenticated using API public/private keys.

This SFR refers to the External System (S.ES).

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.

Application Note 23

The S.User is identified by the entry in the login list. The S.Privileged_User is identified by login and the S.ES (External System) is identified when the TLS connection is established.

FIA_USB.1/User - User-subject binding

FIA_USB.1.1/User: The TSF shall associate the following user security attributes with subjects acting on the behalf of that user:

1) ***security role: user,***
2) ***R.Reference_User_Authentication_Data (identification_and_authentication_data),***
3) ***R.Reference_User_Status (status)***[99]***.***

[95] [assignment: list of multiple authentication mechanisms]
[96] [assignment: rules describing how the multiple authentication mechanisms provide authentication]
[97] [assignment: list of multiple authentication mechanisms]
[98] [assignment: rules describing how the multiple authentication mechanisms provide authentication]
[99] [assignment: list of user security attributes]

FIA_USB.1.2/User: The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users:

1) ***the initial association for security role is user,***
2) ***the initial association for R.Reference_User_Authentication_Data follows FDP_ACC.1.1/User_Maintenance and FDP_ACF.1.1/User_Maintenance***
3) ***the initial association for R.Reference_User_Status is unblocked*** [100]***.***

FIA_USB.1.3/User: The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

1) ***changes to R.Reference_User_Authentication_Data follow FDP_ACC.1.1/User_Maintenance and FDP_ACF.1.1/User_Maintenance***
2) ***changes to R.Reference_User_Status follow FDP_ACC.1.1/ Block-Unblock_User and FDP_ACF.1.1/ Block-Unblock_User*** [101]**.**

FIA_USB.1/Privileged_User - User-subject binding

FIA_USB.1.1/Privileged_User: The TSF shall associate the following user security attributes with subjects acting on the behalf of that user:

1) ***security role: privileged_user;***
2) ***R.Privileged_User (identification_data)***
3) ***R.Reference_Privileged_User_Authentication_Data (authentication_data)***[102]

FIA_USB.1.2/Privileged_User: The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users:

1) ***the initial association for security role is privileged_user,***
2) ***the initial association for R.Privileged_User is to be configured during TOE installation,***
3) ***the initial association for R.Reference_Privileged_User_Authentication_Data is to be configured during TOE installation***[103]***.***

FIA_USB.1.3/Privileged_User: The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

1) ***only an authenticated S.Privileged_User may change R.Privileged_User and R.Reference_Privileged_User_Authentication_Data***[104]**.**

6.3.5. Security Management (FMT)

FMT_MSA.1/User_Management - Management of security attributes

FMT_MSA.1.1/User_Management The TSF shall enforce the ***User_Maintenance_SFP and Block-Unblock_User_SFP***[105] to restrict the ability to **modify**[106] user security attributes ***R.Reference_User_Authentication_Data and R.Reference_User_Status*** [107] to **S.ES** ***and S.Privileged_User*** [108].

FMT_MSA.1/Signing - Management of security attributes

[100] [assignment: rules for the initial association of attributes].
[101] [assignment: rules for the changing of attributes]
[102] [assignment: list of user security attributes]
[103] [assignment: rules for the initial association of attributes].
[104] [assignment: rules for the changing of attributes]
[105] [assignment: access control SFP(s), information flow control SFP(s)]
[106] [selection: change, default, query, modify, delete [assignment: other operations]**]**
[107] [assignment: list of security attributes**]**
[108] [assignment: the authorised identified roles:]

FMT_MSA.1.1/Signing The TSF shall enforce the ***Signing SFP*** [109] to restrict the ability to **create**[110] user security attributes ***R.Document and R.EmbeddedBioSignature*** [111] to ***S.User*** [112].

FMT_MSA.1/TOE_Management - Management of security attributes

FMT_MSA.1.1/TOE_Management The TSF shall enforce the ***TOE Maintenance SFP*** [113] to restrict the ability to **modify**[114] user security attributes ***R.TSF_DATA*** [115] to ***S.Privileged_User*** [116].

FMT_MSA.3/User - Static attribute initialisation

FMT_MSA.3.1 The TSF shall enforce the ***User Maintenance SFP***[117] to provide **restrictive**[118] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the ***S.ES***[119] to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3/TOE - Static attribute initialisation

FMT_MSA.3.1 The TSF shall enforce the ***TOE Maintenance SFP***[120] to provide **restrictive**[121] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the ***S.Privileged_User***[122] to specify alternative initial values to override the default values when an object or information is created.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1: The TSF shall be capable of performing the following management functions:

1) ***User Maitenance as defined in User Maintenance SFP,***
2) ***TOE Maintenance as defined in TOE Maintenance SFP,***
3) ***block-unblock user as defined in Block-Unblock User SFP*** [123].

FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1: The TSF shall maintain the roles ***User, Privileged User, ES***[124].

FMT_SMR.2.2: The TSF shall be able to associate users with roles.

---

[109] [assignment: access control SFP(s), information flow control SFP(s)]
[110] [selection: change, default, query, modify, delete [assignment: other operations]**]**
[111] [assignment: list of security attributes**]**
[112] [assignment: the authorised identified roles:]
[113] [assignment: access control SFP(s), information flow control SFP(s)]
[114] [selection: change, default, query, modify, delete [assignment: other operations]**]**
[115] [assignment: list of security attributes**]**
[116] [assignment: the authorised identified roles:]
[117] [assignment: access control SFP, information flow control SFP]
[118] [selection, choose one of: restrictive, permissive, [assignment: other property]]
[119] [assignment: the authorised identified roles]
[120] [assignment: access control SFP, information flow control SFP]
[121] [selection, choose one of: restrictive, permissive, [assignment: other property]]
[122] [assignment: the authorised identified roles]
[123] [assignment: list of management functions to be provided by the TSF]
[124] [assignment: authorized identified roles]

FMT_SMR.2.3: The TSF shall ensure that the conditions ***an entity cannot be assigned more than one role***[125] are satisfied.

