Common Criteria

Protection Profile for an ArchiSafe Compliant Middleware for Enabling the Long-Term Preservation of Electronic Documents

Version	Date	Author	Remarks
1.0.0	24/10/09	Dr. Wolf Zimmer	Final version
1.2	28/03/14	Dr. Wolf Zimmer	Final Version

Table of Contents

Figures

Tables

1 PP Introduction

This document represents a Protection Profile (PP) for products enabling the preservation of evidence of (cryptographically signed) electronic documents for long terms by implementing the ArchiSafe concept[1] developed by the Physikalisch-Technische Bundesanstalt (PTB) - the German National Metrology Institute providing scientific and technical services.[2] An essential goal of the ArchiSafe concept is to decouple client software application from storage solutions, which preserve the archived data, in a secure manner.

1.1 PP Reference

PP Name:	Common Criteria Protection Profile for an ArchiSafe Compliant Middleware for Enabling the Long-Term Preservation of Electronic Documents
Certification ID:	BSI-CC-PP-0049-2014
PP Version:	1.2
Date:	28.03.2014
Applicant:	Bundesamt für Sicherheit in der Informationstechnik (German Federal Office for Information Security)
Authors:	Dr. Wolf Zimmer, CSC Deutschland Solutions GmbH
Keywords:	ArchiSafe, TR-ESOR, Protection Profile
CC Version:	3.1 Revision 4

1.2 TOE Overview

Legally compliant electronic business based on electronic documents is not possible without serious precautions to ensure the authenticity and integrity of the digital information, at least for the time schedule of legally specified and regulated retention times. The ArchiSafe approach to long-term preservation of evidence of (cryptographically signed) electronic documents claims:

- to use permanent and standardized document formats for the contents data only, which guarantees the long-term readability of the stored information,

[1] http://www.archisafe.de

[2] The ArchiSafe concept motivates the German BSI Guideline 03125 “Preservation of Evidence of Cryptographically Signed Documents” (also abbreviated as TR-ESOR). The Technical Guideline BSI-TR 03125 provides a detailed guide for German Federal Agencies describing how especially electronically signed data and documents can be stored in a trustworthy manner in the sense of legally valid preservation of evidence over long periods of time – until the end of the retention periods. For the Technical Guideline BSI TR 03125 see also https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03125/index_htm.html

- to package the contents data together with all the business information, required for a complete reconstruction of the business operation in the future in a self-contained archive object,
- to protect the evidential integrity and authenticity of the actual content (primary information) by strong cryptographic operations, like digital signatures and digital time-stamps,
- to sustain the non-repudiation of cryptographically signed and archived information objects by evidential proof and renewal of the electronic signatures,
- to reduce the dependencies from obsolescent IT infrastructure and storage technology by a straight service-oriented, multi-tier and client capable architecture.

The TOE specified in this PP enforces decoupling and access control to storage systems used for the long-term preservation of (cryptographically signed) electronic documents. The TOE also enforces the provisioning of a justification, if archived data shall be deleted before its retention time.

1.2.1 Usage and major security features of the TOE

The target of evaluation (TOE) is a product or part of a product providing the core of a middleware which acts as security gateway to storage solutions. The TOE mainly decouples the data flow (i.e. the flow of data objects to be archived) between third party applications, such as document management systems, and the storage solutions. The architecture of such a system is exemplarily shown in Figure 1.

The **client software application (CS)** submits the (cryptographically signed) information to be preserved in a **submission information package (SIP)**[3] to the **storage unit (SU)** via the TOE. The TOE identifies and authenticates the requesting CS and manages the verification of the submission information packages for compliance to rules defined by the administrator of the TOE.[4] This includes the management of checks concerning the existence, the quality and the validity of digital signatures potentially contained in the submission information package or the execution of cryptographic operations like creation of signatures or timestamps for sealing (unsigned) data before depositing them in the storage. For cryptographic operations the TOE interfaces an external crypto provider, denominated as **Crypto-Module** in Figure 1.

---

[3] The denomination follows in general the OAIS framework for sharing archival notions. OAIS distinguishes between what is preserved, an Archival Information Package (OAIS AIP), what is submitted to the archive, a Submission Information Package (OAIS SIP), and what is delivered to the archive clients, a Dissemination Information Package (OAIS DIP), s. also: http://www.personal.leeds.ac.uk/~ecldh/cedars/ieee00.html Deviating from OAIS framework and for reasons of better distinctness this document uses the denomination Submission Information Package for all information packages to be archived which will be submitted from a client software application via the TOE to a storage unit. Vice versa all information packages stored in a storage unit which can be requested by client software application are denominated as Archival Information Packages.

[4] See definition of “verification” in chapter 3.1.3 in this PP

The storage unit in the back-end receives the submitted submission information package from the TOE - or another trustworthy component (usually an **Evidence Preservation Component** as shown in Fig. 1) which in turn received the submission information package from the TOE - for saving. The stored data object is now called **archival information package (AIP)**.

The TOE quits the successful storage of the AIP by sending back a unique **archive object identifier (AOID)** to the requesting CS. This **AOID** may be generated outside the TOE, e.g. by the storage unit or by another non-TOE part of the middleware and is required for accessing the archive information package in the future by the CS.

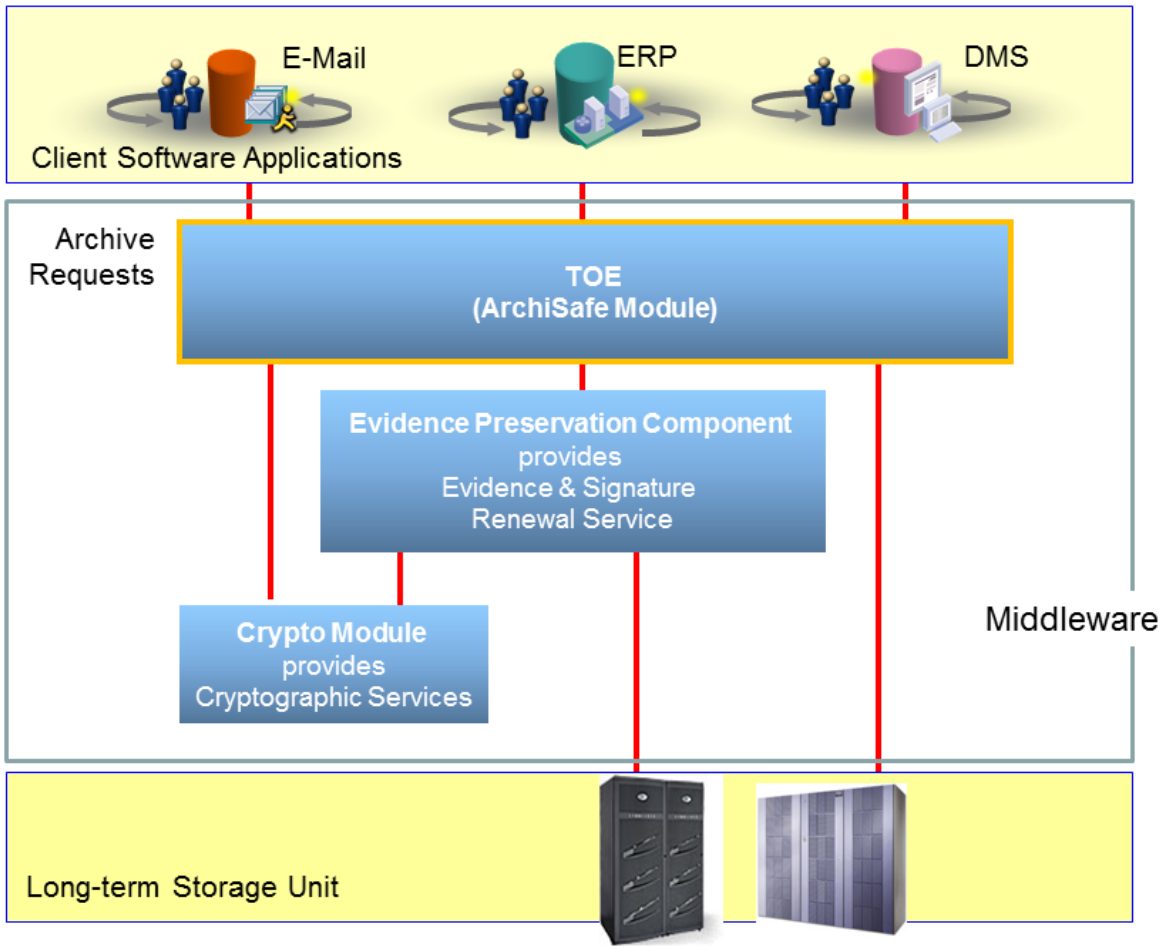

Figure 1: Architectural Overview

The trustworthy and non-TOE **Evidence Preservation Component** in Fig. 1 manages the execution of necessary functionalities and/or mechanisms to preserve the integrity, authenticity and non-repudiation of the saved data. For cryptographic operations the **Evidence Preservation Component** interfaces the Crypto-Module.

