Security Target for ABox 1.0

Version 1.3

21.06.2006

BSI-DSZ-CC-0365

T-Systems International GmbH
Dachauer Strasse 651
D-80995 München
Germany

Author: Franco Maoro
Phone: +49 (711) 972-45196
E-mail: Franco.Maoro@T-Systems.com
File Name ABox_ST_v13.doc

Document history

Version	Date	Changes	Remarks
Version 0.1	09 September 2005	Document created.	
Version 0.2	06 December 2005	Description of the intended usage of the TOE and of the administration tool; modification of the security problem definition and its effects; completion of FDP_IFF.1; completion of the functional requirements sufficiency rationale; revision of the TOE security functions; editorial changes.	
Version 0.3	13 February 2006	Modifications resulting from the Kickoff-Meeting on 8 February 2006 at BSI (mainly changing the SOF claim to “basic”). Adaption of the TOE description in chapter 2 (Admin Client is now part of the TOE). Renaming the extended TOE “ABox 1.0” instead of “Abox Core 1.0”. Adding of some terms to the glossary in chapter 7.	
Version 1.0	13 March 2006	Modifications resulting from the evaluation.	
Version 1.1	4 April 2006	Comments from the BSI.	
Version 1.2	27 April 2006	Minor changes	
Version 1.3	21 June 2006	Minor changes	

Contents

Tables

1 ST Introduction

1.1 ST Identification

Title:	Security Target - ABox 1.0
Sponsor:	T-Systems International GmbH
Editors:	Franco Maoro
CC Version:	2.2
Assurance Level:	EAL 3
General Status:	Final Version
Version Number:	1.3
Certification ID:	BSI-DSZ-CC-0365
Evaluation Facility:	SRC Security Research & Consulting GmbH
Certification Body:	Bundesamt für Sicherheit in der Informationstechnik, Germany
Keywords:	ABox

1.2 ST Overview

1 This Security Target defines the security objectives and requirements for the ABox 1.0. It addresses the security services provided by this software, mainly:

- encryption and decryption of sensitive data;
- pseudonymisation of sensitive data and resolution of the pseudonyms;
- authorisation of users to read and write sensitive data on basis of a user group concept.

1.3 CC Conformance Claim

2 This Security Target claims conformance to

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.2, January 2004
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.2, January 2004
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.2, January 2004

or equivalently to

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999

- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999

- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999

under consideration of all relevant agreed RIs

as follows:

- Part 2 conformant

- Part 3 conformant

- Package conformant to EAL3.

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation; Version 2.2, January 2004

or equivalently

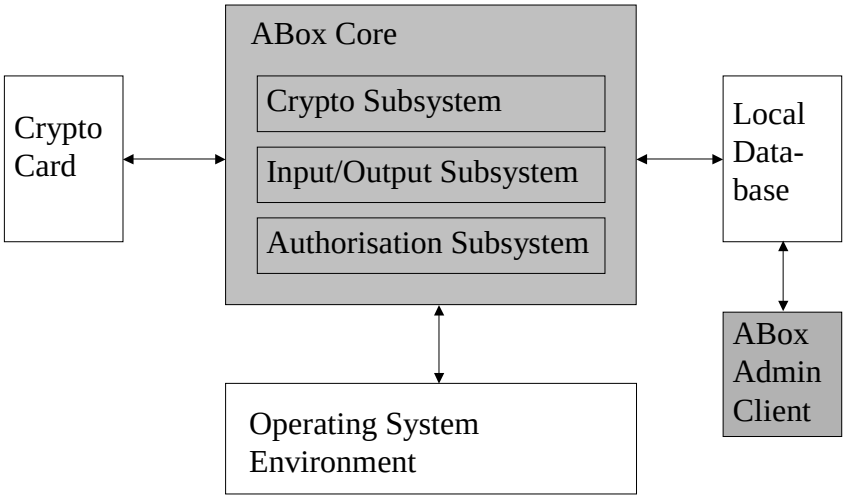
- Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 1.0, August 1999

under consideration of all relevant agreed RIs.

2 TOE Description

2.1 TOE scope and TOE environment

3 The TOE of this Security Target is a software, the so-called ABox 1.0 together with guidance documentation. The TOE comprises two parts, the ABox Core and the ABox Admin Client (the shaded parts in the following figure). The scope of delivery comprises the executables, libraries, configuration files and installation scripts for the ABox Core and ABox Admin Client, furthermore user and installation guidance (the user guidance covers also administration aspects). The ABox Core consists of the three indicated sub-systems, the ABox Admin Client constitutes a fourth subsystem.



4 The ABox Core is embedded into a local system ("local ABox system") in which it is connected to an operating system environment (application software), to a database ("ABox database") and optionally to a crypto card. TOE users interact with the TOE via the operating system environment. TOE management and configuration data are stored in the ABox database and are administered via an administrative interface, the ABox Admin Client.

5 The local ABox system is embedded into a larger system ("ABox WAN") including a remote central database which can be accessed via a network (connected to the operating system environment) and including further local ABox systems, see the following figure. In the ABox WAN, sensitive and non-sensitive user data are transmitted as HTTP data stream and stored in the central database. Data stream from a browser to the central database is denoted as "input stream", the response stream as "output stream" (as seen from the central database). The TOE shall ensure that sensitive data does not leave the local ABox system in cleartext. For this, input and output stream are directed through the ABox Core which "desensitises" input stream and "sensitises" output stream. This is done basically by two alternative methods, encryption/decryption and "pseudonymisation"/"pseudonym resolution".

6 In the local ABox system, user authentication is done by the operating system environment. User requests can only reach the ABox Core if the user beforehand has successfully been authenticated.

7 User write requests are formatted into an input stream which is directed through the ABox Core by the operating system environment. The ABox Core desensitises sensitive data contained in the input stream before forwarding it to the central database.

8 User read requests are forwarded to the central database and initiate an output stream which is also directed through the ABox Core by the operating system environment. The ABox Core replaces desensitised data by the original data in this stream, if the user belongs to the user group who may see these data.

2.2 Administration of access rights

9 Data access rights are defined on the basis of so-called user groups and may grant read or write access (write access includes read access) for a group. Data that is written under a certain group may only be seen by users who have read access for this group. Pseudonyms may also be activated for other groups by the administrator.

10 There are three types of administrators:

- supervisor: defines groups and their attributes, introduces users in the system;
- group owner: manages a specific user group, assigns and revokes users' write or read permissions for this group, may create user configurations, but may assign read or write permission only for the own group, can designate a deputy for himself;
- deputy: continues the group owner's tasks in his absence (apart from designating a deputy).

11 The management data like group and user configuration data, users' read and write permissions is stored in the ABox database and administered via a designated interface, the ABox Admin Client. The files used to store these management data and the ABox Admin Client are part of the TOE.

12 Users may be authorised to read or write for several groups, but at any time only one (write) group assignment is active. This active group assignment is relevant if the user requests to write or read certain data. The user group which is initially active after ABox login is a configurable default setting. The user can change his active group dynamically within the write groups assigned to him by invoking an ABox function.

2.3 Input stream

13 Sensitivity of user data is defined by setting "input markers" when entering the data into the local system. Sensitivity may be defined on file level, on field level and on the level of arbitrarily defined text blocks within fields of MIME-type "text". The "input stream" (input data, input markers, information about the originating user and possibly further management data) is sent via the operating system environment to the ABox Core.