6.3.6. TSF physical protection (FPT)

FPT_ITT.1 Basic internal TSF data transfer protection

FPT_ITT.1.1: The TSF shall protect TSF data from **disclosure, modification** [126] when it is transmitted between separate parts of the TOE.

Application Note 24

It is necessary to satisfy this objective because it ensures that TSF data is protected when it is transmitted between components of the TOE. The communication between the App on the tablet and the Core on the server is encrypted to prevent the disclosure and modification of information. This data would include the biometric data as it leaves the capture device, or as it is transmitted between other parts of the TOE. It is realized with TLS 1.3.

FPT_STM.1 Reliable time stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

Application Note 25

It is ensured by operating system on the server that it synchronizes its system clock with trusted NTP server. The TOE trusts the hosts system time, but also compares the system time with trusted NTP server (during biocertiX software start-up) to check it. This time stamp source is required for audit purposes.

FPT_TST.1 TSF testing

FPT_TST.1.1 The TSF shall run a suite of self-tests **during initial start-up**[127] to demonstrate the correct operation of ***time stamp service for audit (FPT_STM.1 Reliable time stamps)***[128].

FPT_TST.1.2 The TSF shall provide authorised users with the capability to verify the integrity of **TSF data**[129].

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of **TSF**[130].

6.3.7. TOE Access (FTA)

FTA_SSL.3 TSF-initiated termination

FTA_SSL.3.1 The TSF shall terminate an interactive session after a ***time configured by the R.Privileged_User in system parameters (R.TSF.Data)***[131].

Application Note 26

The TOE should provide an ability to terminate a session (administrator session and user session) after defined period (the session time is set to 30 min. by the S.Privileged_User using a configuration file). Any session time change requires a restart of the TOE.

6.3.8. Trusted Paths/Channels (FTP)

FTP_ITC.1/ES_to_TOE - Inter-TSF trusted channel

[125] [assignment: conditions for the different roles]
[126] [selection: disclosure, modification]
[127] [selection: during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self test should occur]]
[128] [selection: [assignment: parts of TSF], the TSF]
[129] [selection: [assignment: parts of TSF data], TSF data]
[130] [selection: [assignment: parts of TSF], TSF]
[131] [assignment: time interval of user inactivity]

FTP_ITC.1.1 The TSF shall provide a communication path between itself and another trusted IT product that is logically distinct from other communication paths and provides ensured authentication and identification of its end points and protection of the communicated data from modification or disclosure

FTP_ITC.1.2: The TSF shall permit the **another trusted IT product**[132], to initiate communication via the trusted channel.

FTP_ITC.1.3: The TSF shall initiate communication via the trusted channel for ***transfer PDF documents to signaturiX Core to bind with biosignatures***[133].

Application Note 27

The FTP_ITC.1/ES_to_TOE is realized with the TLS 1.3.

FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign - Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall provide a communication path between itself and another trusted IT product that is logically distinct from other communication paths and provides ensured authentication and identification of its end points and protection of the communicated data from modification or disclosure

FTP_ITC.1.2: The TSF shall permit the **TSF**[134], to initiate communication via the trusted channel.

FTP_ITC.1.3: The TSF shall initiate communication via the trusted channel for ***sending notifications to S.ES, sending audit logs to signaturiX audit, sending credentials to vault, sealing and embedding reliable trusted time stamps in PDF documents using SimplySign***[135].

Application Note 28

The FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign is realized with:

- TLS 1.3 in regard to S.ES and Audit

- TLS 1.2 and TLS 1.3 in regard so SimplySign (depending on the option supported by SimplySign)

- TLS 1.2 in regard to vault.

6.4. Security Assurance Requirements

The security assurance requirement level is EAL2. The assurance components are identified in the table below.

Assurance Class	Assurance Components
Development (ADV)	Security architecture description (ADV_ARC.1)
	Security-enforcing functional specification (ADV_FSP.2)
	Basic design (ADV_TDS.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life-cycle support (ALC)	Use of a CM system (ALC_CMC.2)
	Parts of the TOE CM coverage (ALC_CMS.2)

[132] [selection: the TSF, another trusted IT product]
[133] [assignment: list of functions for which a trusted channel is required]
[134] [selection: the TSF, another trusted IT product]
[135] [assignment: list of functions for which a trusted channel is required]

	Delivery procedures (ALC_DEL.1)
Security Target evaluation (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives (ASE_OBJ.2)
	Derived security requirements (ASE_REQ.2)
	Security problem definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Tests (ATE)	Evidence of coverage (ATE_COV.1)
	Functional testing (ATE_FUN.1)
	Independent testing - sample (ATE_IND.2)
Vulnerability assessment (AVA)	Vulnerability analysis (AVA_VAN.2)

Table 6.16. Security Assurance Requirements

7. Rationale

7.1. Security Requirements Rationale - Coverage

The following table is used to demonstrate that every SFR is used to cover an objective and that every objective is covered by an SFR

	OT.USER_MANAGEMENT	OT.ES_AUTHENTICATION/API_SIGNING	OT. BIOSIGNATURE_INTEGRITY_CONFIDENTIALITY	OT.ROBUST_TOE_ACCESS	OT.SECURE_CHANNEL	OT.INTEGRITY	OT.CRYPTOGRAPHIC_FUNCTIONS	OT.MANAGE	OT.AUDIT	OT.ROLES
Security Audit										
FAU_GEN.1						X			X	
Cryptographic Support										
FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data			X				X			
FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys		X			X		X			