Based on the functionality to decouple the data flow between client applications and the storage systems, the TOE provides the following general security functionalities:

(SS 1) preventing the access to the storage systems from unknown client applications by reliable identification and authentication of these external entities,

(SS 2) preventing the storage of submission information packages (SIP) which in whole or in part cannot be verified successfully corresponding to the rules deposited in the TOE in order to guarantee interoperability between client applications and storage systems,

(SS 3) forwarding of successfully verified SIP's to the dedicated storage systems only or another trusted application which in turn forwards the SIP to the dedicated storage systems only,

(SS 4) preventing the deletion of AIP's before the expiry of their retention time without a justification.

(SS 5) retrieval and delivery of AIP from the dedicated storage system (to the CS) only

The TOE itself does not provide any mechanisms for the preservation of the integrity, authenticity and non-repudiation of (cryptographically signed) electronic documents by creation, proof or renewal of evidence data or data relevant to evidence, like electronic signatures or timestamps. The TOE does also not protect the confidentiality of the documents.

1.2.2 TOE Type

The TOE is an IT middleware component or part of an IT middleware component that trustworthy and reliably mediates and controls the access to a SU for submission of SIPs, retrieval or deletion of AIPs or requests of evidence records of AIPs.

1.2.3 Required non-TOE hardware/software/firmware

The TOE runs as an application on an IT system and needs the protection by the underlying system platform, e.g. the operating system.

The CS, the Crypto-Module, Evidence Preservation Component, and the SU (or another trustworthy applications interfacing with the TOE and the SU) are not part of the TOE but the TOE depends on some functions provided by these components or other components outside the TOE.

The TOE itself does not implement any cryptographic mechanisms for protecting or evaluating the integrity and authenticity of the data to be saved. For this purpose the TOE uses trustworthy crypto providers which are explicitly not part of the TOE. Crypto providers may be implemented in hardware, software or firmware.

The TOE itself also does not provide any functionality and/or mechanisms to preserve the integrity, authenticity and non-repudiation of the saved data and to renew security measures which serve for the preservation of the integrity, authenticity and the non-repudiation of the saved data. For this purpose the TOE uses trustworthy components (e. g. denominated as **Evidence Preservation Component** in Fig.1) which are explicitly not part of the TOE. These components may be implemented in hardware, software or firmware.

2 Conformance Claim

2.1 CC Conformance Claim

The Protection Profile is based upon

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 3.1, Revision 4, CCMB-2012-09-001,
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; Version 3.1, Revision 4, CCMB-2012-09-002,
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; Version 3.1, Revision 4, CCMB-2012-09-003,

referenced hereafter as [CC].

This Protection Profile claims the following CC conformance:

- part 2 conformant
- part 3 conformant
- evaluation assurance level (EAL) 3

2.2 Conformance Statement

Security targets or other PPs wishing to claim conformance to this PP can do so as *Strict PP conformance*. *Demonstrable PP conformance* is not allowed for this PP.

3 Security Problem Definition

3.1 Definitions

3.1.1 Subjects

Administrator (Admin)

The Administrator installs the TOE and is in charge of the correct configuration of the TOE. In particular the Administrator is responsible for the correct implementation of the rules needed for a →verification of submission information packages.

Another trustworthy application

This term is usually equivalent with → Evidence Preservation Component in Fig. 1 but can also identify any other trustworthy external i. e. non-TOE component which is interconnected between the TOE and the →storage unit and provides an interface to the TOE equivalent with the storage interface.

Client

An agency or company who operates at least one →CS.

Client Software Application (CS)

An active external IT entity which is acting on behalf of an authorized user and capable and authorized to use the TOE for submitting →archive requests to the →SU.

Crypto-Module (also called Crypto Provider)

A trusted external i. e. non-TOE component which will be used by the TOE and other non-TOE components to perform trustworthy cryptographic operations.

Evidence Preservation Component

A trustworthy external, i.e. a non-TOE, component which provides or manages any functionality and/or mechanisms to preserve the integrity, authenticity and non-repudiation of the saved data and to renew security measures which serve for the preservation of the integrity, authenticity and the non-repudiation of the saved data.

Organization using the TOE

An agency or company who operates and/or uses the TOE.

It may be possible that the →clients and their applications and/or the →storage unit(s) are owned by another agency/company but this will not be differentiated in this PP.

Storage System or Storage Unit (SU)

A storage system which stores data for a long-term.

3.1.2 Objects

Primary Information

The contents data (primary information) representing the business information to be stored.

Application Note: This PP does not want to specify the data structure or format of primary information submitted to the archive. However, it is strongly recommended to use standard formats like ASCII, PDF/A or TIFF. In case of XML-based submission information packages the primary information may be converted into a native text format (MIME Base64 coded) for embedding it in XML.

Meta Information (Metadata)

Data associated with →primary information in the →submission information package serving for the identification and reconstruction of the business and archive context of the primary Information.

Cryptographic data relevant to evidence

Data like cryptographic signatures, certificates or any other cryptographic data which serve to assure the integrity and authenticity of data to be archived. This cryptographic data relevant to evidence is also stored in the →submission information package.

Submission Information Packages (SIPs)

A conceptual data container which may comprise →primary information, →metadata and →cryptographic data relevant to evidence, required for an evidentiary reconstruction of business transactions in the future. Submission information packages will be denominated as →archival information packages when they are saved in the →storage system.

Application Note: This PP does not want to specify data structures of a submission information package in detail. Product developers shall be free to specify data structures of submission information packages which can be successfully verified and/or processed by their own procedures and rules deposited in the TOE.

Archival Information Packages (AIPs)

Once a →submission information package was successfully checked and processed by the TOE and delivered to the →SU, it is called **archival information package (AIP)**. Archival information packages contain all →primary information, →metadata and → cryptographic data relevant to evidence, required for an evidentiary reconstruction of business transactions in the future stored in the specified format. Archival information packages can be accessed by a uniquely identified and authorized →CS only which provides a valid →AOID.

Application Note: This PP does not want to specify data structures of an archival information package in detail. Product developers shall be free to specify data structures of the stored archival information packages. Due to necessary preservation measures however, relating to legally prescribed retention times, it is strongly recommended to use self-contained data structures which might be verified and/or processed by rules deposited in the TOE for any retrieval request. In addition, archival information packages may be augmented with a reference to the submitting CS (e. g. stored as meta information by the TOE during the ingest).

Archive Objects

Archive Objects is the generic term for →submission information packages, →archive information packages, →cryptographic data relevant to evidence or particular data which will be read from chosen archival information packages.

TOE configuration data

TOE internal data required for the correct execution of the security functionalities, especially for the correct and reliable identification and authentication of other units which are not part of the TOE as well as for verification of →SIPs and processing of →archive requests.

Rules

The rules are part of the →TOE configuration data and specify operations the TOE must perform on →archive objects and →archive requests. Rules must be specified by the organization using the TOE.

Application Note: The rules may specify that the TOE

- *must initiate to digitally sign or timestamp any submission information package.*[5]
- *has to start the generation of an evidence record for any or a particular request for retrieval of archival information packages. For this purpose, the TOE may interface to an external crypto provider or to another special and trustworthy application.*

Protocol Data

Log information produced by the TOE.

