Security Target

Bdrive Windows Client

developed by

Bundesdruckerei GmbH

Status:	Final
Classification	Public
Date:	2021-03-16
Version:	1.00
Author:	SRC/mfs, TÜViT/wgr / TÜViT/bay
QA:	SRC/job, SRC/ed, SRC/sch

Document history

Version	Date	Approval	Remark
1.00	2021-03-16	TÜViT/bay	Public release

Contents

1 ST Introduction 6
1.1 ST Reference 6
1.2 TOE Reference 6
1.3 TOE Overview 6
1.3.1 The Bdrive system 6
1.3.2 Bdrive Server & Identity Provider 7
1.4 TOE Description 8
1.4.1 General description 8
1.4.2 Security features 9
1.4.3 TOE deliverables 10
1.4.4 TOE Hardware and Software Environment 10
1.4.5 Connectivity Aspects 11
1.4.6 TOE Boundaries 11
1.4.7 TOE Delivery 12
2 CC Conformance 13
2.1.1 PP claim 13
2.1.2 Package claim 13
2.1.3 Conformance rationale 13
3 Security Problem Definition 14
3.1 Assets 14
3.2 Subjects, Objects and Operations 16
3.2.1 Subjects 16
3.2.2 Objects 17
3.2.3 Operations 17
3.3 Threats 18
3.3.1 T.UnauthLocalAccess 18
3.3.2 T.UnauthRemoteAccess 18
3.3.3 T.DisclosureKey 18
3.3.4 T.DisclosurePW 18
3.3.5 T.DisclosureMKey 18
3.3.6 T.Residual 19
3.4 Organizational Security Policies 19
3.4.1 OSP.Recovery 19
3.4.2 OSP.UserApproval 19
3.5 Assumptions 19
3.5.1 A.Installation 19
3.5.2 A.Credentials 19
3.5.3 A.Malware 20
3.5.4 A.Admin 20
3.5.5 A.User 20
3.5.6 A.Physical 20
3.5.7 A.TrustedBackend 20
4 Security Objectives 21
4.1 Security Objectives for the TOE 21
4.1.1 O.Keygen Key generation 21

4.1.2 O.Access Access to data 21
4.1.3 O.UserAuthentication Authentication of BDrive Users 21
4.1.4 O.Integrity Integrity of file contents and file information 21
4.1.5 O.Authentication Authentication of external entities 21
4.1.6 O.Residual Protection of residual information 21
4.2 Security Objectives for the Operational Environment 21
4.2.1 OE.Installation 21
4.2.2 OE.Credentials 22
4.2.3 OE.Malware 22
4.2.4 OE.Admin 22
4.2.5 OE.User 22
4.2.6 OE.Physical 22
4.2.7 OE.TrustedBackend 22
4.2.8 OE.MasterKey 22
4.3 Security Objective Rationale 22
5 Extended Components Definition 25
5.1 FCS_RNG Generation of random numbers 25
6 Security Requirements 26
6.1 Security functional policy (SFP) 26
6.2 Security Functional Requirements 27
6.2.1 Class FCS: Cryptographic support 27
6.2.2 Class FDP: User Data Protection 29
6.2.3 Class FIA: Identification and Authentication 31
6.2.4 Class FMT: Security Management 32
6.2.5 Class FTA: TOE Access 33
6.2.6 Class FTP: Trusted path/channels 34
6.3 Security Assurance Requirements 35
6.4 Security Requirements Rationale 35
6.4.1 Security Functional Requirements Rationale 35
6.4.2 Rationale for SFR dependencies 39
6.4.3 Security Assurance Requirements Rationale 41
7 TOE Summary Specification 42
7.1 Login 42
7.2 Logout 42
7.3 Management of access rights to files and folders 42
7.4 Generation of user-specific meta data 43
7.5 Generation of general meta data 43
7.6 Key generation 43
7.7 File encryption 43
7.8 File authentication 43
7.9 File fragmentation 43
7.10 Key encryption & decryption 44
7.11 Secure channels to other trusted IT products 44
7.12 Certificate Validation 44
Appendix 45
Bibliography 46