FCS_CKM.2/Symmetric_key_transport_protocol			X				X			
FCS_CKM.2/Key_establishment_of_session_keys		X			X		X			
FCS_CKM.2/Exchange_of_X.509_certificates		X			X		X			
FCS_CKM.4			X				X			
FCS_COP.1/Biometric_data_encryption_in_App			X				X			
FCS_COP.1/Biometric_data_encryption_in_Core			X				X			
FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations		X			X		X			
FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations		X			X		X			
FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations		X			X		X			
FCS_COP.1/Digital_signature_RSA		X			X		X		X	
FCS_COP.1/Digital_signature_ECDSA					X		X			
FCS_COP.1/hashing_operations		X	X		X	X	X			
FCS_RNG.1			X				X			
User Data Protection										
FDP_ACC.1/User_Maintenance	X									
FDP_ACF.1/User_Maintenance	X									
FDP_ACC.1/TOE_Maintenance	X							X		
FDP_ACF.1/TOE_Maintenance	X							X		
FDP_ACC.1/Block-Unblock_User	X							X		
FDP_ACF.1/Block-Unblock_User	X							X		
FDP_ACC.1/Signing			X							
FDP_ACF.1/Signing			X							
Identification & Authentication										
FIA_ATD.1				X						X
FIA_UAU.2	X	X		X				X		X
FIA_UAU.5/Privileged_User	X			X				X		X
FIA_UAU.5/ES		X								
FIA_UID.2	X	X		X				X		X
FIA_USB.1/User				X						X
FIA_USB.1/Privileged_User				X						X
Security Management										

FMT_MSA.1/User_Management	X							X		
FMT_MSA.1/Signing			X							
FMT_MSA.1/TOE_Management	X							X		
FMT_MSA.3/User	X									
FMT_MSA.3/TOE	X							X		
FMT_SMF.1	X							X		
FMT_SMR.2				X				X		X
Protection of the TSF										
FPT_ITT.1					X					
FPT_STM.1									X	
FPT_TST.1						X				
TOE access										
FTA_SSL.3				X						
Trusted Path/Channels										
FTP_ITC.1/ES_to_TOE					X					
FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign					X				X	

Table 7.1. Security requirement coverage

7.1.1. Rationale

OT.USER_MANAGEMENT is handled by FIA_UAU.2, FIA_UAU.5/Privileged_User, FIA_UID.2, which ensure that S.Privileged_User must be identified and authenticated to gain access to protected resources. The requirements for access control in FDP_ACC.1/TOE_Maintenance, FDP_ACF.1/TOE_Maintenance, FDP_ACC.1/User_Maintenance FDP_ACF.1/User_Maintenance, FDP_ACC.1/Block-Unblock_User and FDP_ACF.1/Block-Unblock_User together with the TOE management functions in FMT_SMF.1 ensure that any modification to R.Reference_User_Authentication_Data is performed under control of the S.Privileged_User. Specifically, the FDP_ACC.1/TOE_Maintenance and FDP_ACF.1/TOE_Maintenance ensure that only the S.Privileged_User may switch TOE to configuration mode (enable config service). The FDP_ACC.1/User_Maintenance and FDP_ACF.1/User_Maintenance ensure that TOE accepts requests for modification of Reference_User_Authentication_Data only in configuration mode. The FDP_ACC.1/Block-Unblock_User and FDP_ACF.1/Block-Unblock_User ensure only the S.Privileged_User may block (or unblock) the R.Reference_User_Status (the status of R.Reference_User_Authentication_Data).

The FMT_MSA.1/User_Management, FMT_MSA.1/TOE_Management provide the ability to manage related security attributes to S.Privileged_User and S.ES (specifically: R.Reference_User_Status and R.TSF_DATA to S.Privileged_User; R.Reference_User_Authentication_Data to S.ES under the control of S.Privileged_User). The FMT_MSA.3/User and FMT_MSA.3/TOE provide functionality of static attribute initialisation for these attributes.

It is noted this objective is supported by identification and authentication of S.Privileged_User and S.ES covered by OT.ROBUST_TOE_ACCESS and OT.ES_Authentication_Signing.

OT.ES_Authentication_Signing: is handled by FIA_UAU.2, FIA_UAU.5/ES and FIA_UID.2 which ensure that identification and authentication of the S.ES. This is supported by:

- FCS_COP.1/Digital_signature_RSA used for signing of API (REST & SOAP API);
- The FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys, FCS_CKM.2/Key_establishment_of_session_keys, FCS_CKM.2/Exchange_of_X.509_certificates. FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations,

FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations, FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations and FCS_COP1/Digital_signature_RSA, and the FCS_COP.1/hashing_operations.
These SFRs cover the establishment of TLS channel including the authentication of the S.ES.

OT.BIOSIGNATURE_INTEGRITY_CONFIDENTIALITY is handled by FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data, which describes requirements for generation of encryption and decryption key for biometric data; the FCS_CKM.4 which describes requirements for destruction of these keys; the FCS_CKM.2/Symmetric_key_transport_protocol, which describes requirements for secure transfer of this key between the parts of TOE and FCS_COP.1/Biometric_data_encryption_in_App and FCS_COP.1/Biometric_data_encryption_in_Core, which describes requirements for encryption confidentiality of biometric data during transmission between biocertiX App and signaturiX Core, between signaturiX Core and SimplySign and between signaturiX Core and S.ES. Additionally, R.Document with R.Embedded.BioSignature is hashed following the FCS_COP.1/hashing_operations for transmission between signaturiX Core and SimplySign (for sealing purpose). The FDP_ACC.1/Signing and FDP_ACF.1/Signing ensure access control for the signing operation, specifically assure that R.Document is authenticated by the authenticated S.User using R.Reference_User_Authentication_Data and the Device is authenticated using R.Reference_Device_Authentication_Data. The FMT_MSA.1/Signing ensures that only S.User can create the R.Document and R.EmbeddedBioSignature following the Signing SFP. The QR/AC code generated during signing operation is resistant to analysis following the FCS_RNG.1

It is noted this objective is supported by identification and authentication of S.User and S.ES covered by OT.ROBUST_TOE_ACCESS and OT.ES_Authentication_Signing.

OT.ROBUST_TOE_ACCESS is handled by FIA_ATD.1, FIA_UAU.2, UAU.5/Privileged_User, FIA_UID.2, FIA_USB.1/User and FIA_USB.1/Privileged_User, which ensure that only identified and authenticated System Administrators (S.Privileged_User) and users (S.Users) gain access to protected resources. The FMT_SMR.2 puts restriction on that entity cannot be assigned more than one user role. Additionally, FTA_SSL.3 limits session (administrator session and user session) to a defined time period.