Evidence Data

According to the specification of the IETF[6] cryptographic data for all AIPs calculated and maintained in order to be able to prove the integrity and authenticity of →archival information packages at and since a certain time. Evidence Data as specified by the IETF are generated and maintained outside the TOE. Evidence Data are generated and/or retrieved on request as an Evidence Record[6] for a certain AIP.

3.1.3 Operations

Archive Requests

An archive request is a call from the →Client Software Application to the TOE to perform a certain operation on the →storage unit. The following Archive Requests must at least be supported by the TOE:

- **Archive Submission Request** means, the Client Software Application wants to store (new) →submission information packages into the storage unit. The submission information packages are included in this archive request.
- **Archive Retrieval Request** means, a Client Software Ap-

[5] In cases, for example, that unsigned data shall be saved or added to the archive, cryptographic operations performed on the data may serve as a proof about the availability of the data at a certain time.

[6] Gondrom, T., Brander, R., Pordesch, U.: IETF RFC 4998 : *Evidence Record Syntax (ERS)*, at http://www.ietf.org/rfc/rfc4998.txt and Blazic, A. J., Saljic, S., Gondrom, T. : *Extensible Markup Language Evidence Record Syntax* at http://tools.ietf.org/html/rfc6283

plication wants to read archival information packages from the storage unit. The retrieval request shall return the archival information packages in self-contained, open and standardized data structures and formats agreed between the organization using the TOE and the organization which operates the storage unit. Modifications of the archive information packages during the retrieval must not be possible.

- **Archive Deletion Request** means, the Client Software Application wants to delete particular archival information packages from the storage unit. A deletion request may happen before or after the →retention time of the archival information package. The TOE enforces a justification, if archival information packages shall be deleted before expiration of the retention time.
- **Archive Evidence Request** means, the Client Software Application requests →evidence data to the fact that archival information packages does exist unmodified within the storage unit since a certain point of time until now.

Application Note: A product or a part of a product which claims to serve as a TOE may implement additional requests and/or functionalities out of the scope of this protection profile. These additional requests/functionalities anyway must not compromise the security objectives of this PP.

Verification of →archive objects

Verifications of →archive objects mean that the TOE enforces the processing of the archive objects in accordance with a set of →rules stored in the →configuration data of the TOE. This may include managing the execution of cryptographic operations which checks the validity of potentially existing digital signatures, the execution of cryptographic operations which serve for protecting the integrity and authenticity of archive objects or renewing →evidence data which prove the unmodified existence of →archival information packages in the →storage.

Application Note: The TOE may verify for example the data structures and/or data formats of the submission information package (e. g. against a valid XML schema).

Archive Submission Request See “Archive Requests”

Archive Retrieval Request See “Archive Requests”

Archive Deletion Request See “Archive Requests”

Archive Evidence Request See “Archive Requests”

3.1.4 Security Attributes

Client Software Application Identity All →Client Software Applications which use the TOE shall have a unique identity, e.g. a numeric value or a unique name. The TOE shall connect to client software applications only whose identity is known by the TOE.

Application Note: This PP does not want to specify the identity of a Client Software Application in detail. Product developers shall be free to use their own attributes and procedures.

Crypto Provider Identity Any →crypto provider (denominated as Crypto-Module in Fig. 1) connected to the TOE and used for performing cryptographic operations shall have a unique identity, e.g. a numeric value or a unique name. The TOE shall connect to crypto providers only whose identity is known by the TOE.

Application Note: This PP does not want to specify the identity of a Crypto Provider in detail. Product developers shall be free to use their own attributes and procedures. But, it is worth to note, that TOE itself is required to implement secure cryptographic operations or other measures needed to provide assured identification of its endpoints and to establish a trusted channel between itself and the crypto provider. It is not acceptable to assume that the environment of the TOE will provide this channel.

Storage Unit Identity / Trustworthy Application Identity Each →storage unit connected to the TOE or another trustworthy application which in turn connects to the storage unit (e. g. the → **Evidence Preservation Component** in Fig. 1) must have a unique identifier, e.g. a numeric value or a unique name. The TOE shall only connect to storage units/trustworthy applications whose identity is known by the TOE.

Application Note: This PP does not want to specify the identity of a storage unit or other trustworthy applications in detail. Product developers shall be free to use their own attributes and procedures.

Retention Time The retention time of an →archival information package is an optional attribute storing the date and time when this AIP can be deleted without justification. The value will be specified for each archival information package.

Justification In case of an → Archive Deletion Request before end of the → Retention Time a justification must be given documenting the reason for that premature deletion. That can be done by a free

	text field or selection boxes or other means.
Archive Object ID (AOID)	The archive object ID is a unique identifier of any →archival information package stored in the storage unit. This AOID may be generated outside the TOE, e.g. by the storage unit or by a non-TOE part of the middleware, when a →submission information package will be sent to the TOE and stored in the SU. This AOID will be returned to the submitting →client software application by the TOE for using it as a security attribute for accessing the archival information package.
Archive Object Specific Credentials	The → Archive Object ID (AOID) and the → Retention Time and in case of an → Archive Deletion Request before end of the Retention Time a → Justification. *Application Note: This PP does not want to specify in detail additional Archive Object Specific Credentials or any combinations of them which serve to confirm the identity of an archive object. Product developers shall be free to use additional credentials specified by the organization using the TOE.*
Another trustworthy Application Identity	All →another trustworthy applications which are used by the TOE shall have a unique identity, e.g. a numeric value or a unique name. The TOE shall connect to other trustworthy applications only, if those identities are known by the TOE. *Application Note: This PP does not want to specify the identity of another trustworthy application in detail. Product developers shall be free to use their own attributes and procedures.*

3.2 Assumptions

The description of assumptions illustrates the security aspects of the environment in which the TOE is intended to be used.

A.ADMIN The administrators of the TOE, of the crypto provider or other trustworthy 3rd party components connected to the TOE, of the storage system, the underlying systems, and of the communication connections (e.g. the LAN) are not careless, wilfully negligent, or hostile, and will follow and abide the instructions provided by the administrator's guidance. They are well trained to securely and trustworthy administer all aspects of the TOE operations as well as all other involved processes or operations in accordance with the guidance.

The administrators will protect their credentials used for authentication. Credentials must not be disclosed to other individual.

A.AUTHENT All CS, SU, and any trustworthy special applications (e.g. the **Evidence Preservation Component** in Fig. 1) which are authorized by the IT-Environment for using the TOE or to be used by the TOE, identify and authenticate the TOE before data transfer.

A.COMMUNICATION The communication interconnections between the TOE and all non-TOE components and systems, are protected by the environment – by physical or logical security measures – against disclosure as appropriate regarding the need for information disclosure of the clients.

A.CONFIGURATION The TOE is securely configured and all data required for the configuration operation of the TOE are secure and reliable transported to and installed on the machine which runs the TOE.

A.EVIDENCEDATA The generation, storage, management and renewal of evidence data for proving the unmodified existence of archive information packages at a certain time will be provided by trustworthy special applications (e.g. the **Evidence Preservation Component** in Fig. 1) in a secure non-TOE environment.

A.NO_BYPASS The TOE is integrated in the IT environment in such a way that all storage access by the CS cannot bypass the TOE, if it is mandated or required by policies of the organization which uses the TOE.

A.PHYSPROT The machine on which the TOE runs is protected against unauthor-

	ized physical access and modification.
A.RULES	Rules defined for operating on archive objects and archive requests by the TOE do not introduce any security risk.
A.SERVER	The machine on which the TOE, systems and applications run is free from malware and viruses. Systems and applications running on the server are securely installed. An unauthorized access to functions, processes and data of the TOE is prevented by the security mechanisms of the underlying system. *Application Note: The environment on which the TOE runs does not grant any unauthorized access to TSF (TOE Security Functions) data.*
A.STORAGE	The dedicated SU provides a reliable, secure and available storage of archival information packages (AIP), even for long-terms. *Application Note: Replacement of the SU (e.g. by a newer device or a device with more storage capacity) is acceptable as long as the TOE, the TOE operations and all data relevant for the TOE and its operation as well as the security objectives of this PP are not affected or compromised.*
A.TIMESTAMP	The environment of the TOE is able to provide reliable time-stamps to the TOE.
A.TOKEN	The environment of the TOE, e. g. the SU or another non-TOE part of the middleware, provides a reliably generated unique archive object identifier (AOID) for any successfully archived submission information package.
A.TRUSTAPP	The archive requesting CS is secure and provides reliable measures regarding the authentication and access control of its (human) users.
A.TRUSTCRYPTO	Only trustworthy cryptographic components are used. The cryptographic components do not send any security relevant and confidential data to any external entity and will reliably protect all security relevant and confidential data from disclosure by an external entity.