14 On receiving an input stream, the ABox Core first checks it for syntax errors (f. i. fields with only one limiting parenthesis, incomplete markers, characters less than "blank" (0x20) or larger than 0x7f in a marked area, fields set as "pre-assigned" by the administration component which are not completely marked). Syntax errors lead to a refusal of the input and to an error message.

15 If the input stream is correctly formatted, the ABox Core transforms it by desensitising the contained sensitive information and by replacing the input markers with "output markers". Output markers are different from input markers and include the information if encryption or pseudonymisation has been applied. If by the transformation a field length

is exceeded, the processing of the input is aborted, and the user is notified about this by an error message. The transformed input stream is scanned for blacklisted words (words contained in the group specific blacklist which is stored in the ABox database). A blacklist match leads to a refusal of the input and to an error message.

16 If the input stream has successfully been transformed, the ABox Core sends it via the operating system environment to the central database for storage of the data (inclusive the output markers). Due to the desensitisation, neither unauthorised ABox Core users nor persons directly accessing the central database can read the sensitive information stored there.

17 Two methods of desensitisation may be applied by the ABox Core, encryption and "pseudonymisation". Encryption may be performed either by the ABox Core or by an external crypto card connected to the ABox Core. Pseudonymisation means that sensitive data is replaced by a pseudonym (in the case of a sensitive file the file name plays the role of the pseudonym). The ABox Core automatically stores the contents belonging to the pseudonym in the ABox database. In the input stream, sensitive data is replaced by a pseudonym in the case of a sensitive field or text block, and is replaced by dummy contents in the case of a sensitive file.

18 The method of desensitisation (pseudonymisation or encryption; encryption by software of hardware; the encryption key to be used) is determined by the active write group assignment of the user who originated the input (if no write group is assigned, the user is not allowed to input). The assignment of desensitisation methods to user groups is done in the administration component.

2.4 Output stream

19 A read request is forwarded by the ABox Core to the central database which in response creates an "output stream" consisting of the requested data in desensitised form including the output markers. The ABox Core scans the output stream for output markers and tries to resolve them on basis of the active group of the requesting user which determines the sensitisation method to be applied.

- If pseudonymisation is to be applied, the contents belonging to the output marker is interpreted as pseudonym. The ABox Core looks up the pseudonym in the ABox database under the groups the user is authorised to read. If it is found there, then in the output stream the pseudonym is replaced by the corresponding contents, otherwise by a specified text (which is fixed per ABox) indicating the non-availability of the data.

- If decryption is to be applied, the contents belonging to the output marker is interpreted as cryptogram. The ABox Core applies the group specific decryption method to it, and replaces it in the output stream by the result.

20 If the contents belonging to the output marker has been replaced by the ABox Core, the output marker itself is removed. If the contents belonging to the output marker could not be replaced by the ABox Core (when a pseudonym has to be resolved, but is not found), the output marker is retained.

21 The transformed output stream is forwarded to the originating user via the operating system environment.

3 Security Problem Definition

22 The Security Problem Definition is the part of a ST, which describes

- **assets**, which the TOE shall protect,
- **users** of the TOE are humans or machines, who might use the TOE rightly or who might be threat agents (i. e. attack the security of the assets),
- **operational security policies**, which describe overall security requirements defined by the organisation in charge of the overall system including the TOE. In particular this may include legal regulations, standards and technical specifications.
- **threats** against the assets, which shall be averted by the TOE together with its environment
- **assumptions** on security relevant properties and behaviour of the TOE's environment

3.1 Introduction

3.1.1 Assets

23 The assets to be protected by the TOE and its environment are as follows

Name of asset	Description
sensitive user data	Confidential user data stored or processed in the system.
user authentication data	The user identifier and password entered by a user to authenticate himself to the system, and the reference values stored in the ABox database used for verification.
management data	Definition of user groups and their read/write members, the group specific data desensitisation method (pseudonymisation respectively which encryption algorithm with which key; software or hardware encryption).
encryption passwords	Passwords used for the key generation.
cryptographic keys	Keys used in the system for the encryption and decryption of sensitive user data.
pseudonym lists	Group specific lists stored in the ABox database indicating what pseudonyms are actually used in this group to reference sensitive user data and what contents each pseudonym stands for.
blacklists	Group specific lists of words that must not be contained in an input.
TOE software code	The programming code the TOE consists of.

Table 1: Assets to be protected by the TOE and its environment

3.1.2 Subjects

24 This Security Target considers the following subjects, which can interact with the TOE:

Name of subject	Description
ABox user	The ABox user is the legitimate user of the ABox Core.
supervisor	The supervisor defines users, user groups, their group owners and configuration data.
group owner	The group owner manages a particular user group, assigns and revokes users' write or read permissions for this group, may create user configurations, but may assign read or write permission only for the own group . Also he can designate a deputy.
deputy	The deputy continues the group owner's tasks (except for deputy designation) in his absence.
ABox database	The ABox database is connected to the TOE and stores the user authentication data, the cryptographic keys, the management data, and the pseudonym lists.
ABox WAN	The wide area network the ABox is embedded in, into which input stream is directed (after passage through the ABox Core) and from which output stream is received (which is directed through the ABox Core).
other person	All persons who interact with the TOE without being so authorised (as one of the preceding roles).

Table 2: Subjects

3.2 Organisational Security Policies

25 The concrete security services to be provided by the TOE are defined by the specification documents which is formulated as an organisational security policy. For this reason the organisational security policies define here the greater part of the security needs for the TOE compared to lists of individual threats.

26 OSPs will be defined in the following form:

OSP.name Short Title

Description.

27 The TOE and its environment shall comply to the following organisational security policies (which are security rules, procedures, practices, or guidelines imposed by an organisation upon its operations, see CC Part 1 [5], sec. 3.2).

OSP.Desens Desensitisation of data leaving the local ABox system

Data marked as sensitive shall leave the local ABox system only after being desensitised.

OSP.Conceal Concealing of sensitive data in the output

Data marked as sensitive shall not be output (browser display, printer, …) to the user who has not the right to see it, in the case of pseudonymisation even not the belonging pseudonym (instead a constant message stating the non-availability of the data).

OSP.Administration Administration of User and Group Configurations

The configuration data of user groups and the assignment of users to user groups may be managed by supervisors, group owners and deputies as indicated in Table 2 about Subjects. It shall be possible that users with write permission for more than one user group may choose at the runtime which they want to exercise.

3.3 Threats

28 This section describes the threats to be averted by the TOE or by the IT environment of the TOE or by a collaboration of both. These threats result from the TOE method of use in the operational environment and the assets stored in the TOE.

29 Threats will be defined in the following form:

T.name Short Title

Description, for example starting “An attacker tries to ...”.

30 The TOE shall avert the threats as specified below. As potential attackers all kinds of subjects as listed in Table 2 are considered, as far as they

- try to perform actions, which they are not allowed to by their access rights as defined in this ST and
- may have expertise, resources and motivation as expected from an attacker with low attack potential.

T.Compromise Compromise of sensitive data

An attacker tries to acquire and disclose sensitive data.

3.4 Assumptions

31 The assumptions describe the security aspects of the environment in which the TOE will be used or is intended to be used.

32 The format for assumptions will be as follows:

A.name short title

Description.

33 The following assumptions hold for the usage environment:

A.Users Trustworthiness of ABox users

The users of the local ABox system will use the TOE according to the guidance and all other security instructions.