OT.SECURE_CHANNEL is handled by:

- The FTP_ITC.1/ES_to_TOE and FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign ensure that the TOE communicates via trusted channels (accomplished with TLS protocol) with all external entities (ES, VAULT, SimplySign).
- The FPT_ITT.1 ensures that TSF data is confidential and integral (accomplished with TLS protocol) when it is transmitted between separate parts of the TOE.

The TLS itself (TLS 1.2 and TLS 1.3) is handled by:

FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys, FCS_CKM.2/Key_establishment_of_session_keys, FCS_CKM.2/Exchange_of_X.509_certificates, FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations, FCS_COP.1/hashing_operations FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations, FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations, FCS_COP.1/Digital_signature_RSA, FCS_COP1/Digital_signature_ECDSA.

All mentioned above SFRs cover the generation and the establishment of session keys, the exchange of X.509 v3 certificates and all cryptographic operations required for TLS (TLS 1.2 and TLS 1.3) protocol implementation.

The evaluated TOE configuration covers only the following cipher suites:

- TLS 1.3 suites: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

- TLS 1.2 suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

OT.INTEGRITY is handled by the FPT_TST.1 providing self-tests during initial start-up end ensuring that the integrity of software and data has been maintained. This is supported by the FCS_COP1/hashing_operations to ensure the integrity of TOE. It is also handled by FAU.GEN.1 which requires to generate self-test results.

OT.CRYPTOGRAPHIC_FUNCTIONS is handled by FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data, which describe requirements for generation of

encryption and decryption key for biometric data. The FCS_CKM2/Symmetric_key_transport_protocol ensure secure transfer of this key between the parts of TOE. The FCS_COP.1/Biometric_data_encryption_in_App and FCS_COP.1/Biometric_data_encryption_in_Core ensure confidentiality of biometric data.

The FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys, FCS_CKM.2/Key_establishment_of_session_keys, FCS_CKM.2/Exchange_of_X.509_certificates, FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations, FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations and FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations, FCS_COP.1/Digital_signature_RSA and FCS_COP1/Digital_signature_ECDSA cover the generation and the establishment of session keys, the exchange of X.509 v3 certificates and all cryptographic operations required for TLS protocol implementation.

The FCS_COP.1/Digital_signature_RSA used by signing of API ensures the security of signing of REST & SOAP API. The FCS_COP.1/Digital_signature_RSA used for signing audit records ensure that the content of each audit record/log is electronically signed with a dedicated private key separate from the key used to sign/seal the signed document. The FCS_COP.1/hashing_operations covers requirement to ensure the integrity of all the data stored, processed and transmitted to/from TOE and between parts of TOE.

The FCS_RNG.1 requires that random number generation is compliant with NIST SP 800-90 Rev.1 and NIST SP 800-22 Rev.1a. The FCS_CKM.4 requires that the TSF zeroises keys to be destroyed.

The above cryptographic functions implement approved cryptographic algorithms as defined in related SFRs.

OT.MANAGE is handled by FMT_SMF.1 that contains the TOE management functions for the S.Privileged_User The requirements for access control to these functions are handled by FDP_ACC.1/TOE_Maintenance and FDP_ACF.1/TOE_Maintenance, FDP_ACC.1/Block-Unblock_User and FDP_ACF.1/Block-Unblock_User. The FIA_UAU.2, FIA_UAU.5/Privileged_User, and FIA_UID.2 ensure that only identified and authenticated S.Privileged_Users gain access to TOE management functions. The FMT_SMR.2 puts restriction that entity cannot be assigned more than one user role.

The FMT_MSA.1/User_Management and FMT_MSA.1/TOE_Management provide the ability to manage R.Reference_User_Status and R.TSF_DATA security attributes to S.Privileged_User. The FMT_MSA.3/TOE provide functionality of static attribute initialisation for the TOE attributes.

OT.AUDIT is handled by the requirements for audit record generation FAU_GEN.1 using reliable time stamps FPT_STM.1. The FCS_COP.1/Digital_signature_RSA used for signing audit records ensure that the content of each audit record/log is electronically signed with a dedicated private key. The FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign ensure that the TOE provides trusted channels via the TLS protocol used to transfer the audit logs to the external signaturiX Audit.

OT.ROLES is handled by FIA_ATD.1, FIA_UAU.2, FIA_UAU.5/Privileged_User, FIA_UID.2, FIA USB.1/User and FIA USB.1/Privileged_User which ensure that only identified and authenticated users (S.Users and S. Privileged_Users) gain access to protected resources. The FMT_SMR.2 puts restriction on these roles that entity cannot be assigned more than one role.

7.2. SFR Dependencies

The dependencies between SFRs are addressed as shown in Table 7.2.

Requirement	Dependencies	Fulfilled by
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data	[FCS_CKM.2 or FCS_COP.1], FCS_CKM.4	FCS_COP.1/Biometric_data_encryption_in_App, and FCS_COP.1/Biometric_data_encryption_in_Core, FCS_CKM.4
FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_CKM.2/Key_establishment_of_session_keys, and

Requirement	Dependencies	Fulfilled by
		FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations, and FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations FCS_CKM.4: This dependency is not necessary as the keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_CKM.2/Symmetric_key_transport_protocol	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data FCS_CKM.4
FCS_CKM.2/Key_establishment_of_session_keys	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys FCS_CKM.4: This dependency is not necessary as the keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives
FCS_CKM.2/Exchange_of_X.509_certificates	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys FCS_CKM.4: This dependency is not useful as X.509 certificates are public

Requirement	Dependencies	Fulfilled by
		(consequently there is no need to destroy them after use).
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data
FCS_COP.1/Biometric_data_encryption_in_App	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data FCS_CKM.4
FCS_COP.1/Biometric_data_encryption_in_Core	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data FCS_CKM.4
FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys FCS_CKM.4: This dependency is not necessary as the keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys FCS_CKM.4: This dependency is not necessary as the keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks

Requirement	Dependencies	Fulfilled by
		to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]: The keys for TLS protocol asymmetric operation (RSA encryption and decryption) are established during TLS handshake protocol phase using FCS_CKM.2/Exchange_of_X.509_certificates. Consequently, this dependency is addressed by different SFR (FCS_CKM.2/Exchange_of_X.509_certificates). FCS_CKM.4: This dependency is not necessary as the keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_COP.1/Digital_signature_RSA	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]: Keys for the: - digital signature verification for REST API, SOAP API, and - digital signature creation for Audit are installed in the signaturiX Core during TOE initialization phase (these keys are neither generated in the TOE nor they imported by the TOE) following user guidance. The keys for TLS protocol digital signature are established during TLS handshake protocol phase using FCS_CKM.2/Exchange_of_X.509_certificates. Consequently, this dependency is addressed by TOE installation process (in relation to REST API, SOAP API and Audit) and by different SFR

Requirement	Dependencies	Fulfilled by
		(FCS_CKM.2/Exchange_of_X.509_certificates) in relation to TLS. FCS_CKM.4: The keys for the: - digital signature verification for REST API, SOAP API, and - digital signature creation for Audit are securely stored in the keystore file (in encrypted form) protected with the password stored in the vault. Consequently, this dependency is not useful. The keys for TLS protocol digital signature keys reside solely in volatile memory of operating systems of trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_COP.1/Digital_signature_ECDSA	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2]: Keys for the: The keys for TLS protocol digital signature are established during TLS handshake protocol phase using FCS_CKM.2/Exchange_of_X.509_certificates. Consequently, this dependency is addressed by TOE installation process (in relation to REST API, SOAP API and Audit) and by different SFR (FCS_CKM.2/Exchange_of_X.509_certificates) in relation to TLS. FCS_CKM.4: The keys for TLS protocol digital signature keys reside solely in volatile memory of operating systems of

Requirement	Dependencies	Fulfilled by
		trusted operational environment of the TOE, that is on biocertiX server (protected with OE.ENVIRONMENT) and on biocertiX App (protected with OE.BIOSIGNER_DEVICE). Moreover, it is noted, that reference documents for the TLS 1.2 (RFC5246[24], RFC8447[22]), and the TLS 1.3 (RFC8446[11]) do not require (nor even address) the key destruction. Consequently, this dependency is considered as not necessary in the operational environment of the TOE thanks to the OE.ENVIRONMENT and OE.BIOSIGNER_DEVICE objectives.
FCS_COP.1/hashing_operations	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]: The hashing operation (SHS: SHA-256, SHA-384, SHA-512) do not use any cryptographic keys. Consequently, this dependency is not useful. FCS_CKM.4: The hashing operation (SHS: SHA-256, SHA-384, SHA-512) do not use any cryptographic keys. Consequently, this dependency is not useful.
FCS_RNG.1	None	
FDP_ACC.1/User_Maintenance	FDP_ACF.1	FDP_ACF.1/User_Maintenance
FDP_ACC.1/TOE_Maintenance	FDP_ACF.1	FDP_ACF.1/TOE_Maintenance
FDP_ACC.1/Block-Unblock_User	FDP_ACF.1	FDP_ACF.1/Block-Unblock_User
FDP_ACC.1/Signing	FDP_ACF.1	FDP_ACF.1/Signing
FDP_ACF.1/User_Maintenance	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/User_Maintenance FMT_MSA.3/User
FDP_ACF.1/TOE_Maintenance	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/TOE_Maintenance FMT_MSA.3/TOE
FDP_ACF.1/Block-Unblock_User	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Block-Unblock_User FMT_MSA.3/User
FDP_ACF.1/Signing	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Signing FMT_MSA.3/User
FIA_ATD.1	None	

Requirement	Dependencies	Fulfilled by
FIA_UAU.2	FIA_UID.1	FIA_UID.2 (Fulfils. It is hierarchical to FIA_UID.1)
FIA_UAU.5/Privileged_User	None	
FIA_UAU.5/ES	None	
FIA_UID.2	None	
FIA_USB.1/User	FIA_ATD.1	FIA_ATD.1
FIA_USB.1/Privileged_User	FIA_ATD.1	FIA_ATD.1
FMT_MSA.1/User_Management	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/User_Maintenance FMT_SMR.2 (Fulfils. It is hierarchical to FMT_SMR.1) FMT_SMF.1
FMT_MSA.1/Signing	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/Signing FMT_SMR.2 (Fulfils. It is hierarchical to FMT_SMR.1) FMT_SMF.1
FMT_MSA.1/TOE_Management	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/TOE_Maintenance FMT_SMR.2 (Fulfils. It is hierarchical to FMT_SMR.1) FMT_SMF.1
FMT_MSA.3/User	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/User_Management FMT_SMR.2 (Fulfils. It is hierarchical to FMT_SMR.1)
FMT_MSA.3/TOE	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/TOE_Management FMT_SMR.2 (Fulfils. It is hierarchical to FMT_SMR.1)
FMT_SMF.1	None	
FMT_SMR.2	FIA_UID.1	FIA_UID.2 (Fulfils. It is hierarchical to FIA_UID.1)
FPT_ITT.1	None	
FPT_STM.1	None	
FPT_TST.1	None	
FTA_SSL.3	None	
FPT_ICT.1/ES_to_TOE	None	
FPT_ICT.1/TOE_to_ES/Audit/Vault/SimplySign	None	

Table 7.2. Dependencies

7.3. Rationale for SARs

The EAL2 has been selected to:

- demonstrate that the set of security assurance requirements selected for the TOE are internally consistent, and,
- gain an initial assurance that all required functionalities are correctly implemented by the TOE.