3.3 Threats

The threat agents can be categorized as either

- Unidentified individuals or client software applications, i.e. entities not known by the TOE but having access to the communication interfaces exposed by the TOE or to the client software applications, or
- Identified users of the TOE, i.e. individuals or entities, which may access resources controlled by the TOE.

The threat agents are assumed to originate from a well-known user community in a non-hostile environment. The TOE therefore protects against threats of inadvertent or casual attempts to breach the system security. The TOE is not intended to be used in environments where protection is required against determined and hostile attacks to breach the system security at all. Resuming, the following threats need to be countered by the TOE:

T.CRYPTO_SPOOF	An attacker attempts to substitute the crypto provider or to intercept and manipulate the communication between the TOE and the crypto provider.
T.DATA_ACCESS1	An attacker attempts to gain unauthorized access to the SU by using an authorized client software application in an unintended way, e.g. by sending manipulated AOIDs.
T.DATA_ACCESS2	An attacker attempts to gain unauthorized access to the SU by spoofing external entities, e.g. by simulating an authorized client software application.
T.DATA_ACCESS3	An attacker attempts to gain unauthorized access to archive objects by exploiting requests or functionalities additionally implemented by the TOE but not specified in this PP.
T.DATA_DELETION	A (user of a) CS attempts to delete an archival information package before expiry of the retention time of the AIP without any justification.
T.DATA_MODIFY	An attacker attempts to modify an archive object in a specific manner during transmission between CS and the TOE. Objective of the attacker is that the manipulated archive object will be stored or that the CS assumes that the manipulated archive object was actually stored.
T.EVIDCOMP_SPOOF	An attacker attempts to substitute the Evidence Preservation Component or to intercept and manipulate the communication between the TOE and the Evidence Preservation Component.

T.STORAGE_SPOOF	An attacker attempts to substitute the SU or another trustworthy application which in turn is dedicated to forward the SIP to the SU or to manipulate the communication between the TOE and the SU or the other trusted applications.
T.TOE_SPOOF	An attacker attempts to feign TOE functionalities to external components like the CS or the SU.

3.4 Organizational Security Policies

P.ACCESS The TOE has to provide at least the following operations:

- Archive Submission Request,
- Archive Retrieval Request,
- Archive Deletion Request and,
- Archive Evidence Request.

P.AOID The TOE must not interpret or change (modify) the archive object ID.

P.CONFIGURATION The TOE must select the right configuration data per archive request, must interpret it in a correct manner and execute the rules defined within in the configuration data in the right order.

P.RETURN After successful storage of a submission information package the TOE has to return the archive object ID (AOID) to the requesting CS.

P.RULES In order to decouple CS and SU the TOE has to verify Archive objects according to specified rules. The verification may be performed either in the context of a submission request or vice versa in the context of a retrieval request.

When the verification fails the TOE has to react in an appropriate way.

Application Note: The PP does not want to specify whether an error message shall be generated or whether a submission shall be finished and the SIP shall be stored or not. This is up to the product developer and can be a fixed or configurable property. However, "appropriate" does not mean that errors will just be ignored.

4 Security Objectives

This section defines the security objectives for the TOE and its supporting environment. The security objectives are categorized as security objectives for the TOE or for the environment.

4.1 Security Objectives for the TOE

O.ACCESS

The TOE allows at least the following operations:

- Archive Submission Request,
- Archive Retrieval Request,
- Archive Deletion Request and
- Archive Evidence Request.

Application Note: A product or a part of a product which claims to serve as a TOE may implement additional requests and/or functionalities out of the scope of this protection profile. These additional requests/functionalities anyway must not compromise the security objectives of this PP.

O.AOID

The TOE must not interpret or change (modify) the archive object ID.

O.AUTH_REQUEST

The TOE shall authorize archive requests based on the authenticity of the requesting client and archive object specific credentials provided (e.g. the AOID).

O.CONFIGURATION

The TOE assures the selection and application of the appropriate configuration, interprets the configuration data in a correct manner and executes the rules defined within in the configuration data in the right order. The TOE denies an archive request, if any operation defined by the rules failed or cannot completely be executed.

O.CRYPTO_SPOOF

The TOE assures that the crypto provider cannot be substituted without notice.

O.DATA_EXAM

The TOE assures that either the submission information packages at the point of submission or the archival information packages at the point of retrieval request will be verified according to the specified rules.

O.DELETION

The TOE assures that archival information packages can only be deleted by client requests before expiry of the retention time when the delete request will be submitted together with a justification.

O.DELETION_LOG	The TOE must log any delete requests and the accompanying justification, if the retention time of these archive objects is not yet expired.
O.RETURN	After successful storage of submission information packages the response of the TOE to the requesting CS must at least contain the archive object IDs (AOIDs).
O.STORAGE_SPOOF	The TOE assures that the SU or another trustworthy application which in turn is connected to the SU and will be used for saving and retrieving the archive data objects cannot be replaced without notice. (this includes especially also an Evidence Preservation Component)
O.TOE_AUTHENT	The TOE shall be capable to authenticate itself against external non-TOE entities.
O.TOE_COMM	The TOE shall be capable to protect the communication between itself, the CS, the SU, the crypto provider and all other trustworthy application (e. g. an Evidence Preservation Component as shown in Fig. 1) against modification.

4.2 Security Objectives for the Operational Environment

OE.ADMIN The administrators of the TOE, of the crypto provider cryptographic or other trustworthy 3rd party components connected to the TOE, of the storage system, the underlying systems, and of the communication connections (e.g. the LAN) must not be careless, wilfully negligent, or hostile, and will follow and abide the instructions provided by the administrator's guidance. They shall be well trained to securely and trustworthy administer all aspects of the TOE operations as well as all other involved processes or operations in accordance with the guidance.

The administrators shall protect their credentials used for authentication. Credentials must not be disclosed to other individual.

OE.AUTHENT The client software applications (CS), the SU, and any trustworthy special applications (e.g. the Evidence Preservation Component in Fig. 1) which are authorized by the IT-Environment for using the TOE or to be used by the TOE, have to be configured in such a way that they identify and authenticate the TOE before any data transfer.

OE.COMMUNICATION The communication interconnections between the TOE and all non-TOE components and systems, have to be protected by the environment – by physical or logical security measures – against disclosure as appropriate regarding the need for information disclosure of the clients. The communication interconnections between the TOE and all non-TOE components and systems must be protected by the environment – by physical or logical security measures – against threats (e. g. disclosure.) which may compromise the security objectives of this PP.

OE.CONFIGURATION The TOE has to be securely configured and all data required for the configuration of the TOE must secure and reliable transported to and installed on the machine which runs the TOE.

OE.EVIDENCEDATA The generation, storage, management and renewal of evidence data for proving the unmodified existence of archive information packages at a certain time shall be provided by trustworthy special applications (an Evidence Preservation Component as shown in Fig. 1) in a secure non-TOE environment.

OE.NO_BYPASS	The TOE must be integrated in the IT environment in such a way that all storage access by the CS cannot bypass the TOE, if it is mandated or required by policies of the organization which uses the TOE.
OE.PHYSPROT	The machine on which the TOE runs must be protected against unauthorized physical access and modification.
OE.RULES	Rules defined for operating on archive objects and archive re-quests by the TOE must not introduce any security risk.
OE.SERVER	The machine on which the TOE, systems and application run must be free from malware and viruses. Systems and applications running on the server must be securely installed. An unauthorized access to functions, processes and data of the TOE must not be possible.
OE.STORAGE	The dedicated SU has to provide a reliable, secure and available storage of archival information packages (AIP), even for long-terms.
OE.TIMESTAMP	The environment shall be able to provide reliable time-stamps to the TOE.
OE.TOKEN	The environment of the TOE, e. g. the SU or another non-TOE part of the middleware, has to provide a reliably generated unique archive object identifier (AOID) for any successfully archived submission information package.
OE.TRUSTAPP	The archive requesting CS has to provide sufficient trust to be assumed as secure and has at least to provide reliable measures regarding the authentication and access control of its (human) users.
OE.TRUSTCRYPTO	Only trustworthy cryptographic components are allowed to be used. The cryptographic components must not send any security relevant and confidential data to any external entity and have to reliably protect all security relevant and confidential data from disclosure by an external entity.