A.Administrators Trustworthiness of ABox administrators

The authorised administrators of the local ABox system will use the local ABox database according to the guidance and all other security instructions.

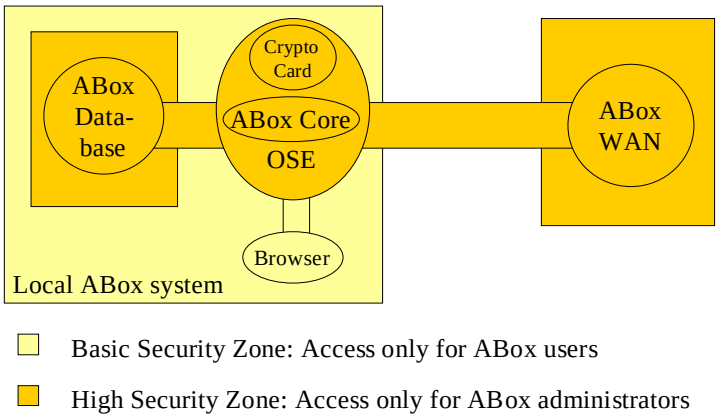
A.ABox_Access Access to the local ABox system

The local ABox system is physically protected and access-controlled so that it may be accessed only by authorised users.

A.WAN Security of the ABox WAN

All storage media and transmission lines in the ABox WAN are protected against tapping of the transmitted data. The personnel having access to the data is trustworthy and will not compromise sensitive user data.

Graphical representation of the access assumptions



4 Security Objectives

34 This chapter describes the security objectives for the TOE and the security objectives for the TOE environment. The security objectives for the TOE environment are separated into security objectives for the development and production environment and security objectives for the operational environment.

Application note 1: The separation of the security objectives for the TOE environment follows the approach of CC version 2.4 and does not violate the CC version 2.2. The CC version 2.2 addresses the operational environment only.

4.1 Security Objectives for the TOE

35 This section describes the security objectives for the TOE, which address the aspects of identified threats to be countered by the TOE and organisational security policies to be met by the TOE.

36 Objectives for the TOE will be defined in the following form

OT.name short title

Description of the objective.

37 The following objectives shall be upheld by the TOE:

OT.Cryptography Implementation of Cryptographic Algorithms

The used cryptographic algorithms are implemented according to their definition.

These algorithms are:

- AES128, AES192, AES256, TDES in CFB and CBC mode.

OT.Stream_Process Processing of Input and Output Stream reaching the ABox Core

In the end usage phase, the TOE shall implement the information flow control policy **SFP_Stream_Process**, which is defined in the following table:

SFP_Stream_Process
The following subjects causing information flow are covered by this policy: (see also section 3.1.2, Table 2):
ABox user, other person
The following subject security attributes are taken into account by this policy:
user identifier
The following information flow is covered by this policy:
input stream, output stream

The following information security attributes are taken into account by this policy: input markers The following operations causing information flow are covered by this policy: user read or write request (causes input stream), answer to user read request (causes output stream) The following rules are defined for the processing of an input stream: The security profile of the belonging user is loaded from the ABox database. If the input stream contains syntax errors (f. i. fields with only one limiting parenthesis, incomplete markers, characters less than “blank” (0x20) or larger than 0x7f in a marked area, fields set as “pre-assigned” by the administration component which are not completely marked), the input stream is rejected and an error message sent to the originator. If the input stream is a correctly formatted read request, it is forwarded into the ABox WAN. If the input stream is a correctly formatted write request, it is scanned for sensitive regions (regions that are marked by input markers). These are desensitised by replacing the sensitive data with non-sensitive data and the input markers with output markers. If by the transformation a field length is exceeded the input stream is rejected and an error message sent to the originator. The transformed input stream is scanned for blacklisted words (words contained in the group specific blacklist which is stored in the ABox database). A blacklist match leads to a refusal of the input and to an error message. If the write request has successfully been transformed, it is sent via the operating system environment into the ABox WAN. The following rules are defined for the processing of an output stream: The security profile of the belonging user is loaded from the ABox database. The output stream is scanned for desensitised regions (regions that are marked by output markers). It is attempted to replace these regions with the original clear text in the following way: Basis of the transformation is the active group of the requesting user which determines the sensitisation method to be applied. If pseudonym-resolution is to be applied, the belonging contents is interpreted as pseudonym. The pseudonym is looked up in the ABox database under the groups the user is authorised to read and, if found, replaced by the corresponding contents in the output stream; otherwise it is replaced by a specified message (which is fixed per ABox) stating the non-availability of the data. If decryption is to be applied, the belonging contents is interpreted as cryptogram. Then the group specific decryption method is applied to it, and the cryptogram is replaced with the result of the decryption in the output stream. If the contents belonging to the output marker has been replaced by the ABox Core, the output marker itself is removed. If the contents belonging to the output marker could not be replaced by the ABox Core (when a pseudonym has to be resolved, but is not found), the output marker is retained. After transformation the output stream is sent via the operating system environment to the originating user.

Table 3: SFP for processing input and output stream reaching the ABox Core

OT.Administration Administration of User and Group Configurations

In the end usage phase, the TOE shall implement the access control policy **SFP_Administration**, which is defined in the following table:

SFP_Administration
The following subjects requesting access are covered by this policy: (see also section 3.1.2, Table 2): supervisor, group owner, deputy, ABox user, other person

The following subject security attributes are taken into account by this policy: user identifier The following objects are covered by this policy: user and group configuration data The following operations causing access requests are covered by this policy: user request to change configuration data (group configuration data include the processing mode of this group (encryption/pseudonymisation/no encryption), in the case of encryption the encryption password and the encryption method, in the case of pseudonymisation the maximum number of synonyms) The following rules are defined for the processing of a user access request: Supervisors may create a group configuration or create or delete a user configuration. Supervisors may modify all fields in all configurations except for read-only fields. Group owners may assign a deputy for their group. Group owners and deputies may create user configurations, but may assign read or write permission only for their group; they may modify user configurations with respect to read or write permissions for their group. A user may change his active write group within the write groups he is authorised for.

Table 4: SFP for access to management data

4.2 Security Objectives for the Operational Environment

OE.ABox_Access Access to the local ABox system

The local ABox system is physically protected and access-controlled so that it may be accessed only by authorised persons .

OE.User_Info Information about secure usage of the ABox Core

The users of the ABox Core need to be informed clearly about secure usage of the product.

In particular secure usage includes

- not to concede access to the local ABox system to unauthorised persons;
- not to tell their passwords to others;
- not to tell sensitive data to others;
- not to use encryption of field content if the length of the text conceals inadmissible information about the content, in that case either to encrypt larger passages or to use pseudonymisation instead of encryption.

OE.Users Trustworthiness of ABox users

The legitimate users of the local ABox system will use the ABox Core according to the guidance and all other security instructions.

OE.ABox Use of ABox in all local ABox systems

In all involved local ABox systems, a genuine and correctly configured ABox is used.

OE.OSE Protection of sensitive data by the operating system environment

The operating system environments in all involved local ABox systems protect the data processed by the ABox Core during a user session against unauthorised access. After termination of the session the data are made unavailable. The operating system environments ensure that all input and output streams are directed through the ABox Core.

OE.DB_Access Access to storage media in the ABox system

The storage media in the ABox WAN and all the involved local ABox databases are physically protected and access-controlled so that they may be accessed only by authorised administrators.

OE.Admin_Info Information about secure administration

The administrators of the local ABox databases and of the WAN components need to be informed clearly about secure administration.