8. TOE Summary Specification

The TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6 provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 8.1 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

	Security Audit	**Cryptographic Support**	**User Data Protection**	**Identification & Authentication**	**Security Management**	**Protection of the TSF**	**TOE Access**	**Trusted Path/Channels**
FAU_GEN.1	X							
FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data		X						
FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys		X						
FCS_CKM.4		X						
FCS_CKM.2/Symmetric_key_transport_protocol		X						
FCS_CKM.2/Key_establishment_of_session_keys		X						
FCS_CKM.2/Exchange_of_X.509_certificates		X						
FCS_COP.1/Biometric_data_encryption_in_App		X						
FCS_COP.1/Biometric_data_encryption_in_Core		X						
FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations		X						
FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations		X						
FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations		X						
FCS_COP.1/Digital_signature_RSA		X						
FCS_COP.1/Digital_signature_ECDSA		X						
FCS_COP.1/hashing_operations		X						
FCS_RNG.1		X						

FDP_ACC.1/User_Maintenance			X					
FDP_ACF.1/User_Maintenance			X					
FDP_ACC.1/TOE_Maintenance			X					
FDP_ACF.1/TOE_Maintenance			X					
FDP_ACC.1/Block-Unblock_User			X					
FDP_ACF.1/Block-Unblock_User			X					
FDP_ACC.1/ Signing			X					
FDP_ACF.1/ Signing			X					
FIA_ATD.1				X				
FIA_UAU.2				X				
FIA_UAU.5/Privileged_User				X				
FIA_UAU.5/ES				X				
FIA_UID.2				X				
FIA_USB.1/User				X				
FIA_USB.1/Privileged_User				X				
FMT_MSA.1/User_Management					X			
FMT_MSA.1/Signing					X			
FMT_MSA.1/TOE_Management					X			
FMT_MSA.3/User					X			
FMT_MSA.3/TOE					X			
FMT_SMF.1					X			
FMT_SMR.2					X			
FPT_ITT.1						X		
FPT_STM.1						X		
FPT_TST.1						X		
FTA_SSL.3							X	
FTP_ITC.1/ES_to_TOE								X
FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign								X

Table 8.1 Security Functions vs. Requirements Mapping

8.1. TOE Security Functions

8.1.1. Security Audit (FAU)

The TOE logs the security events described in FAU_GEN.1.1. The Security relevant events are all changes to the system that may impact the overall system security and include all operations invoked through the S.Privileged_User and S.User.

- Logging of all events except the results of tests related to FPT_TST.1

  Each log entry contains the date and time of the event (using a reliable timestamp), the type of event, the identity of the entity that initiated the event. Log entries are associated with the user (S.User, S.Privileged_User) that caused the event and the outcome (success or failure) of the event.

  Please note that the content of each audit record/log is electronically signed with a dedicated private key separate from the key used to sign the PDF document using cryptographic SFRs described is section 8.1.2.2 (specifically FCS_COP.1/Digital_signature).

  The resulting audit records are securely stored in an external database and are protected from modification

- Logging of results of tests related to FPT_TST.1

  The results of tests related to FPT_TST.1 are stored on biocertiX server (as the failure of these test impact the correct export of logs to the audit database.

The security functionality described above meets the requirements:

- FAU_GEN.1

8.1.2. Cryptographic Support (FCS)

8.1.2.1. Cryptographic key management

a) Cryptographic key generation

The used cryptographic mechanisms ensure the quality of the generated and distributed keys, etc., which are invoked by the TOE with appropriate parameters such as key type and size.

Biometric data captured in biometriX App are encrypted before sending to signaturiX Core, and then after decryption and converting biometric data into a standardized format once again encrypted before embedding in R.Document. For both encryption purposes AES-256 appropriate keys are generated.

The security functionality described above meets the requirements:

- FCS_CKM.1/Generation_of_encryption_keys_for_biometric_data

The generation of the client and server symmetric encryption keys and client and server MAC keys for the TLS protocol is necessary for establishment of secure communication between S.ES and TOE and between TOE and non-TOE components (Audit, Vault and SimplySign). Appropriate keys are generated during handshake protocol of TLS according EC Diffie-Hellman (ECDHE) algorithm:

for the following TLS 1.3 cipher suites: TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256 and TLS_AES_128_GCM_SHA256;

for the following TLS 1.2 cipher suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 and TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

The security functionality described above meets the requirements:

- FCS_CKM.1/Generation_of_the_client_and_server_encryption_keys

b) Cryptographic key distribution

AES-256 keys used to encrypt biometric data in biocertiX App and decrypt them in signaturiX Core, and to encrypt standardized biometric data before embedding them in R.Document, are secured by the symmetric key transport protocol from biocertiX App to signaturiX Core, and from signaturiX Core to external word, appropriately. For this purpose, asymmetric encryption with 4096-bits RSA is used.

The security functionality described above meets the requirements:

- FCS_CKM.2/Symmetric_key_transport_protocol

The handshake part of TLS protocol enables establishment of client and server symmetric encryption keys and client and server MAC keys. There is also the necessity of exchange of client and server X.509 v3 certificates in the TLS protocol.

The security functionality described above meets the requirements:

- FCS_CKM.2/Key_establishment_of_session_keys &
- FCS_CKM.2/Exchange_of_X.509_certificates & FCS_CKM.2/Key_establishment_of_session_keys

c) Cryptographic key destruction

When the cryptographic keys are no longer used, they are destroyed by the zeroising. Zeroization is done by overwriting once with zeroes.

The security functionality described above meets the requirements:

- FCS_CKM.4

8.1.2.2. Cryptographic operation

Biometric data captured in biometriX App are encrypted before sending to signaturiX Core, and then after decryption and converting biometric data into a standardized format once again encrypted before embedding in R.Document. For both encryption purposes AES-256 is used.

The security functionality described above meets the requirements:

- FCS_COP.1/Biometric_data_encryption_in_App &
- FCS_COP.1/Biometric_data_encryption_in_Core

The secure communication between S.ES and TOE and between TOE and non-TOE components (Audit, Vault and SimplySign) is obtained due to the implementation of TLS protocol.