1 ST Introduction

1.1 ST Reference

Document Title:	Security Target Bdrive Windows Client
Document Version	Version 1.00
Document Date:	2021-03-16
Company:	Bundesdruckerei GmbH (BDR)
CC Version:	3.1, Revision 5
Evaluation Assurance Level:	EAL2

1.2 TOE Reference

TOE Name:	Bdrive Windows Client
TOE Version:	BDrive 3.50.89.4
Developer:	Bundesdruckerei GmbH
Product Type:	Cryptographic file scrambler for exchange of company data
Certification-ID:	BSI-DSZ-CC-1106

1.3 TOE Overview

1.3.1 The Bdrive system

The Bdrive system provides a solution for storing and sharing for files of all types. The system consists of the following components, each of them contributing to the security of the whole system:

1. Bdrive Client (the TOE which is a cryptographic file scrambler for exchange of company data (TOE type))
2. Bdrive Server, including the Access Control List (ACL) database
3. Identity Provider (IDP), including the certificate database
4. Web frontends for administration
5. Web Services for File Sharing (Linkshare, Droppad, Web Client)

The Bdrive Server and the IDP are server solutions that are located in secured operating environments at the premises of Bundesdruckerei GmbH.

The following entities outside of the Bdrive system are involved:

- PKI provided by D-TRUST (connected to the components IDP and to the Bdrive Client via OCSP)
- Cloud servers used as storage locations from several cloud providers (connected to the Bdrive Server and Client). To increase reliability and availability the number of physically separate servers should be as high as possible and of special importance when the cloud storage providers are selected.
- the workstation (incl. hardware, software and operating system) in the end-user environment where the Bdrive Client is installed

The TOE ensures:

- Confidentiality of the files during up- and download
- Integrity of the files

As a main feature of the Bdrive system, plain files of a user are available and decrypted only on the workstation the Bdrive Client is installed on, cf. Figure 1. The Bdrive Server, the IDP and the cloud storage locations have no access to the plain files of Users.

The BDrive System utilizes Erasure Encoding for forward error correction (erase encoding). Files encrypted by the Client will be split into n different chunks such that k < n arbitrarily chosen chunks suffice to restore the original encrypted file. That is achieved by computing m additional parity chunks, which can be used to replace at most m missing chunks. In order to ensure a high availability, these n chunks are stored on distinct cloud servers (see information on the cloud servers above).

The TOE is the Bdrive Client, which is a software solution delivered to the end-user and can be installed on devices running the operating system Windows 7 Professional, Windows 7 Enterprise and Windows 10, see sec. 1.4 for more details.

The Bdrive system allows for the recovery of file contents that belong to a specific company with the help of a so-called Company Masterkey. Please note that this recovery mechanism is not realized by the TOE itself but requires assistance from the Bdrive administration.

Additionally the TOE offers the ability to share files with users outside the company without access to own account utilizing the Linkshare functionality. In this scenario the TOE encrypts the metadata for exactly one file for a time-restricted virtual user, which allows him to download the encrypted file parts via link from the cloud and reassemble them in the browser. No access to other folders or files is granted and after expiring the link and the corresponding access keys will be automatically deleted.

The TOE also allows the user to receive files from users without an own account via Droppad. An upload link created by the TOE permits users in ownership of this link to drop files for the creator of the link, e.g. like a mailbox. The files are encrypted and split in chunks in the Browser but are only downloaded by the TOE after explicitly accepting them.

1.3.2 Bdrive Server & Identity Provider

Beyond the TOE, two other components of the Bdrive system are briefly described in the following:

Bdrive Server

- generation and distribution of storage coordinates on the Cloud Servers in form of Signed URLs; the Bdrive Server retrieves the storage coordinates for read/write access to the Cloud Servers via connections (5) in Figure 1. These so called Signed URLs are made available to the TOE via connection (1) in Figure 1. The TOE uses these Signed URLs in order to download/upload the encrypted file chunks via connection (6) in Figure 1 directly to the Cloud Storage Provider.
- (physical) RNG
- verification of the authentication token sent from the Bdrive Client via connection (1) in Figure 1 after initial client side TLS connection.
- management and storage of general Meta Data Arrays (MDAs, cf. Section 1.4.1); these general MDAs are exchanged between Bdrive Client and Bdrive Server via connection (1) in Figure 1.
- establishing TLS connection (see (3) in Figure 1) to IDP;
- management and storage of Access Control Lists (ACLs) for files and folders; Please be aware that die BDrive Client is used to generate invitations to a shared folder. The technical implementation for the ACLs is done on server side.