Secure usage includes:

- not to concede access to the components to unauthorised persons;
- not to tell sensitive data to others.

For administrators of a local ABox database secure usage particularly includes:

- in the software encryption case, to generate the cryptographic data (keys and initialisation vectors) in a secure manner, in particular to choose the encPasswords for the key generation with a minimum length of 50 keyboard entries (including usage of the shift key) and in a sufficiently random manner; to take care that the cryptographic data have no weaknesses;
- to introduce cryptographic data into the ABox database respectively into the crypto card in a manner that protects their integrity and confidentiality;
- to use the administrative interface of the ABox database in a way that maintains the security of the sensitive data.

OE.Administrators Trustworthiness of database administrators

The authorised administrators of the involved local ABox databases and of the WAN components use and administer the components in accordance with the guidance and all other security instructions.

OE.WAN Security of transmission lines

All the transmission lines within the ABox WAN and the involved local ABox databases are protected against tapping of the transmitted data. Only authorised administrators have the right to access these data.

4.3 Security Objectives Rationale

38 The following table shows, which objectives for the TOE and the environment support which OSP, help to avert which threat and correspond to which assumption. The table shows, that for every OSP, threat and assumption there is at least one objective and vice versa.

	OT.Cryptography	OT.Stream_Process	OT.Administration	OE.ABox_Access	OE.User_Info	OE.Users	OE.OSE	OE.ABox	OE.DB_Access	OE.Admin_Info	OE.Administrators	OE.WAN
OSP.Desens	X	X					X	X				
OSP.Conceal		X					X	X				
OSP.Administration			X									
T.Compromise	X	X		X	X	X	X	X	X	X	X	X
A.Users						X						
A.Administrators											X	
A.ABox_Access				X								
A.WAN									X	X	X	X

Table 5: Mapping of objectives to OSPs, threats, assumptions

39 The following text describes for every OSP, threat and assumption, how they are covered by Security Objectives.

40 The organisational security policy **OSP.Desens** “Desensitisation of data leaving the local ABox system” is implemented by the following TOE security objectives:

- OE.OSE ensures that data leaving the local ABox system is beforehand directed through the ABox Core.
- OE.ABox ensures that a genuine and correctly configured ABox is used.
- OT.Stream_Process ensures that data in an input stream marked as sensitive is desensitised either by pseudonymisation or by encryption.
- OT.Cryptography provides the cryptographic algorithms required for encryption.

41 The organisational security policy **OSP.Conceal** “Concealing of sensitive data in the output” is implemented by the following TOE security objectives:

- OE.OSE ensures that incoming data is directed through the ABox Core before being passed to the browser.
- OE.ABox ensures that a genuine and correctly configured ABox is used.
- OT.Stream_Process ensures that pseudonymised data in an output stream which the actual user is not authorised to see is replaced by a specified message (which is fixed per ABox) stating the non-availability of the data.

42 The organisational security policy **OSP.Administration** “Administration of User and Group Configurations” is implemented by the TOE security objective OT.Administration:

- Implementation of supervisor activities in Table 2:
  - He may define a user by creating a user configuration.
  - He may define a user group by creating a user group configuration.
  - He may define a group owner by modifying the user group configuration.
- Implementation of group owner and deputy activities in Table 2:
  - They may assign or revoke read or write membership to their own group by inserting or deleting read or write permission for their group into a user configuration.
  - They may configure parameters like the desensitisation method used for the group, the key in the case of encryption, etc. by modifying the configuration of their group. Deputy assignment may be done only by the group owner, not by the deputy.
- Users with write permission for more than one group may choose at the runtime which they want to exercise by changing their active write group.

43 The threat **T.Compromise** is countered by the following combination of objectives:

- Administrators in the system are presupposed not to compromise sensitive data by OE.Admin_Info and OE.Administrators.
- Attackers who are neither administrators nor users in the overall ABox system are not able to acquire sensitive data neither in cleartext nor in desensitised form since
  - they are not authorised to access any of the local ABox systems which are protected against unauthorised access by OE.ABox_Access;
  - the authorised users will not concede them access to the local ABox system and will not tell them their passwords or sensitive data by OE.Users;

- the transmission lines in the ABox WAN are protected against tapping by OE.WAN.

- A user of a local ABox system who is not administrator

  - is presupposed not to compromise the sensitive data he is authorised to see by OE.User_Info and OE.Users;

  - is not able to acquire sensitive data he is not authorised to see in cleartext and in the case of pseudonymised data even not the pseudonyms within his local ABox system since

    - if output stream containing such data arrives at the local ABox system, it is (according to OE.OSE) protected against disclosure by the operational system environment and passed to the ABox Core which is genuine and correctly configured according to OE.ABox;

    - the ABox Core ensures that sensitive data will not be output to the unauthorised user in cleartext by OT.Stream_Process;

    - the ABox Core ensures that pseudonyms will not be output to the unauthorised user by OT.Stream_Process (instead a specific message will be output stating the non-availability of the data);

    - the transmission lines within the local ABox system are protected against tapping by OE.WAN;

    - the user is not authorised to access the local ABox database which is protected against unauthorised access by OE.DB_Access;

    - the administrators of the local ABox database will not tell him such data or concede to him logical access rights to such data or physical access to the database by OE.Admin_Info and OE.Administrators;

  - is not able to exploit the cryptograms he can see because of the strength of the applied cryptographic algorithms (OT.Cryptography and OT.Stream_Process), because by OE.Admin_Info and OE.Administrators administrators take care for the secrecy and strength of the applied keys, and because by OE.User_Info users will not encipher field content when the length of the text would give away inadmissible information about the contents;

  - is not able to acquire sensitive data neither in cleartext nor in desensitised form outside of his local ABox system for the analogous reasons as given above for attackers which are neither administrators nor users in the overall ABox system.

44 The assumption **A.Users** “Trustworthiness of ABox users” is fully covered by OE.Users, which is essentially identical to A.Users.

45 The assumption **A.Administrators** “Trustworthiness of database administrators” is fully covered by OE.Administrators, which is essentially identical to A.Administrators.

46 The assumption **A.ABox_Access** “Access to the local ABox system” is fully covered by OE.ABox_Access, which is essentially identical to A.ABox_Access.

47 The assumption **A.WAN** “Security of the ABox WAN” is countered by the following combination of objectives:

- OE.DB_Access and OE.WAN ensure that the storage media and the transmission lines in the ABox WAN are access-controlled, and that only authorised administrators have access right.

- Administrators in the system are trustworthy and will not to compromise sensitive data by OE.Admin_Info and OE.Administrators.

5 Security Requirements

48 The CC allows several operations to be performed on functional requirements; refinement, selection, assignment, refinement and iteration are defined in section 4.4.1.3.2 of Part 1 of the CC [5]. Each of these operations is used in this ST.

49 The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is either

- denoted by the word “Refinement” in bold text and an explication following or
- included in text as underlined text and marked by a footnote.

50 The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections that have been made by the ST authors are denoted as underlined text and the original text of the component is given by a footnote. Selections to be filled in by the ST author appear in square brackets with an indication that a selection is to be made, [selection:], and are italicised.

51 The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments that have been made by the ST authors are denoted by showing as underlined text and the original text of the component is given by a footnote. Assignments to be filled in by the ST author appear in square brackets with an indication that an assignment is to be made [assignment:], and are italicised.