The security functionality described above meets the requirements:

FCS_COP.1/TLS_protocol_AES_cryptographic_symmetric_operations & FCS_COP.1/TLS_protocol_CHACHA20_POLY1305_cryptographic_symmetric_operations & FCS_COP.1/TLS_protocol_cryptographic_asymmetric_operations & FCS_COP.1/Digital_signature_RSA & FCS_COP.1/Digital_signature_ECDSA.

Besides TLS secure communication an additional security mechanism is used for TOE data exchange with the S.ES and signaturiX Audit. REST & SOAP APIs are digitally signed with 4096 bits RSASSA-PSS version of algorithm.

The security functionality described above meets the requirements:

- FCS_COP.1/Digital_signature_RSA used by signing of API & by signing of API Audit

Several validated versions of the hash functions are available in TOE. SHA-256 is used for the purpose of DTBS/R preparation for electronic sealing of PDF document by Certum SimplySign. For the purpose of time stamping by Certum SimplySign SHS (SHA-256, SHA-384, SHA-512) variants of SHS are used. Inside TOE SHA-256 is used for integrity checks. In TLS protocol SHA-384 and SHA-256 are used as specified in cipher suites for TLS 1.2 and TLS 1.3.

The security functionality described above meets the requirements:

- FCS_COP.1/hashing_operations

8.1.2.3. Generation of random numbers

The TOE implements RNG/PRNG (random Number Generation/ pseudorandom number generation) seeded with software-based entropy source provided by JAVA 11 software (java.security.SecureRandom). It is used to generate the QR/AC used during signing operation.

The security functionality described above meets the requirements:

- FCS_RNG.1

8.1.3. User Data Protection (FDP)

8.1.3.1. Access Control

d) User Maintenance

The User (S.User) maintenance is handled by the S.ES using the implemented API of TOE (signaturiX Core). Note, that the S.ES before it handles any task related to the S.User's maintenance (e.g., the task of modifying the security attributes of the S.User according to R. Reference_User_Authentication_Data) must first be authenticated (the authentication of the S.ES is covered in section 8.1.4).

The security functionality described above meets the requirements:

- FDP_ACC.1/User_Maintenance & FDP_ACF.1/User_Maintenance

e) TOE Maintenance

The TOE maintenance provides administrative interface that is handled by the S.Privileged_User. The S.Privileged_User before it handles any task related to the TOE maintenance (e.g., the task of manage the TSF_Data) must first be authenticated (the authentication of the S.Privileged_User is covered in section 8.1.4).

The security functionality described above meets the requirements:

- FDP_ACC.1/TOE_Maintenance & FDP_ACF.1/TOE_Maintenance

f) Block-Unblock User

The Block-Unblock User (S.User) is handled by the S.Privileged_User and it allows to block a given S.User from performing signing operation. The S.Privileged_User before it handles any task (e.g., the task of Block-Unblock the R.Reference_User_Authentication_Data of S.User) must first be authenticated (the authentication of the S.Privileged_User is covered in section 8.1.4).

The security functionality described above meets the requirements:

- FDP_ACC.1/Block-Unblock_User & FDP_ACF.1/Block-Unblock_User

g) Signing

In order to Sign the document(s), the Signing operation must be authorized. Authorisation is handled by the TOE after the S.User scans the QR code or input AC code via the Tablet keyboard. Then credentials present in QR/AC code is used to display the document Documents to S.BioSigner on the Tablet. The S.BioSigner uses S Pen to place a handwritten signature, thus creating 'primary biometric data' in a format appropriate for the biometric sampling device used.

Where the all biosignature operations are verified and accepted by the TOE (see 1.4) the S.User approves the signed document(s) on the Tablet and logged out of the TOE.

The security functionality described above meets the requirements:

- FDP_ACC.1/Signing & FDP_ACF.1/Signing

8.1.4. Identification and authentication (FIA)

8.1.4.1. User security attribute

The TOE maintains the following S.User security attributes associated with authentication and identification:

- security role: user,

- identification_and_authentication_data: R.Reference_User_Authentication_Data,

- status: R.Reference_User_Status.

The TOE requires S.ES to be the only entity that can modify the R.Reference_User_Authentication_Data security attribute. The TOE requires S.Privileged_User to be the only entity that can modify the R.Reference_User_Status security attribute.

The TOE maintains the following S.Privileged_User security attributes associated with the authentication and identification:

- security role: privileged_user,

- identification_data: R.Privileged_User,

- authentication_data: R.Reference_Privileged_User_Authentication_Data.

The security functionality described above meets the requirements:

- FIA_ATD.1, FIA_USB.1/User & FIA_USB.1/Privileged_User

8.1.4.2. User authentication and identification

Each user (S.Privileged_User, S.User and S.ES) when remotely interacting with the TOE must be unambiguously identified and authenticated before allowing any actions on their behalf. The TOE considers a user to be identified and authenticated when their credentials are verified in the login list and QR/AC code. The S.Privileged_User is authenticated with a login and a strong password and TOTP before they can perform any actions in the Administration application. The S.ES (External System) is authenticated in TOE using TLS certificate, in addition, each API message is authenticated using API public/private keys.

The security functionality described above meets the requirements:

- FIA_UAU.2, FIA_UID.2, FIA_UAU.5/Privileged_User & FIA_UAU.5/ES

8.1.5. Security Management (FMT)

8.1.5.1. Management of security attributes

The TOE security features restrict management of system security attributes and data to S.User, S.ES and S.Privileged_User. Specifically:
- The TOE security features restrict the ability to modify S.User security attributes to S.ES and S.Privileged_User.
- The TOE security features restrict the ability to create R.Document and R.EmbeddedBioSignature to S.User.
- The TOE security features restrict the ability to modify R.TSF_DATA to S.Privileged_User

The security functionality described above meets requirements:

- FMT_MSA.1/User_Management, FMT_MSA.1/Signing and FMT_MSA.1/TOE_Management,

8.1.5.2. Static attribute initialisation

The TOE provides restrictive default values for security attributes that are used to enforce its security policy functions security.