IDP (Identity Provider)

- supports the client in remote user authentication. There is one instance of the IDP for the following authentication method:
  - authentication by client certificate
- establishing of TLS connections to Bdrive Client (connection (2) in *Figure 1*) and Bdrive Server (connection (3) in Figure 1);
- connection to the PKI (external) including the revocation of certificates via connection (4) in Figure 1; The TOE itself has also a connection to the external PKI (7) to send OCSP requests to check the validity of certificates. Finally the TOE uses the services of the workstation as a platform (8).
- generation of certificate requests towards Certificate Service Manager (CSM), an external entity provided by D-Trust that receives these certificate requests. Certificates are issued by D-Trust upon successful verification of certification requests.

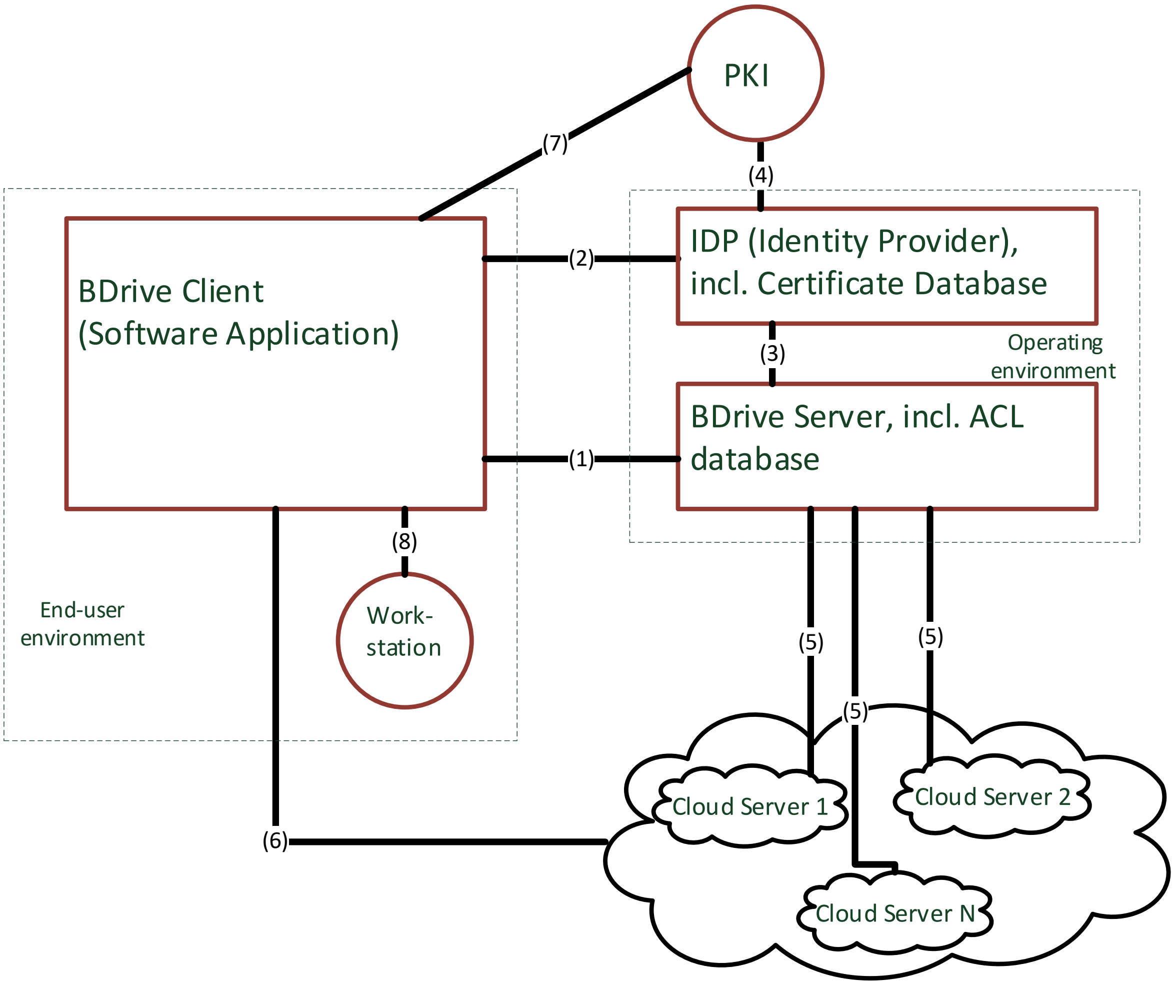


Figure 1: Overview of the Bdrive system components and the external entities involved.

1.4 TOE Description

1.4.1 General description

The TOE implements the Bdrive Client for the Windows platform. It implements a secure, distributed file storage allowing the following authentication mode (cf. the description in sec. 7.1):

- User authentication certificate plus password which protects the private part of the authentication key.

Each consumer device of a user receives a unique authentication certificate, and files are shared between all devices of the user. Optionally, a user has the possibility to share files/folders with several other users in the same company. Additionally the user can utilize Linkshare or Droppad functionality to exchange or receive files with arbitrary individuals outside the company context.

The storage scheme realizes forward error correction (erase coding) together with cryptographic means for encryption and authentication. As soon as the Bdrive Client has retrieved k<n file fragments, it is possible to restore the complete and plain file. The file authenticity is validated using HMAC.

It uses SQLite for state handling.

1.4.2 Security features

The TOE is the Bdrive Client, a software solution delivered to end-users for installation on devices with the operating system Windows.

The TOE provides the following security features:

- User Authentication and Session handling
- symmetric (AES-256) encryption & decryption of files
- file authentication (HMAC)
- forward error correction method (Erasure Coding) for files, i.e. splitting files into n certain fragments such that only k<n parts are necessary to recover the whole file.