4.3 Rationale For Security Objectives

This chapter explains how each aspect of the security environment of the TOE will be covered by the security objectives. In addition the security environment is explained.

4.3.1 Coverage of the Assumptions

Table 1: Coverage of the Assumptions

Assumptions	O.ACCESS	O.AOID	O.AUTH_REQUEST	O.CONFIGURATION	O.CRYPTO_SPOOF	O.DATA_EXAM	O.DELETION	O.DELETION_LOG	O.RETURN	O.STORAGE_SPOOF	O.TOE_AUTHENT	O.TOE_COMM	OE.ADMIN	OE.AUTHENT	OE.COMMUNICATION	OE.CONFIGURATION	OE.EVIDENCEDATA	OE.NO_BYPASS	OE.PHYSPROT	OE.RULES	OE.SERVER	OE.STORAGE	OE.TIMESTAMP	OE.TOKEN	OE.TRUSTAPP	OE.TRUSTCRYPTO
A.ADMIN													X													
A.AUTHENT														X												
A.COMMUNICATION															X											
A.CONFIGURATION																X										
A.EVIDENCEDATA																	X									
A.NO_BYPASS																		X								
A.PHYSPROT																			X							
A.RULES																				X						
A.SERVER																					X					
A.STORAGE																						X				
A.TIMESTAMP																							X			
A.TOKEN																								X		
A.TRUSTAPP																									X	
A.TRUSTCRYPTO																										X

A.ADMIN: A.ADMIN is directly covered by OE.ADMIN.

A.AUTHENT: A.AUTHENT is directly covered by OE.AUTHENT.

A.COMMUNICATION: A.COMMUNICATION is directly covered by OE.COMMUNICATION

A.CONFIGURATION: A.CONFIGURATION is directly covered by OE.CONFIGURATION.

A.EVIDENCEDATA: A.EVIDENCEDATA is directly covered by OE.EVIDENCEDATA

A.NO_BYPASS: A.NO_BYPASS is directly covered by OE.NO_BYPASS

A.PHYSPROT: A.PHYSPROT is directly covered by OE.PHYSPROT

A.RULES: A.RULES is directly covered by OE.RULES

A.SERVER: A.SERVER is directly covered by OE.SERVER

A.STORAGE: A.STORAGE is directly covered by OE.STORAGE

A.TIMESTAMP: A.TIMESTAMP is directly covered by OE.TIMESTAMP

A.TOKEN: A.TOKEN is directly covered by OE.TOKEN

A.TRUSTAPP: A.TRUSTAPP is directly covered by OE.TRUSTAPP

A.TRUSTCRYPTO: A.TRUSTCRYPTO is directly covered by OE.TRUSTCRYPTO

4.3.2 Encounter the Threats

Table 2: Coverage of the Threats

Threats	O.ACCESS	O.AOID	O.AUTH_REQUEST	O.CONFIGURATION	O.CRYPTO_SPOOF	O.DATA_EXAM	O.DELETION	O.DELETION_LOG	O.RETURN	O.STORAGE_SPOOF	O.TOE_AUTHENT	O.TOE_COMM	OE.ADMIN	OE.AUTHENT	OE.COMMUNICATION	OE.CONFIGURATION	OE.EVIDENCEDATA	OE.NO_BYPASS	OE.PHYSPROT	OE.RULES	OE.SERVER	OE.STORAGE	OE.TIMESTAMP	OE.TOKEN	OE.TRUSTAPP	OE.TRUSTCRYPTO
T.CRYPTO_SPOOF					X							X														
T.DATA_ACCESS1			X						X																	
T.DATA_ACCESS2			X						X																	
T.DATA_ACCESS3	X		X															X								
T.DATA_DELETION							X	X																		
T.DATA_MODIFY											X	X		X	X											
T.EVIDCOMP_SPOOF										X		X														
T.STORAGE_SPOOF										X		X														
T.TOE_SPOOF											X			X												

T.CRYPTO_SPOOF: This threat is covered by O.CRYPTO_SPOOF (prevents spoofing of the crypto provider without notice) and O.TOE_COMM (prevents unnoticed manipulation of communication between TOE and the crypto provider).

T.DATA_ACCESS1: This threat is covered by O.AUTH_REQUEST (enforces an access control policy) and O.RETURN (ensures that only submitting CS also receives the respective AOID to be used for later access).

T.DATA_ACCESS2: This threat is covered by O.AUTH_REQUEST (enforces an access control policy) and O.RETURN (ensures that only submitting CS also receives the respective AOID to be used for later access).

T.DATA_ACCESS3: This threat is covered by O.ACCESS (specification of the core functions of the TOE which must be part of this PP), O.AUTH_REQUEST (enforces an access control policy on all functions the TOE may provide) and OE.NO_BYPASS (ensures that the TOE and its access control function cannot be bypassed by other means provided by the IT environment).

T.DATA_DELETION: This threat is directly covered by O.DELETION. In addition O.DELETION_LOG ensures that all justifications related to such delete operations will be recorded to provide evidence for correct TOE operation or for auditors.

T.DATA_MODIFY: This threat is directly covered by O.TOE_COMM. Additionally, OE.AUTHENT and O.TOE_AUTHENT enforces resp. enables a bi-directionally authentication of CS and TOE, which prevents a simple man-in-the-middle attack. OE.COMMUNICATION protects the network traffic against disclosure, which makes a directed modification more difficult.

T.EVIDCOMP_SPOOF: This threat is covered by O.STORAGE_SPOOF (prevents spoofing of an Evidence Preservation Component without notice) and O.TOE_COMM (prevents unnoticed manipulation of communication between TOE and an Evidence Preservation Component as shown in Fig.1).

T.STORAGE_SPOOF: This threat is covered by O.STORAGE_SPOOF (prevents spoofing of the storage without notice) and O.TOE_COMM (prevents unnoticed manipulation of communication between TOE and the storage).

T.TOE_SPOOF: This threat is directly covered by O.TOE_AUTHENT (enables the TOE to be authenticated by other components) and especially by OE.AUTHENT, which ensures that all the other components authenticate the TOE before any data transfer. This ensures that spoofing of the TOE would be noticed.

4.3.3 Implementation of Organizational Security Policies

Table 3: Coverage of Organizational Policies

Organizational Security Policies	O.ACCESS	O.AOID	O.AUTH_REQUEST	O.CONFIGURATION	O.CRYPTO_SPOOF	O.DATA_EXAM	O.DELETION	O.DELETION_LOG	O.RETURN	O.STORAGE_SPOOF	O.TOE_AUTHENT	O.TOE_COMM	OE.ADMIN	OE.AUTHENT	OE.COMMUNICATION	OE.CONFIGURATION	OE.EVIDENCEDATA	OE.NO_BYPASS	OE.PHYSPROT	OE.RULES	OE.SERVER	OE.STORAGE	OE.TIMESTAMP	OE.TOKEN	OE.TRUSTAPP	OE.TRUSTCRYPTO
P.ACCESS	X																									
P.AOID		X																								
P.CONFIGURATION				X									X			X										
P.RETURN									X																	
P.RULES						X																				

P.ACCESS: This OSP is directly covered by O.ACCESS.

P.AOID: This OSP is directly covered by O.AOID.

P.CONFIGURATION: This OSP is directly covered by O.CONFIGURATION. Additionally, OE.ADMIN and OE.CONFIGURATION ensures that the TOE is correctly and securely installed and that the rules are configured as intended by the organization operating the TOE.

P.RETURN: This OSP is directly covered by O.RETURN.

P.RULES: This OSP is directly covered by O.DATA_EXAM.

5 Security Requirements

This section comprises security functional and security assurance requirements that shall be fulfilled by a product that is conformant to this protection profile.