52 The **iteration** operation is used when a component is repeated with varying operations. Iteration is denoted by showing a slash “/”, and the iteration indicator after the component identifier.

5.1 Security Functional Requirements for the TOE

53 This section on security functional requirements (SFR) for the TOE is divided into sub-sections following the main security functionality. They are usually ordered as in CC Part 2 [6].

5.1.1 Cryptographic support (FCS)

5.1.1.1 Cryptographic key generation (FCS_CKM.1)

54 The TOE shall meet the requirement “Cryptographic key generation (FCS_CKM.1)” as specified below (Common Criteria Part 2).

FCS_CKM.1 Cryptographic key generation

Hierarchical to: No other components.

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm OpenSSL_key_generation[1] and specified cryptographic key sizes 112, 128, 192 or 256 bit[2] that meet the following: none[3].

Dependencies: [FCS_CKM.2 Cryptographic key distribution or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

Application note 2: This SFR requires the TOE to implement the key generation by the mechanism OpenSSL_key_generation from the OpenSSL library. No compliance to a published standard is claimed . However, the mechanism in the form applied by the TOE is rated as SOF-basic and will be investigated in the strength of functions analysis.

5.1.1.2 Cryptographic operation (FCS_COP.1)

55 The TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below (Common Criteria Part 2). The iterations are caused by different cryptographic algorithms to be implemented by the TOE.

FCS_COP.1/AES Cryptographic operation – AES

Hierarchical to: No other components.

FCS_COP.1.1/ AES The TSF shall perform encryption and decryption[4] in accordance with a specified cryptographic algorithm AES in CFB and CBC mode[5] and cryptographic key sizes 128, 192 or 256 bit[6] that meet the following: FIPS 197 [9] and NIST 800-38A [11][7].

[1] [assignment: *cryptographic key generation algorithm*]

[2] [assignment: *cryptographic key sizes*]

[3] [assignment: *list of standards*]

[4] [assignment: *list of cryptographic operations*]

[5] [assignment: *cryptographic algorithm*]

[6] [assignment: *cryptographic key sizes*]

[7] [assignment: *list of standards*]

Dependencies: [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1/TDES Cryptographic operation – TDES

Hierarchical to: No other components.

FCS_COP.1.1/ TDES: The TSF shall perform encryption and decryption[8] in accordance with a specified cryptographic algorithm TDES in CFB and CBC mode[9] and cryptographic key sizes 112 bit[10] that meet the following: FIPS 46-3 [10] and NIST 800-38A [11][11].

Dependencies: [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

5.1.2 User data protection (FDP)

Management data

56 The Security Function Policy SFP_Administration, which is defined in the security objective OT.Administration (section 4.1), is used in the requirements “Subset access control (FDP_ACC.1)” and “Security attribute based access control (FDP_ACF.1)”. Therefore the following SFRs simply refer to this policy in all assignments. Note that all subjects, objects, security attributes, and operations occurring in these SFRs are defined already in this policy.

57 The access control policy SFP_Administration is only defined for the end usage phase of the TOE.

5.1.2.1 Subset access control (FDP_ACC.1)

58 The TOE shall meet the requirement “Subset access control (FDP_ACC.1)” as specified below (Common Criteria Part 2).

[8] [assignment: *list of cryptographic operations*]

[9] [assignment: *cryptographic algorithm*]

[10] [assignment: *cryptographic key sizes*]

[11] [assignment: *list of standards*]

FDP_ACC.1 Subset access control

Hierarchical to: No other components.

FDP_ACC.1.1 The TSF shall enforce the SFP_Administration[12] on all subjects, information, and operations defined by SFP_Administration[13].

Dependencies: FDP_ACF.1 Security attribute based access control

5.1.2.2 Security attribute based access control (FDP_ACF.1)

59 The TOE shall meet the requirement "Security attribute based access control (FDP_ACF.1)" as specified below (Common Criteria Part 2).

FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

FDP_ACF.1.1 The TSF shall enforce the SFP_Administration[14] to objects based on the following: all subjects and objects as defined in SFP_Administration[15].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: as defined in SFP_Administration[16].

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following rules: none[17].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following rules: none[18].

---

[12] [assignment: *access control SFP*]

[13] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

[14] [assignment: *access control SFP*]

[15] [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*]

[16] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

[17] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[18] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

User data

60 The Security Function Policy SFP_Stream_Process, which is defined in the security objective OT.Stream_Process (section 4.1), is used in the requirements “Subset Information flow control (FDP_IFC.1)”, “Simple security attributes (FDP_IFF.1)” and “Basic data exchange confidentiality (FDP_UCT.1)”. Therefore the following SFRs simply refer to this policy in all assignments. Note that all subjects, objects, security attributes, and operations occurring in these SFRs are defined already in this policy.

61 The information flow control policy SFP_Stream_Process is only defined for the end usage phase of the TOE.

5.1.2.3 Subset Information flow control (FDP_IFC.1)

62 The TOE shall meet the requirement “Subset Information flow control (FDP_IFC.1)” as specified below (Common Criteria Part 2).

FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

FDP_IFC.1.1 The TSF shall enforce the SFP_Stream_Process[19] on all subjects, information, and operations defined by SFP_Stream_Process[20].

Dependencies: FDP_IFF.1 Simple security attributes

5.1.2.4 Simple security attributes (FDP_IFF.1)

63 The TOE shall meet the requirement “Simple security attributes (FDP_IFF.1)” as specified below (Common Criteria Part 2).

FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

[19] [assignment: *information flow control SFP*]

[20] [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]

FDP_IFF.1.1 The TSF shall enforce the SFP_Stream_Process[21] based on the following types of subject and information security attributes: all subjects and information with their security attributes as defined in SFP_Stream_Process[22].

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: as defined in SFP_Stream_Process[23].

FDP_IFF.1.3 The TSF shall enforce the no additional information flow control SFP rules[24].

FDP_IFF.1.4 The TSF shall provide the following no additional SFP capabilities[25].

FDP_IFF.1.5 The TSF shall explicitly authorise an information flow based on the following rules: none[26].

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules: none[27].

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

5.1.2.5 Inter-TSF-Transfer (FDP_UCT.1)

Application note 3: FDP_UCT.1 requires the TOE to protect the confidentiality of sensitive user data transmitted between the TOE and a connected device. The rules for the data transfer are defined in the security policy SFP_Stream_Process defined in the section 4.1.

64 The TOE shall meet the requirement “Basic data exchange confidentiality (FDP_UCT.1)” as specified below (Common Criteria Part 2).

FDP_UCT.1 Basic data exchange confidentiality

Hierarchical to: No other components.

---

[21] [assignment: *information flow control SFP*]

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

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

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

[25] [assignment: *list of additional SFP capabilities*]

[26] [assignment: *rules, based on security attributes, that explicitly authorise information flows*]

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

FDP_UCT.1.1 The TSF shall enforce the SFP_Stream_Process[28] to be able to transmit and receive[29] objects in a manner protected from unauthorised disclosure.

Dependencies: [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

5.1.3 Security Management (FMT)

5.1.3.1 Management of security attributes (FMT_MSA.1)

The TOE shall meet the requirement "Management of security attributes (FMT_MSA.1)" as specified below (Common Criteria Part 2).

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1 The TSF shall enforce the SFP_Administration[30] to restrict the ability to modify, delete[31], create[32] the security attributes group and user configuration data[33] to administrators (supervisors, group owners, deputies) and users[34].