- upload/download of encrypted file fragments according to storage coordinates provided by the Bdrive Server
- source of entropy, for generation of
    - symmetric keys for file authentication and encryption
    - initialization vectors
    - asymmetric key pairs for key encryption (encryption keys)
    - asymmetric key pairs for authentication towards IDP (authentication keys)
    - random padding bytes
    - as Botan input
- using a TLS connection, to fulfill a client-side TLS authentication
- asymmetric (RSA) encryption & decryption of symmetric file keys
- local management, import and secure storage of certificates and corresponding private keys, including checks for revocation of keys
- checks for revocation of certificates via OCSP interface of the PKI

The client generates two types of meta data arrays (MDA) which will be referred to in the sequel:

- **user specific MDA:** Contains the following information:
    - file name
    - local storage location

  - checksum of plain file

The user specific MDA for a file are encrypted and stored. Afterwards, the encrypted data is authenticated, and the resulting MAC is appended to the encrypted data. This resulting cryptogram (encrypted file incl. user specific MDA, appended MAC) is split according to the parameters of the forward error correction method (erase encoding) and then uploaded to the cloud storage servers.

- **general MDA:** Contains the following information:
  - filesize of the plain file
  - timestamp
  - assigned storage folder by pseudonym structure
  - filesize of the encrypted, non-fragmented file
  - filesizes and sha256 checksums[1] of the fragments after Erasure Coding
  - parameters of the used Erasure Coding method
  - storage coordinates on the different cloud servers
  - list of user IDs together with individually public key encrypted symmetric keys for each authorised user
    - For the sake of availability, the symmetric keys in the general MDA are additionally encrypted by the TOE with the so-called Company Masterkey. The Company Masterkey is the public key of an asymmetric key pair generated by a Bdrive-Admin of the corresponding customer company. This public key is then uploaded during the first login to the web administration front end (CWA) and is treated by the TOE like the public key of another user from the same company. In case a device and therefore the private key of a user of a particular company is lost, the corresponding Company Masterkey can be used in order to recover the symmetric keys that are needed for the decryption and integrity check of the encrypted file chunks. Please note that the decryption with the help of this Company Masterkey is not part of the TSF.