- Selections performed have been marked in *italics*.
- Assignments performed have been marked in **bold**.
- Refinements have been marked as underlined.
- Iterations of security requirements have been marked by applying an additional identifier to the appropriate component names.
- Operations, which are not executed, are reproduced from the [CC] without any changes.
- Uncompleted Operations are still written in brackets containing at first the executed part of the operation and subsequently the specification of the operation to be performed.

5.1 Security Policies

5.1.1 Access Control Policy (TSP_ACC)

The TOE shall control the access to the archive according to the following rules:

- Only securely identified and authenticated Client Software Applications (CS) will get permission for accessing the storage unit for writing a new SIP.
- Only securely identified and authenticated Client Software Applications (CS) which uses valid archive requests and provides archive object specific credentials will get permission for accessing the storage unit and the respective archive objects for read, delete and read evidence data.
- Only securely identified and authenticated Client Software Applications (CS) which uses valid archive requests and provide a justification will get permission to delete AIP before expiry of its retention time.

5.1.2 Information Flow Control Policy (TSP_IFC)

The TOE shall implement an information flow control policy which follows the following rules:

- All rules specified for archive object verification shall be performed by the TOE, either at submission or at retrieval request.
- The TOE must not perform an archive request, if an operation defined by the rules deposited in the TOE cannot be performed successfully.
- The TOE shall return the archive object ID as result of a successful archive submission request.

Application Note: All rules specified for archive object verification as well as potential additional rules specified by the organization using the TOE or the product developer shall be performed by the TOE in accordance with the specification and in the context of the respective archive request.

5.2 Security Functional Requirements

5.2.1 Class FAU: Security Audit

5.2.1.1 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

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 [selection, choose one of: *minimum, basic, detailed, not specified*] level of audit; and

c)

- **Successful and unsuccessful archive deletion requests for archival information packages whose retention time is not yet expired.**
- **Unsuccessful authentications of Client Software Applications, Crypto Providers, the storage unit and other trustworthy applications connected to the TOE**
- **Unsuccessful attempts to access Archival Information Packages**
- [assignment: *other specifically defined auditable events]*.

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST,

- **for successful archive deletion requests to archival information packages whose retention time is not yet expired, the justification,**
- [assignment: *other audit relevant information, resulting from additional implemented requests and/or functionalities*].

5.2.2 Class FDP: User Data Protection

5.2.2.1 FDP_ACC.1 Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1 The TSF shall enforce the **TSP_ACC** on

a) **list of subjects: Client Software Applications**
b) **objects: Archive Objects**
c) **operations: archive requests** [assignment: *any other operations which are out of scope of this PP but added to a product or part of a product which claims to serve as a TOE*]

Application Note: The uncompleted operations give a product developer the ability to add some more request types to the TOE. These additional requests / functionalities anyway must not compromise the security objectives of this PP.

5.2.2.2 FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1 The TSF shall enforce the **TSP_ACC** to objects based on the following:

a) list of subjects: Client Software Applications

- **Security Attribute: Client Software Application Identity**

b) objects: Archive Objects

- **Security Attribute(s): Archive Object Specific Credentials (the AOID, the retention time)**

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

- **Only an identified and authenticated CS is allowed to submit a SIP for storage.**
- **Only an identified and authenticated CS which provides a valid Archive Object Specific Credential, at least the AOID, is authorized to read-out or delete the respective AIP.**
- **Only an identified and authenticated CS which provides a valid Archive Object Specific Credential, at least the AOID, and a justification is authorized to delete the respective AIP before expiry of the retention time.**
- **Only an identified and authenticated CS which provides a valid Archive Object Specific Credential, at least the AOID, is authorized to read-out evidence data for the respective AIP**
- [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment*: rules, based on security attributes, that explicitly authorise access of subjects to objects*].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes that explicitly deny access of subjects to objects*].

Application Note: The unfinished operations to enable a product developer to implement some more access control rules for more archive requests. The rules already specified must not be bypassed.

Application Note: These access control rules will be enforced by the ArchiSafe module. They are in addition and completely independent to access controls implemented in the CS or in the SU. The ArchiSafe access control model can also be more complex than depicted here, e.g. group-based or role-based and may consider several clients in parallel. In all cases the access control model has to ensure that unauthorized access, e.g. between different clients with identical AOID ranges, is not possible.

5.2.2.3 FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1 The TSF shall enforce the **TSP_IFC** on

- **Subjects: Client Software Applications, Storage Unit, another trustworthy application which connects to the Storage Unit**
- **Information: Archive Objects, Evidence Data**
- **Operations: Archive Requests**
- [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*].

Application Note: The uncompleted operations give a product developer the ability to control some more information flows.

5.2.2.4 FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control

FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1 The TSF shall enforce the **TSP_IFC** based on the following types of subject and information security attributes:

- **Subject: Client Software Applications,**
  - **Security Attributes: Client Software Application Identity**
- **Subject: Storage Unit**
  - **Security Attributes: Storage Unit Identity**
- **Subject: another trustworthy application which connects to the Storage Unit**
  - **Security Attributes: another trustworthy Application Identity**
- **Information: Archive Objects**
  - **Security Attributes: Type of Archive Request**
- **Information: Evidence Data**
  - **Security Attributes: Type of Archive Request**

[assignment: *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*].

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

- **For all requests**
  - **The TOE must select and execute the appropriate TOE configuration data and rules based on the Client Software Application Identity and/or the archive request type.**
  - **The TOE does not interpret or modify any input or output data, i.e. AOIDs as well as data of SIPs or AIPs (in terms of scripts, etc.)**

    Application Note: Adding data to SIPs in accordance with configuration data or rules defined by the organization using the TOE which govern the handling of SIPs must not compromise the security objectives of this PP

- **Archive Submission Requests**
  - **The TOE forwards the SIP to the Evidence Preservation Component, to the storage unit or to another trustworthy application which in turn forwards the SIP to the storage unit.**

Application Note: The TOE or the IT environment needs to be configured in such a way that the immediate generation of the data for the evidence database is possible based on this information flow.

- If the TOE does not generate the AOID by itself, the TOE shall receive the AOID from the respective component.
- The TOE shall return the AOID for each submitted submission information package to the submitting Client Software Application as result of a successful archive submission request.

- Archive Retrieval Requests
  - The TOE retrieves for each valid AOID the assigned archival information package from the storage unit.
  - The TOE returns for each valid AOID the assigned archival information package to the requesting Client Software Application
- Archive Deletion Requests
  - The TOE deletes the AIP identified by the AOID from the storage unit.
  - The TOE returns the success of the operation to the requesting Client Software Application.
- Archive Evidence Requests
  - The TOE requests Evidence Data from the Evidence Preservation Component for each AIP identified by an AOID.
  - The TOE returns the received Evidence Data to the requesting Client Software Application.

[assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*].

FDP_IFF.1.3 The TSF shall enforce the following

- The TOE has to ensure that the rules for guaranteeing the interoperability of data formats will be performed at Archive Submission or at Archive Retrieval Request.

Application Note: The PP does not want to specify in detail at which point in time the data format will be checked. However, it shall be ensured that for each SIP the rules will be enforced.

[assignment: *additional information flow control SFP rules*].

FDP_IFF.1.4 The TSF shall explicitly authorise an information flow based on the following rules: [assignment: *rules, based on security attributes that explicitly authorise information flows*].

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following

rules:

- **The TOE must not perform an archive request, if the access control rules defined in FDP_ACF.1 denies the access.**
- **The TOE must not perform an archive request, if the verification procedures of the rules deposited in the TOE fail or cannot be completely executed.**

[assignment: *rules, based on security attributes, that explicitly deny information flows*].

Application Note: The uncompleted rules give a product developer the ability to specify some more information flow rules, especially when additionally requests, operations or functionalities out of scope of this PP are implemented. These additional rules must not bypass the rules already specified and must assure that the added requests, operations and/or functionalities does not compromise the security objectives of this PP.