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles

5.1.3.2 Specification of Management Functions (FMT_SMF.1)

The TOE shall meet the requirement "Specification of Management Functions (FMT_SMF.1)" as specified below (Common Criteria Part 2).

---

[28] [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

[29] [selection*: transmit, receive*]

[30] [assignment: *access control SFP and/or information flow control SFP*]

[31] [selection: *change_default, query, modify, delete, clear,* [assignment: *other operations*]]

[32] [assignment: *other operations*]

[33] [assignment: *list of security attributes*]

[34] [assignment: *the authorised identified roles*]

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: Administration of user and group configurations[35].

Dependencies: No dependencies

5.1.3.3 Security Management Roles (FMT_SMR.1)

The TOE shall meet the requirement “Specification of Management Roles (FMT_SMR.1)” as specified below (Common Criteria Part 2).

FMT_SMR.1 Security Roles

Hierarchical to: No other components.

FMT_SMR.1.1 The TSF shall maintain the roles: Supervisor, group owner, deputy, ABox user[36].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Dependencies: FIA_UID.1 Timing of identification

5.2 Security Assurance Requirements for the TOE

65 The assurance components for the evaluation of the TOE and its development and operating environment are those taken from the

Evaluation Assurance Level 3 (EAL3) with no augmentations.

66 The minimum strength of function is SOF-basic. This Security Target does not contain any security functional requirement for which an explicit strength of function claim is required.

[35] [assignment: *list of security management functions to be provided by the TSF*]

[36] [assignment: *the authorised identified roles*]

5.3 Security Functional Requirements for the Environment

67 This Security Target does not describe security functional requirements for the IT environment.

5.4 Explicitly stated Security Requirements

68 This Security Target does not state explicitly any IT security requirements, but only uses those defined in CC Part 2 and 3.

5.5 Security Requirements Rationale

5.5.1 Security Functional Requirements Coverage

69 The following table shows, which SFRs for the TOE support which security objectives of the TOE. The table shows, that every objective is supported by at least one SFR and that every SFR supports at least one objective.

	FCS_CKM.1	FCS_COP.1/AES	FCS_COP.1/TDES	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FDP_UCT.1	FMT_MSA.1	FMT_SMF.1	FMT_SMR.1
OT.Cryptography	X	X	X								
OT.Stream_Process	X	X	X			X	X	X			
OT.Administration				X	X				X	X	X

Table 6: Coverage of Security Objectives for the TOE by SFRs

5.5.2 Functional Requirements Sufficiency

70 The security objective OT.Stream_Process is the central security requirement for the TOE. Therefore it is supported by most of the SFRs. It is mainly implemented by

(i) the SFRs FDP_IFC.1, FDP_IFF.1, and FDP_UCT.1 which require to implement the processing of information flow according to the security policy SFP_Stream_Process, which is defined in OT.Stream_Process,

and supported by

(ii) the SFRs of the FCS class including the key generation algorithm and the cryptographic algorithms AES and Triple-DES, which implement the key generation and the encryption and decryption operations that may be required when processing streams.

71 The TOE security objective OT.Cryptography is implemented by the SFRs of the FCS class. These include the cryptographic algorithms AES and Triple-DES and the key generation algorithm.

72 The TOE security objective OT.Administration is mainly implemented by the SFR FMT_MSA.1 and supported by the SFRs FMT_SMF.1 which provides the required functionality and by FMT_SMR.1 which implements the ability to differentiate the involved security roles.

5.5.3 Dependency Rationale

SFR	Dependencies	Support of the Dependencies
FCS_CKM.1	[FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation]	Fulfilled (FCS_COP.1)
	FCS_CKM.4 Cryptographic key destruction	Justification 1 for non-satisfied dependencies
	FMT_MSA.2 Secure security attributes	Justification 2 for non-satisfied dependencies
FCS_COP.1 /AES	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation]	Fulfilled (FCS_CKM.1)
	FCS_CKM.4 Cryptographic key destruction	Justification 1 for non-satisfied dependencies
	FMT_MSA.2 Secure security attributes	Justification 2 for non-satisfied dependencies
FCS_COP.1 /TDES	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation]	Fulfilled (FCS_CKM.1)
	FCS_CKM.4 Cryptographic key destruction	Justification 1 for non-satisfied dependencies
	FMT_MSA.2 Secure security attributes	Justification 2 for non-satisfied dependencies
FDP_ACC.1	FDP_ACF.1 Security attribute based access control	Fulfilled
FDP_ACF.1	FDP_ACC.1 Subset access control	Fulfilled
	FMT_MSA.3 Static attribute initialisation	Justification 3 for non-satisfied dependencies
FDP_IFC.1	FDP_IFF.1 Simple security attributes	Fulfilled
FDP_IFF.1	FDP_IFC.1 Subset information flow control	Fulfilled
	FMT_MSA.3 Static attribute initialisation	Justification 4for non-satisfied dependencies

SFR	Dependencies	Support of the Dependencies
FDP_UCT.1	[FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path]	Justification 5 for non-satisfied dependencies
	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Fulfilled (FDP_IFC.1)
FMT_MSA.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Fulfilled (FDP_ACC.1)
	FMT_SMF.1 Specification of Management Functions	Fulfilled
	FMT_SMR.1 Security Roles	Fulfilled
FMT_SMF.1	-	-
FMT_SMR.1	FIA_UID.1 Timing of identification	Justification 6 for non-satisfied dependencies

Table 7: Dependency rationale overview

Justification for non-satisfied dependencies:

No. 1: Keys are not destructed in the ABox system since they may be needed for the decryption of “old” data.

No. 2: The security attributes concerned by FCS_CKM.1, FCS_COP.1/AES and FCS_COP.1/TDES are the group specific “encryption passwords” from which the keys and initialisation vectors are generated. These passwords must be chosen “sufficiently random” in order that the keys and initialisation vectors are secure. This is ensured by organisational measures, namely the guidance shall instruct the administrators how to accomplish such a choice.

No. 3: The security attributes concerned by FDP_ACC.1 are the user and group configuration data. These are initialised by administrators on creation of user and group configuration data.

No. 4: The security attributes concerned by FDP_IFC.1 are the input markers. These are initialised by users on creation of an input stream.

No. 5: A trusted channel is not necessary here since sensitive data exist outside of the local ABox system only in a desensitised form which is not readable by unauthorised users; the desensitation is sufficient to uphold the confidentiality of the data.

No. 6: The identification and authentication of users is done by the operating system environment which is not part of the TOE. Only requests of authenticated users come up to the TSF. So all TSF-mediated actions take place automatically after identification and authentication of the user.

5.5.4 Rationale for the Assurance Requirements

73 The EAL3 was chosen to permit a developer to gain maximum assurance from positive security engineering at the design stage without substantial alteration of existing sound development practices. EAL3 is applicable in those circumstances where developers or users require a moderate level of independently assured security and require a thorough investigation of the TOE and its development without substantial re-engineering.

74 The minimal strength of function “basic” was selected to ensure resistance against direct attacks with low potential on functions based on probabilistic or permutational mechanisms.

5.5.5 Security Requirements – Mutual Support and Internal Consistency

75 The following part of the security requirements rationale shows that the set of security requirements for the TOE consisting of the security assurance requirements (SARs) and the security functional requirements (SFRs) together forms a mutually supportive and internally consistent whole.