The security functionality described above meets the requirements:

- FMT_MSA.3/User & FMT_MSA.3/TOE

8.1.5.4. Management, specification and restrictions on security data

The TOE provides capability of User Maintenance, TOE Maintenance and block-unblock of S.User. Any request to use this capability can only be executed by an authorised entity (S.User, S.Privileged_User, S.ES). Any entity cannot be assigned more than one role.

The security functionality described above meets the requirements:

- FMT_SMF.1 & FMT_SMR.2

8.1.6. TSF Physical Protection (FPT)

8.1.6.1 Basic internal TSF data transfer protection

The data transmitted between separate parts of the TOE is protected from modification and disclosure. The communication between the App on the tablet and the signaturiX Core on the server is encrypted. This data would include the biometric data as it leaves the capture device, or as it is transmitted between other parts of the TOE. The protection is assured with TLS 1.3 protocol that is supported by cryptographic SFRs described in the section 8.1.2.

The security functionality described above meets the requirements:

- FPT_ITT.1

8.1.6.2. Reliable time stamps

The TOE provides time reliable timestamp using two independent time sources,

- one provided by the biocertiX server operating system's clock as defined in A.TIME_STAMPS
- one provided by external NTP server.

If the TOE detects a time deviation, it automatically notifies the S.Privileged_User by logging the event and suspends its operations (the TOE does not accept signing requests).

The security functionality described above meets the requirements:

- FPT_STM.1

8.1.6.3. TSF testing

The TOE ensures that the TSF software have not been corrupted. The TSF shall run a suite of self-tests during initial start-up to demonstrate correct operation of time stamp service for audit. The TOE provide capability to verify the integrity of TSF and TSF data using hashing (as defined in FCS_COP.1.1/hashing_operations).

The security functionality described above meets the requirements:

- FPT_TST.1.

8.1.7. TOE ACCESS (FTA)

8.1.7.1. TSF-initiated termination

TOE limits the exposure of an administrative (S.Privileged_User) and S.User session (signing operation) that is inactive for whatever reason. If an administrative (S.Privileged_User) or S.User session becomes inactive for a S.Privileged_User defined period, the session is terminated.

The security functionality described above meets the requirements:

- FTA_SSL.3

8.1.8. Trusted Paths/Channels (FTP)

8.1.8.1. Inter-TSF trusted channel

The TOE provides trusted communication channel between itself and another trusted IT product, which is logically separated from other communication channels and provides authentication of its endpoints and protection of the transmitted data from modification or disclosure. The TOE permits the following trusted IT products to initiate communication via the trusted channel: S.ES. The TOE initiate communication via the trusted channel with the following trusted IT products: signaturiX Audit, Vault and SimplySign.

The security functionality described above meets the requirements:

- FTP_ITC.1/ES_to_TOE & FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign

The protection is assured with TLS protocol that is supported by cryptographic SFRs described in the section 8.1.2. The FTP_ITC.1/ES_to_TOE and FTP_ITC.1/TOE_to_ES/Audit/Vault/SimplySign are realized with:

- TLS 1.3 in regard to S.ES and Audit

- TLS 1.2 and TLS 1.3 in regard so SimplySign (depending on the option supported by SimplySign, see)

- TLS 1.2 in regard to vault.

It is noted that the evaluated TOE configuration covers only the following cipher suites:

- TLS 1.3 suites: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

- TLS 1.2 suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.

These suits are enforced by the TOE and its operational environment under the condition that the security objectives for the operational environment are satisfied

Other cipher suites supported by the TOE are out of the scope of TOE evaluation.

Bibliography

1. ETSI EN 319 142-1 V1.1.1 Electronic Signatures and Infrastructures (ESI), PAdES digital signatures, Part 1: Building blocks and PAdES baseline signatures
2. https://developer.android.com/reference/java/security/SecureRandom
3. https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/3900
4. https://csrc.nist.gov/publications/detail/fips/140/2/final
5. SP 800-90A Rev. 1: Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015
6. ISO/IEC 19794-7:2014 Information technology — Biometric data interchange formats — Part 7: Signature/sign time series data
7. Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017
8. Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017
9. Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; CCMB-2017-04-003, Version 3.1, Revision 5, April 2017
10. Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2017-04-004, Version 3.1, Revision 5, April 2017
11. E. Rescorla: The Transport Layer Security (TLS) Protocol Version 1.3. - RFC8446, http://www.ietf.org/rfc/rfc8446.txt, 2018
12. K. Moriarty B. Kaliski,. J. Jonsson, A. Rusch: Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.2. RFC8017, http://www.ietf.org/rfc/rfc8017.txt, 2016
13. https://www.inf.pucrs.br/~calazans/graduate/TPVLSI_I/RSA-oaep_spec.pdf
14. D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC5280, http://www.ietf.org/rfc/rfc5280.txt, 2008
15. FIPS197 - Specification for the ADVANCED ENCRYPTION STANDARD (AES). http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, 2001
16. SP800-38A - Recommendation for Block Cipher Modes of Operation: Methods and Techniques. Version NIST Special Publication 800-38A 2001 Edition, http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf
17. https://csrc.nist.gov/publications/detail/sp/800-38d/final
18. https://csrc.nist.gov/publications/detail/fips/180/4/final
19. Y. Nir, A. Langley, ChaCha20 and Poly1305 for IETF Protocols – RFC8439, https://www.rfc-editor.org/rfc/rfc8439, 2018.
20. SP 800-90A Rev. 1, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015, https://csrc.nist.gov/publications/detail/sp/800-90a/rev-1/final
21. SP 800-22 Rev. 1a, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, April 2010, https://csrc.nist.gov/publications/detail/sp/800-22/rev-1a/final
22. https://www.rfc-editor.org/rfc/pdfrfc/rfc8447.txt.pdf
23. Digital Signature Standard (DSS), July 19, 2013, https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.186-4.pdf
24. E. Rescorla: The Transport Layer Security (TLS) Protocol Version 1.2. – RFC5246, 2008, https://www.rfc-editor.org/pdfrfc/rfc5246.txt.pdf.

2024.06.19
07:50:59
+02'00'