1.4.3 TOE deliverables

The TOE consists of

1. the installable program code of the Bdrive Client (Software Only) [“Bdrive-3.50.89.4-win64-signed.exe”, hash code: 27bc09450ccf94225b15ec7622a17a12a27fa0064b9ca4a7bc931541a8df25d6]
2. the Bdrive user’s guide [“Bdrive_Nutzerhandbuch_Release_3_50.pdf”, hash code: fbcac6a252eb7af54a952ae841d83b4f1e18db19c183a8c5cd01d4ddfc18e180]
3. an operating manual for Bdrive [“Betriebshandbuch-Bdrive-v-1.1.pdf”, hash code: 611a1324c2a947d4d58c532d20838b968c9f16842ba52797bdeede3164d04707]
4. and a user interface reference document [UI-Reference-Bdrive.pdf”, hash code: 8e6eb91d8662d8f8550cac804428ba51265961d26de5e0921f85cc512b754a83]

1.4.4 TOE Hardware and Software Environment

1.4.4.1 Hardware requirements

The TOE runs on personal computer systems. The system needs at least the requirements to run the operating systems as described in 1.4.4.2.

[1] Not TSF-relevant for authenticity

1.4.4.2 Software requirements

The version of Bdrive Client under this evaluation is provided for the following operating systems:

- Windows 7 Professional with Extended Security Updates,
- Windows 7 Enterprise with Extended Security Updates, or
- Windows 10.

The Bdrive Client works with all available file systems under the operating systems mentioned above.

1.4.5 Connectivity Aspects

The TOE communicates directly with TOE external components via network, cf. *Figure 1*. Connections with IDP and Bdrive Server require a secure, (mutually) authenticated channel that is established using TLS 1.2. The other components involved are a PKI and Cloud Servers. Although the TOE does not offer network services itself, safeguarding of the workstation is highly recommended (e.g. Firewall).

The workstation, where the TOE as a part of Bdrive system shall work, must have a network connection to the Bdrive Server and the IDP, providing the features as described in sec. 1.3.1. Additionally the TOE must be able to reach the Cloud Storage Providers to upload or receive chunks.

These data connections are secured by a Transport Layer Security (TLS) connection.

When the TOE is operated on a workstation within a network with connection to the Internet, a correctly installed and maintained firewall system shall be established in order to prevent access to this workstation's hard disk(s) and memory by unauthorised individuals from outside.

1.4.6 TOE Boundaries

1.4.6.1 TOE Representation

The Bdrive Client is a pure software product.

1.4.6.2 TOE External interfaces

The TOE provides the following external interfaces:

- An interface to the filesystem of the underlying operating system for the processing of hard disk read/write accesses.
- An interface to the Bdrive Server, the Bdrive Server being not part of the TOE.
- An interface to the IDP, the IDP being not part of the TOE.
- An interface to the user: this is a graphical user interface, where the user can read data on the screen and can enter data via keyboard and other input devices (e.g. mouse). This interface is used for user identification/authentication and for TOE administration support.

- An interface to the CSP to upload and download data chunks

1.4.7 TOE Delivery

The delivery of the TOE is secured in a way that any user can determine the integrity and authenticity of the software package. After the download from the service portal of the Bundesdruckerei GmbH the user can check the integrity and authenticity by checking the validity of the code signing certificate during installation (which will result in an error message when not valid) and ensure that the issuer of the software is “Bundesdruckerei GmbH”. The valid code signing and the issuer of the software can also be later checked on the “Bdrive.exe” executable in the “properties” dialog, in th “digital signatures” tab.. Only authenticated users have access to the service portal and can download the client and distribute it in their organization.

2 CC Conformance

This security target claims conformance to

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model; Version 3.1, Revision 5 [CC_P1]
- Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components; Version 3.1, Revision 5 [CC_P2]
- Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components; Version 3.1, Revision 5 [CC_P3]

as follows

- Part 2 extended (cf. chapter 5),
- Part 3 conformant.

The

- Common Methodology for Information Technology Security Evaluation, Evaluation methodology; Version 3.1, Revision 5, [CEM]

has to be taken into account.

2.1.1 PP claim

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

2.1.2 Package claim

This Security Target claims conformance to the assurance package EAL2.