76 The analysis of the TOE’s security requirements with regard to their mutual support and internal consistency demonstrates:

- The assurance class EAL3 is an established set of mutually supportive and internally consistent assurance requirements.
- The dependency analysis in section 5.5.3 for the security functional requirements shows that the basis for mutual support and internal consistency between all defined functional requirements is satisfied. All dependencies between the chosen functional components are analysed, and non-dissolved dependencies are appropriately explained.
- The following additional reasons support consistency and mutual supportiveness of the SFRs:
  - The chosen SFRs of class FCS implement the cryptographic algorithms as required by the ABox specifications.
  - The chosen SFRs of the class FDP support the access control policy SFP_Administration as defined in the objective OT.Administration and the information flow control policy SFP_Stream_Process as defined in the objective OT.Stream_Process.
  - The chosen SFRs of the class FMT implement the access control policy SFP_Administration, the required TOE management functions and the recognition of security roles.

  In detail these connections between the SFRs can be seen from section 5.5.2.
- Inconsistency between functional and assurance requirements could only arise if there are functional-assurance dependencies which are not met. Furthermore, as discussed in section 5.5.4, the chosen assurance components are adequate for the functionality of the TOE. So the assurance requirements and security functional

requirements support each other and there are no inconsistencies between the goals of these two groups of security requirements.

6 TOE Summary Specification

6.1 TOE Security Functions (TSF)

77 The following list gives an overview of the main Security Functions provided by the TOE during the usage phase. In order to refer to these functions, short identifiers are defined:

TSF_Authorisation: The TOE implements an authorisation function. This function receives the user identifier and returns the group of the user which is set active at the time in the session context.

TSF_Key_Generation: The TOE implements a key generation function. This function receives a password and the target cryptographic algorithm AES128, AES192, AES256 or Triple DES, and returns a key and an initialisation vector appropriate for the algorithm.

TSF_Crypt: The TOE implements a function for encryption and decryption. This function receives a data block, a user group to which is assigned a cryptographic method, the cryptographic algorithm AES128, AES192, AES256 or Triple DES and the mode of application CFB or CBC to be applied, the instruction to encrypt respectively decrypt, and in the case of decryption a key usage counter. The function loads from the ABox database the group specific key, the original initialisation vector, and in the case of encryption the key usage counter . The function transforms the original into the actual initialisation vector by integrating the key usage counter and returns the result of the encryption respectively decryption of the data block with the indicated cryptographic algorithm, mode of application and the actual initialisation vector.

TSF_Pseudonymisation: The TOE implements a pseudonymisation function. This function receives a data block, a user group to which is assigned pseudonymisation, and an “admissible number” (how many different pseudonyms may be assigned to the same data block) and returns a pseudonym for the data block.
In more detail, the function determines whether already pseudonyms have been assigned to this data block, and in that case how many. If the admissible number has already been reached, one of these pseudonyms is selected by the function and returned. Otherwise a new pseudonym is created (by an inquiry of the ABox database it is checked whether the pseudonym has already been assigned, in that case the creation process is repeated until a new pseudonym has been found). The new pseudonym is returned and is stored in the ABox database along with the contents it stands for and the user group it belongs to.

TSF_Pseudonym-Resolution: The TOE implements a pseudonym resolution function. This function receives a pseudonym and a user identifier and returns the contents the pseudonym stands for. For the pseudonym resolution, all the user’s read groups are taken into account.

TSF_Administration: The TOE implements a function which provides users with the possibility to manage group or user configuration data in accordance with the access control policy SFP_Administration (see section 4.1).

The following Security Functions are realised by probabilistic or permutational mechanisms: TSF_Key_Generation, TSF_Crypt, TSF_Pseudonymisation. They are rated as SOF-basic and will be investigated in the strength of functions analysis.

6.2 TOE Security Functions Rationale

6.2.1 TOE Security Functions Coverage

78 The following table shows, which TOE Security Functions support which SFRs for the TOE. The table shows, that every SFR is supported by at least one Security Function and that every Security Function supports at least one SFR.

	TSF_Authorisation	TSF_Key_Generation	TSF_Crypt	TSF_Pseudonymisation	TSF_Pseudonym-Resolution	TSF_Administration
FCS_CKM.1		X				
FCS_COP.1/AES			X			
FCS_COP.1/TDES			X			
FDP_ACC.1						X
FDP_ACF.1						X
FDP_IFC.1	X		X	X	X	
FDP_IFF.1	X		X	X	X	
FDP_UCT.1	X		X	X	X	
FMT_MSA.1						X
FMT_SMF.1						X
FMT_SMR.1						X

Table 8: Coverage of SFRs by TOE Security Functions

6.2.2 TOE Security Functions Sufficiency

79 The key management SFR FCS_CKM.1 is directly implemented by the Security Function TSF_Key_Generation which receives the encPassword and the target cryptographic algorithm AES128, AES192, AES256 or Triple DES, and returns a key and an initialisation vector appropriate for the algorithm.

80 The cryptographic operation SFRs FCS_COP.1/AES, FCS_COP.1/TDES are all implemented by the Security Function TSF_Crypt which after importing the relevant key and building the relevant initialisation vector applies the selected encryption or decryption algorithm AES128, AES192, AES256, or TDES.

81 The user data protection SFRs FDP_ACC.1, FDP_ACF.1 both deal with the enforcement of the policy SFP_Administration (see section 4.1). This policy is directly implemented by the Security Function TSF_Administration.

82 The user data protection SFRs FDP_IFC.1, FDP_IFF.1, FDP_UCT.1 all deal with the enforcement of the policy SFP_Stream_Process (see section 4.1). This policy is implemented by the Security Functions TSF_Authorisation (to load the active user group), TSF_Crypt (for an input or output stream with an assigned cryptographic method), TSF_Pseudonymisation (for an input stream with pseudonymisation assigned), and TSF_Pseudonym-Resolution (for an output stream with pseudonymisation assigned).

83 The management SFRs FMT_MSA.1 “Management of security attributes”, FMT_SMF.1 “Specification of Management Functions”, FMT_SMR.1 “Security Roles” are implemented by the Security Function TSF_Administration which checks the security role of the requesting user and offers him the corresponding management functions.

6.2.3 TOE Security Functions – Mutual Support and Internal Consistency

84 The detailed description of the TOE Security Functions in section 6.1 and their mutual support to implement the SFRs in section 6.2.2 demonstrate how the defined functions work together and support each other. Furthermore, this description shows that no inconsistencies exist.

6.3 Assurance Measures

85 Appropriate assurance measures will be employed by the developer of the TOE to satisfy the security assurance requirements defined in section 5.2.

86 The ABox is developed by T-Systems International GmbH in Leinfelden-Echterdingen, Germany. The developer team works in the T-Systems building in the Fasanenstr. 5. The building is access controlled by physical and organisational measures.

87 For configuration management of the ABox items, in particular for version control, the system “CVS” is used. In the configuration management plan, the development roles (leader of development, developer, administrator, tester) and their responsibilities, the

life cycle phases of CM items and TOE releases together with the corresponding acceptance procedures are defined.

88 The finished TOE is delivered on a CD to the customer by a person trusted by the developer. The CD is provided with an MD5 checksum which enables the customer to verify the integrity of the delivery.

89 For the evaluation of the TOE, the developer will provide appropriate documents describing these measures in more detail and containing further information supporting the check of the conformance of these measures against the claimed assurance requirements.

90 The following table gives a mapping between the assurance requirements and the documents containing the relevant information for the respective requirement. The table contains only the directly related documents, references to further documentation can be taken from the mentioned documents.