5.2.3 Class FIA: Identification and Authentication

5.2.3.1 FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

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

5.2.3.2 FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

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

5.2.4 Class FMT: Security management

5.2.4.1 FMT_MSA.1 (Access) Management of security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1	The TSF shall enforce the **TSP_ACC** to restrict the ability to *modify and delete* the security attributes **access control rules** to **Administrators.**

Application Note: It is worth to mention that the role "Administrator" may be maintained by the TOE or the IT environment. The term "access control rules" encompasses all rules defined by TSP_ACC as well as potential additional access control rules defined by the product developer or the organization using the TOE.

5.2.4.2 FMT_MSA.1 (Rules) Management of security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1	The TSF shall enforce the **TSP_IFC** to restrict the ability to *modify and delete* the security attributes **TOE configuration data and rules** to **Administrators.**

Application Note: It is worth to mention that the role "Administrator" may be maintained by the TOE or the IT environment. The term "TOE configuration data and rules" encompasses all security relevant attributes which serve to confirm the identity of components connected to the TOE, allowed types of archive requests, as well as access control rules and information flow control rules.

5.2.4.3 FMT_MSA.3 (Access) Static attributes initialisation

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the **TSP_ACC** to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow **nobody** to specify alternative initial values to override the default values when an object or information is created.

Application Note: This SFR shall ensure that all security attributes relevant for accessing archive objects (e.g. the possible types of archive requests) will be initialized with secure default values and that these defaults cannot be changed.

5.2.4.4 FMT_MSA.3 (Rules) Static attributes initialization

Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the **TSP_IFC** to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow **nobody** to specify alternative initial values to override the default values when an object or information is created.

Application Note: This SFR shall ensure that all security attributes relevant for the information flow control (e.g. the TOE configuration data and the rules for verification) will be initialized with secure default values and that these defaults cannot be changed. This holds also valid for the mandatory format verification at submission or retrieval request.

5.2.4.5 FMT_SMR.1 Security roles

Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification

FMT_SMR.1.1	The TSF shall maintain the roles **authorized Client Software Application,** **[**assignment: *the authorised identified roles*].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

Application Note: The roles "Administrator" and "Organization using the TOE" may be defined by the operational environment and is then not maintained by the TSF. Otherwise, the unfinished operation above should be used to consider also these roles. The term "Users" denominates in a first step the different client software applications accessing the archive or vice versa an authorized Client Software Application denominates active external entities acting on behalf of an authorized user.

5.2.5 Class FPT: Protection of the TSF

5.2.5.1 FPT_TDC.1 Inter-TSF basic TSF data consistency

Hierarchical to:	No other components.
Dependencies:	No dependencies.

FPT_TDC.1.1	The TSF shall provide the capability to consistently interpret **TOE configuration data,** [assignment: *list of TSF data types*] when shared between the TSF and another trusted IT product.
FPT_TDC.1.2	The TSF shall use [assignment: *list of interpretation rules to be applied by the TSF*] when interpreting the TSF data from another trusted IT product.

Application Note: This SFR ensures that the TOE can read and interpret the configuration data correctly and in the right order. The operation of the interpretation rules was not detailed because the interpretation of these configuration data may follow different rules/standards in different products. For example one product has its own XML-based configuration data in a file, another product managed that by a central configuration database provided by the Operating System.

5.2.6 Class FTP: Trusted path/channels

5.2.6.1 FTP_ITC.1 (CRYPTO) Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a crypto provider that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

Application Note: It is worth to note, that the TOE itself is required to implement secure cryptographic operations or other measures needed to provide assured identification of the channels endpoints and to establish a trusted channel between itself and the trusted crypto provider. It is not acceptable to assume that the environment will provide this channel.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for **performing all types of cryptographic operations apart from operations which serve to provide assured identification of endpoints between the TOE and non-TOE components as well as to protect the corresponding communication channels from modification or disclosure.**

Application Note: Taking the upper application note into account, product developers shall be free for

(a) using the crypto providers functionality to assure identification of other communication endpoints as well as to protect communication channel data between the TOE and other non-TOE components from modification or disclosure (see other FTP_ITC components) or,

(b) to implement secure cryptographic operations or other measures needed for assured identification of other communication endpoints as well as to protect communication channel data by the TOE itself.

It is worth to mention that, when using the crypto provider functionalities to assure communication endpoints and to establish trusted channels between the TOE and non-TOE components, these functionalities become virtually part of the TOE and are therefore part of a product evaluation.

5.2.6.2 FTP_ITC.1 (CS) Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote Client Software Application that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

Application Note: It is worth to mention, that the TOE itself is required to implement secure cryptographic operations or other measures needed to provide assured identification of its endpoints and to establish a trusted channel between itself and the remote Client Software Application. It is not acceptable to assume that the environment will provide this channel.

FTP_ITC.1.2 The TSF shall permit *remote Client Software Application* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel **only for request responses**.

5.2.6.3 FTP_ITC.1 (STORAGE) Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote storage unit that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

Application Note: It is worth to mention, that the TOE itself is required to implement secure cryptographic operations or other measures needed to provide assured identification of its endpoints and to establish a trusted channel between itself and the remote storage unit. It is not acceptable to assume that the environment will provide this channel.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for

- **Archive Retrieval Requests**
- **Archive Deletion Requests**
- [assignment: *list of additional requests accepted by the TSF*]

Application Note: It is an option that for the archive requests "Retrieval" and "Deletion" the TOE establishes a trusted channel to the Evidence Preservation Component and not to the storage and the Evidence Preservation Component in turn ensures a trusted channel to the storage. For these named requests the Evidence Preservation Component needs typically to be involved for management of evidence data.

Application Note: The uncompleted list of operations gives a product developer the ability to add some more requests to TSF. A product or a part of a product which claims to serve as a TOE may implement additional requests and/or functionalities out of the scope of this protection profile. These additional requests/functionalities anyway must not compromise the security objectives of this PP.

5.2.6.4 FTP_ITC.1 (TAPP) Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and remote trustworthy application (e. g. the Evidence Preservation Component in Fig. 1) that is logically distinct from other communication channels and provides assured identification of its end points and, protection of the channel data from modification or disclosure.

Application Note: It is worth to mention, that the TOE itself is required to implement secure cryptographic operations or other measures needed to provide assured identification of its endpoints and to establish a trusted channel between itself and the remote trustworthy application. It is not acceptable to assume that the environment will provide this channel.

FTP_ITC.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for

- **Archive Submission Requests**
- **Archive Evidence Requests**
- [assignment: *list of additional requests accepted by the TSF*]

Application Note: It is an option that for the archive requests "Submission" and "Evidence" the TOE establishes a trusted channel to the Evidence Preservation Component and not to the storage and the Evidence Preservation Component in turn ensures a trusted channel to the storage. For these named requests the Evidence Preservation Component need typically to be involved for management of evidence data resp. generation of an evidence record.

Application Note: The uncompleted list of operations gives a product developer the ability to add some more requests to TSF. A product or a part of a product which claims to serve as a TOE may implement additional requests and/or functionalities out of the scope of this protection profile. These additional requests/functionalities anyway must not compromise the security objectives of this PP

5.3 Security Assurance Requirements

The following Table 4 gives an overview on the security assurance requirements that have to be fulfilled by the TOE. They correspond to the assurance level EAL3 of part 3 of the Common Criteria.

Table 4: TOE security assurance requirements

Assurance class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.3 Functional specification with complete summary
	ADV_TDS.2 Architectural design
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life-cycle support	ALC_CMC.3 Authorisation controls
	ALC_CMS.3 Implementation representation CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_DVS.1 Identification of security measures
	ALC_LCD.1 Developer defined life-cycle model
ASE: Security target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.2 Analysis of coverage
	ATE_DPT.1 Testing: basic design
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

5.4 Rationale for the Security Functional Requirements

The following table indicates that the security objectives pointed out in section 4.1 will be covered by the security functional requirements represented in section 5.2 of this Protection Profile.