2.1.3 Conformance rationale

This Security Target does not claim any conformance with any Protection Profiles. Therefore, no conformance rationale is provided here.

3 Security Problem Definition

3.1 Assets

The primary assets to be protected by the TOE as long as they are in scope of the TOE are:

Asset	Protection			Description
	Conf.	Int.	Auth.	
Contents of the plain files	X	X		These are the user data contained in the unencrypted, non-fragmented files that the end-user has decided to transfer to the Bdrive system. The user makes this decision by using the Bdrive Client's user interface, which allows the user to put folders and the files contained in these folders under control of the Bdrive Client. It has to be noted that these files remain unchanged (in particular unencrypted) on the filesystem as they were before the user's decision. In this form, they are not in the scope of the TOE. However, by putting the files under control of the Bdrive system, their content is potentially accessible by different means than access to the local filesystem. The TOE is responsible for the achievement CI-protection of these contents insofar as this primary asset or the secondary assets are in its scope. Type: user data These data are protected by the TOE.
User specific MDA associated to the plain files	X	X		See chapter 1.4.2. Type: user data These data are protected by the TOE.

Table 1: Primary Assets

The following secondary assets need to be protected by the Bdrive Client or the environment in order to achieve a sufficient protection of the primary assets:

Asset	Protection			Description
	Conf.	Int.	Auth.	
Private keys	X	X	X	User- and device-specific public/private key pairs are generated by the Bdrive Client when a new device is registered for a user. The private keys never leave the workstation where they have been created by the Bdrive Client, and they are protected by a password. The private key of an authorised user is needed in order to recover the symmetric keys from the general MDA that are needed for file decryption and file authentication. Type: TSF data These keys are protected by the TOE.

Protection passwords	X	X	X	The access to the user private key is protected by a password that is set by the user after the creation of such private keys. Type: TSF data These passwords are protected by the TOE.
authentication and encryption certificates		X	X	User- and device-specific public/private key pairs are generated by the Bdrive Client when a new device is registered for a user. The public keys are transferred to and signed by the PKI (external) via the IDP and the Bdrive Server. The encryption certificates are used to encrypt the symmetric keys for file encryption and file authentication. The authentication certificates are used for Client authentication towards IDP and authentication of file contents. Type: TSF data These certificates are protected by the TOE.
file encryption and authentication keys (FEAK)	X	X		These symmetric keys are generated by the Bdrive Client. They are used for file encryption (AES-256) and file authentication (HMAC). File authentication is performed on top of encryption. Type: TSF data The FEAK are protected by the TOE.
Company Masterkey	X[2]	X	X	A private/public key pair for each company. The public part is used to additionally encrypt the FEAK. By this method, the private part of this key pair may be used to recover the FEAK in case that a private key of a user gets lost. Type: TSF data The public part of the key is protected by the TOE. The private part of the key has to be protected by the environment (see OE.MasterKey).
general MDA		X		See p. 10 Type: user data These data are protected by environmental security measures (i.e. the TLS connection and secure storage on the server).
Authentication Token	X			By means of this authentication token, a trusted channel between Bdrive Client and Bdrive Server is established in the following way: On the basis of a trusted, mutually authenticated channel between Bdrive Client and IDP, the IDP generates this authentication token and retains records on its validity. When the Bdrive Client connects to the Bdrive Server, the client presents this token, the server checks the validity of this token towards IDP, and the IDP invalidates it. Please refer to the OpenID Connect process for more information. Type: TSF data The authentication token is protected by the TOE and the environmental security measures (i.e. the TLS connection).

Table 2: Secondary Assets

[2] Of course, confidentiality is only required for the private part of the key.

3.2 Subjects, Objects and Operations

3.2.1 Subjects

For the operational phase, this SPD considers the following subjects.