Assurance Class	**Family**	**Developer input for the evaluation**
ACM (Configuration Management)	ACM_CAP	Configuration Management for ABox 1.0
	ACM_SCP	Configuration Management for ABox 1.0
ADO (Delivery and Operation)	ADO_DEL	Delivery Procedure for ABox 1.0
	ADO_IGS	Installation Guide for ABox 1.0
ADV (Development)	ADV_FSP	Functional Specification for ABox 1.0
	ADV_HLD	High Level Design for ABox 1.0
	ADV_RCR	Functional Specification for ABox 1.0 High Level Design for ABox 1.0
AGD (Guidance documentation)	AGD_ADM	(Part of the User Guidance for ABox 1.0)
	AGD_USR	User Guidance for ABox 1.0
ALC (Life cycle support)	ALC_DVS	Security Development Environment for ABox 1.0
ATE (Tests)	ATE_COV	Test Documentation for ABox 1.0
	ATE_DPT	Test Documentation for ABox 1.0
	ATE_FUN	Test Documentation for ABox 1.0
	ATE_IND	(The TOE suitable for testing)
AVA (Vulnerability Assessment)	AVA_MSU	User Guidance for ABox 1.0
	AVA_SOF	Strength of Function Analysis for ABox 1.0
	AVA_VLA	Vulnerability Analysis for ABox 1.0

6.4 Assurance Measures Rationale

91 The assurance measures of the developer as mentioned in section 6.3 are considered to be suitable and sufficient to meet the CC assurance level EAL3 as claimed in section 5.2. Especially the deliverables listed in section 6.3 are seen to be suitable and sufficient to document the fulfilment of the assurance requirements in detail.

7 Annexes

7.1 Glossary and Acronyms

Some types of terms are not described here, but at specific places in the text:

- The services provided by the TOE are defined in section 6.1.
- Assets (sensitive data) protected by the TOE are defined in section 3.1.1, Table 1.
- The subjects interacting with the TOE are defined in section 3.1.2, Table 2.

Term	Definition
ABox	The software TOE of this ST which is described in chapter 2 consisting of the ABox Core, the ABox Admin Client and belonging guidance documentation.
ABox Admin Client	The administration component belonging to the TOE by which the TOE data in the local ABox data base are managed.
ABox Core	The part of the TOE doing the key generation and the data processing at the runtime.
ABox database	The local database holding the management data belonging to the ABox Core. The data are managed via the ABox Admin Client.
Desensitisation	Reversible method used by the ABox Core to render sensitive data into non-sensitive data; either encryption or pseudonymisation.
Input markers	Special signs in an input stream that reaches the ABox Core marking “sensitive” areas.
Input stream	Formatted user input to the ABox WAN (which is directed through the ABox Core).
local ABox system	The local environment of the TOE including the TOE itself, its operating system environment, the ABox database, optionally a crypto card, the user input/output devices, the connection lines between these devices, further including the protected room(s) in which these devices and their connection lines are located, and in which the devices are operated.
MIME	MIME (Multipurpose Internet Mail Extensions) is an Internet Standard for the format of e-mail. It is also used in communication protocols such as HTTP, which requires that data be transmitted in the context of e-mail-like messages. MIME defines a collection of headers for specifying additional attributes of the message.
MIME type	The MIME type is an attribute used in MIME headers specifiying the content type of the message. It consists of a type and a subtype. At present, the following types are defined: text, image, video, audio, application, multipart, message, model. There are more than one hundred defined subtypes.
Operating system environment (OSE)	Operating system and application software of the server the Abox Core is installed on.
Output markers	Special signs in an input stream that leaves the ABox Core respectively in an output stream that reaches the ABox Core marking “desensitised” areas in the stream.
Output stream	Response coming from the ABox WAN to a user read request (which is directed through the ABox Core).

Term	Definition
Pseudonymisation	Reversible method used by the ABox Core to render sensitive data into non-sensitive data: A portion of data is in an unpredictable way assigned a byte string (the so-called “pseudonym”), the meaning of which is stored in the belonging ABox database .
TSF data	Data created by and for the TOE, that might affect the operation of the TOE (CC term).
User data	Data created by and for the user, that does not affect the operation of the TSF (CC term).
User security profile	The read and write groups a user is authorised for, which write group (if any) is active, and a comprehensive pseudonym list consisting of all pseudonyms and their resolutions belonging to any of the user groups assigned to pseudonymisation.

Acronyms

Acronyms	Term
*A.***	Naming convention for assumptions in this ST, e. g. A.Usage (see section 3.4)
AES	“Advanced encryption standard” (symmetric cryptographic algorithm implemented by the TOE, also called Rijndael) existing in the variants AES128, AES192, AES256 (the number indicates the bit length of the used key)
CC	Common Criteria
CCIMB	Common Criteria Interpretation Management Board
DES-3, 3DES	“Data encryption standard 3” (symmetric cryptographic algorithm implemented by the TOE, also called Triple-DES)
EAL	Evaluation Assurance Level (CC term)
OSP	Organisational Security Policy (CC term)
*OSP.****	Naming convention for organisational security policies in this ST, e. g. OSP.Desens (see section 3.2)
*OT.****	Naming convention for security objectives for the TOE in this ST, e. g. OT.Stream_Process (see section 4.1)
PP	Protection Profile (CC term)
SAR	Security Assurance Requirement (CC term)
SFP	Security Functional Policy (CC term)
SFP_Administration	Name of the security functional policy defining how the ABox grants access to administrative functions. It is defined in OT.Administration (see section 4.1) and used by access control SFRs (see section 5.1)
SFP_Stream_Process	Name of the security functional policy defining how the ABox Core processes information flow. It is defined in OT.Stream_Process (see section 4.1) and used by information flow control SFRs (see section 5.1)
SFR	Security Functional Requirement (CC term)
ST	Security Target (CC term)
*T.****	Naming convention used for threats in this ST, e. g. T.Intercept (see section 3.3)

Acronyms	Term
TDES, Triple-DES	See DES-3
TOE	Target of Evaluation (CC term)
TSC	TSF Scope of Control (CC term)
TSF	Totality of the TOE Security Functions (CC term)
*TSF_****	Naming convention for the TOE Security Functions in this ST, e. g. TSF_Crypt (see section 6.1)

7.2 Reference Documents

Common Criteria

[1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.1, August 1999.

[2] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.1, August 1999.

[3] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.1, August 1999.

[4] Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 1.0, August 1999.

[5] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 2.2, January 2004.

[6] Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Requirements; Version 2.2, January 2004.

[7] Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; Version 2.2, January 2004

[8] Common Methodology for Information Technology Security Evaluation; Version 2.2, January 2004.

Cryptography

[9] Federal Information Processing Standards Publication (FIPS) Publication 197, Specification for the ADVANCED ENCRYPTION STANDARD (AES), 26.11.2001, U.S. Secretary of Commerce/National Institute of Standards and Technology (NIST).

[10] Federal Information Processing Standards Publication (FIPS) Publication 46-3, DATA ENCRYPTION STANDARD (DES), Reaffirmed 25.10.1999, U.S. Department of Commerce/National Institute of Standards and Technology.

[11] NIST Special Publication 800-38A, 2001 Edition, Recommendation for Block Cipher Modes of Operation, Methodes and Techniques, Morris Dworkin, December 2001, National Institute of Standards and Technology, Technology Administration, U.S. Department of Commerce.