Table 5: Coverage of the security objectives by security functional requirements

	O.ACCESS	O.AOID	O.AUTH_REQUEST	O.CONFIGURATION	O.CRYPTO_SPOOF	O.DATA_EXAM	O.DELETION	O.DELETION_LOG	O.RETURN	O.STORAGE_SPOOF	O.TOE_AUTHENT	O.TOE_COMM
FAU_GEN.1								X				
FDP_ACC.1	X		X				X					
FDP_ACF.1	X		X				X					
FDP_IFC.1		X		X		X			X	X		
FDP_IFF.1		X		X		X			X	X		
FIA_UAU.2			X									
FIA_UID.2			X									
FMT_MSA.1 (Access)			X									
FMT_MSA.1 (Rules)				X								
FMT_MSA.3 (Access)			X									
FMT_MSA.3 (Rules)				X		X						
FMT_SMR.1			X									
FPT_TDC.1				X								
FTP_ITC.1 (CRYPTO)					X						X	X
FTP_ITC.1 (CS)											X	X
FTP_ITC.1 (STORAGE)										X	X	X
FTP_ITC.1 (TAPP)										X	X	X

O.ACCESS: FDP_ACF.1 and FDP_ACC.1 guarantee that the TOE will only allow the specified types of archive requests.

O.AOID: The rules enforced by FDP_IFC.1 and FDP_IFF.1 ensure that the TOE does not interpret any input or output parameters in terms of a script and that the TOE does not change these values. This holds also valid for the AOID.

O.AUTH_REQUEST: FDP_ACC.1 and FDP_ACF.1 enforces the actual access control based on credentials. FIA_UAU.2 and FIA_UID.2 deliver the credential “client application identity” for the access control mechanism. FMT_MSA.1 (Access) and FMT_MSA.3 (Access) ensure that the access control defaults are set restrictive and that this default cannot be changed. FMT_SMR.1 ensures that the TOE is able to manage a role for the authenticated client applications.

O.CONFIGURATION: FDP_IFC.1 and FDP_IFF.1 ensures that the right configuration data will be selected and executed. This includes also the denial of an access in case of incomplete or not successful performance of the rules. FMT_MSA.1 (Rules) and FMT_MSA.3 (Rules) ensures that there are restrictive defaults for the configuration data and that these defaults cannot be changed. FPT_TDC.1 ensures that the configuration rules will be interpreted correctly by the TOE.

O.CRYPTO_SPOOF: FTP_ITC.1 (CRYPTO) enforces a reliable identification of a dedicated crypto provider. Thus, the selected (defined) trustworthy crypto provider cannot be substituted unnoticed.

O.DATA_EXAM: FDP_IFC.1 and FDP_IFF.1 enforce the verification of SIPs/AIPs at the point of submission or at the point of retrieval. FMT_MSA.3 (Rules) ensures that there are restrictive defaults for this and that nobody can change these defaults.
FMT_MSA.1 (Rules) is not relevant here.

O.DELETION: FDP_ACC.1 and FDP_ACF.1 enforce that nobody will be able to delete an archive object before the expiry of its retention time without any justification.

O.DELETION_LOG: FAU_GEN.1 guarantees that any erasure request to archive objects before the expiry of their retention time will be recorded including the justification for that activity.

O.RETURN: FDP_IFC.1 and FDP_IFF.1 enforce that after successful storage of a data object the TOE returns the archive object ID (AOID) to the submitting client software application.

O.STORAGE_SPOOF: FDP_IFC.1 and FDP_IFF.1 ensure that SIPs intended to be stored will be forwarded to the SU or another trustworthy application. FTP_ITC.1 (STORAGE) and FTP_ITC.1 (TAPP) ensure that the SU and the other trustworthy application will be identified and authenticated before it will be used by the TOE and can therefore not be replaced without notice.

O.TOE_AUTHENT: the SFRs FTP_ITC.1 (CRYPTO), FTP_ITC.1 (CS), FTP_ITC.1 (STORAGE) and FTP_ITC.1 (TAPP) require a mutual authentication of the end points of the respective communication connections. This also includes the authentication of the TOE against all the other end points, namely the client software application, the crypto provider, the storage and other trustworthy application (e.g. the Evidence Preservation Component).

O.TOE_COMM: the SFRs FTP_ITC.1 (CRYPTO), FTP_ITC.1 (CS), FTP_ITC.1 (STORAGE) and FTP_ITC.1 (TAPP) require the protection of the communication against modification, namely the communication with the client software application, the crypto provider, the storage and other trustworthy application (e.g. the Evidence Preservation Component).

5.5 Rationale For Assurance Requirements

EAL3 as minimum level for PP compliant products was chosen because the intention of these systems is to provide a trustworthy access point to storage systems including long-term archives.

The definitions of the EALs 1 and 2 states that they are only applicable when a low to medium level of independently assured security is required. Here, a trustworthy long-term archive access point requires a higher level of security.

Due to the fact that the requirements of the German law for authenticity and non-repudiation of digital signatures of documents will not be covered by this PP, the strong requirements of EAL4 are not appropriate.

5.6 Rationale for SFR Dependencies

SFR	Dependencies	Resolved
FAU_GEN.1	FPT_STM.1	Resolved by TOE environment
FDP_ACC.1	FDP_ACF.1	Resolved
FDP_ACF.1	FFP_ACC.1	Resolved
	FMT_MSA.3	Resolved by FMT_MSA.3 (Access)
FDP_IFC.1	FDP_IFF.1	Resolved
FDP_IFF.1	FDP_IFC.1	Resolved
	FMT_MSA.3	Resolved by FMT_MSA.3 (Rules)
FIA_UAU.2	FIA_UID.1	Resolved by hierarchical FIA_UID.2
FIA_UID.2	No dependencies	---
FMT_MSA.1 (Access)	FDP_ACC.1 or FDP_IFC.1	Resolved by FDP_ACC.1
	FMT_SMR.1	Not resolved because the role “Administrator” is assumed to be managed by the IT environment. *Application Note: It may be possible that a specific product manages the role “Administrator”. Then the respective ST shall resolve the dependency.*
	FMT_SMF.1	Not resolved because the management of these security attributes is out of scope. *Application Note: It may be possible that a specific product comes with management functions. Then the respective ST shall resolve the dependency.*
FMT_MSA.1 (Rules)	FDP_ACC.1 or FDP_IFC.1	Resolved by FDP_IFC.1
	FMT_SMR.1	Not resolved because the role “Administrator” is assumed to be managed by the IT environment. *Application Note: It may be possible that a specific product manages the role “Administrator”. Then the respective ST shall resolve the dependency.*
	FMT_SMF.1	Not resolved because the management of these security attributes is out of scope. *Application Note: It may be possible that a specific product comes with management functions. Then the respective ST shall resolve the dependency.*
FMT_MSA.3 (Access)	FMT_MSA.1	Resolved by FMT_MSA.1 (Access)

SFR	Dependencies	Resolved
	FMT_SMR.1	Not resolved because the role “Administrator” is assumed to be managed by the IT environment. *Application Note: It may be possible that a specific product manages the role “Administrator”. Then the respective ST shall resolve the dependency.*
FMT_MSA.3 (Rules)	FMT_MSA.1	Resolved by FMT_MSA.1 (Rules)
	FMT_SMR.1	Not resolved because the role “Administrator” is assumed to be managed by the IT environment. *Application Note: It may be possible that a specific product manages the role “Administrator”. Then the respective ST shall resolve the dependency.*
FMT_SMR.1	FIA_UID.1	Resolved by hierarchical FIA_UID.2
FPT_TDC.1	No dependencies	---
FPT_ITC.1 (CRYPTO)	No dependencies	---
FPT_ITC.1 (CS)	No dependencies	---
FPT_ITC.1 (STORAGE)	No dependencies	---
FPT_ITC.1 (TAPP)	No dependencies	---

6 Acronyms

AIP	Archival Information Package
AOID	Archive Object Identifier
BSI	Bundesamt für Sicherheit in der Informationstechnik (German Federal Office for Information Security)
CC	Common Criteria for IT Security Evaluation
CS	Client Software Application
DIP	Dissemination Information Package
DMS	Document Management System
EAL	Evaluation Assurance Level
ERP	Enterprise Resource Planning
Fig	Figure
IT	Information Technology
LAN	Local Area Network
OAIS	Open Archival Information System
OSP	Organisational Security Policies
PP	Protection Profile
PTB	Physikalisch-Technische Bundesanstalt
SFP	Security Function Policy
SIP	Submission Information Package
ST	Security Target
SU	Storage Unit
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSP	TOE Security Policy