Role	Description
Authorized User / User	A person having performed a successful login to the Bdrive Client. The Bdrive system uses the user's email address as unique identifier for different users.
Unknown user	A person who has not (yet) performed a successful login to the Bdrive Client.
Attacker	A person or process trying to undermine the security policy, in particular the security properties of the assets. Please note that an attacker might appear in any role recognized by the TOE.
Bdrive Server, together with its associated components • Cloud storage locations • ACL database	External entity acting as a technical user. This component is described in sec. 1.3.2.
Company Administrator	Individual with additional rights to perform administrative tasks for the Bdrive system such as • allow access for new users / devices • remove access for users / devices • proof of authenticity of a new user towards IDP These functions are not made available by the TOE, but via web interface at the Bdrive Server that is accessible via soft-token (certificate based) with additional personal verification via d-trust process. Please note that the company administrator is a role of an external entity that is not recognized by the TOE, but by the independent web interface to the Bdrive Server.
IDP, incl. PKI and certificate database	External entity. This component is described in sec. 1.3.2.
the workstation	External entity. Workstation at the premises of the customer where the TOE is installed on. Workstation in the given sense means the whole equipment including hardware, software and operating system.

Table 3: Subjects

During registration to the Bdrive system, each user (identified by the user's email address) is associated with exactly one company (identified by its company ID).

In the Bdrive system, every group (of users) and every user owns a unique virtual folder called *root node*. A user may be a member of several groups. All subfolders and their files refer to their root node. In order to determine all users who are allowed to access a given folder/file, its root node has to be identified. Hence, a user has access to a given file if

- this file is contained in his own root node, or
- this file is contained in a root node that belongs to one of his groups.

Initially, when a new user starts the usage phase of the Bdrive Client, no files are shared with him. Consequently, his own root node is empty. The user may now create objects by moving them to his root node or to specific subfolders shared with a group of other users, provided that these users belong to the same company, are verified guest users of said company or are part of a company which takes part in an intercompany sharing relationship. In particular, every user as well as every group of users is associated to exactly one company or a group of companies that trust each other (identified by its company ID).

As explained above, all files that a given user may access are contained in his own root node or in the root node of one of his groups. The user may change the access rights for exactly these files, simply by moving them to other shared folders. In addition, the user is able to change permissions on owned shared folders by granting additional user access to a share.

Additionally, every user is associated with exactly one list of devices that are registered for him. Please note that the encryption and authentication certificates as defined in Table 2 are device-specific and associated to one user. If a user has access to a given file/folder, it means that file/folder access is possible from every device registered for this user.

3.2.2 Objects

Object	Description
original file, plain file	See Table 1
encrypted file	Cryptogram of the original file - plain file encrypted with AES-256 - plain file authenticated with HMAC-SHA256
Encrypted user-specific MDA	Cryptogram of the user specific meta data (see p.8)
encrypted file chunks	Fragments of the encrypted file according to the Erasure Coding Parameters

Table 4: Objects

Together with the decision to put an original file under control of the Bdrive Client, the user decides whether this file is going to be shared with other users, or whether this file remains accessible to him only. The list of users with access rights to a file are the security attributes to be maintained for all objects. Encrypted files and encrypted file chunks inherit the security attributes of the original file.

3.2.3 Operations

Operation	Description
Encryption	Processes an encryption algorithm to the plaintext data using the encryption key and returns the corresponding ciphertext data. For the TOE this operation includes the generation of a MAC when applied to the above mentioned objects.
Decryption	Processes a decryption algorithm to the ciphertext data using the decryption key and returns the corresponding plaintext data. For the TOE this operation includes the verification of a MAC when applied to the above mentioned objects.
Export	Upload of an object to an external entity
Import	Download of an object from an external entity

Operation	Description
Changing access permissions	The user is able to change permissions on owned shared folders by granting additional user access or remove existing user access to a share.

Table 5: Operations

3.3 Threats

This section describes the threats to be averted by the TOE independently or in collaboration with its IT environment.

The threats to the TOE will be defined in the following manner:

T.Name

The description of the threat.

Taking the preceding considerations into account, the following threats to the TOE are of relevance.

Note: In case that an authorized user is the threat agent in one of the threats listed below, this threat concerns only assets that are not associated to this user.

3.3.1 T.UnauthLocalAccess

An attacker gains access to the plain files or user specific meta data by getting access to local Client PC.

Such attacks compromise the confidentiality and integrity of the assets.

3.3.2 T.UnauthRemoteAccess

An attacker gains access to the plain files or user specific meta data by getting remote access to the TOE by using the external TOE interface.

Such attacks compromise the confidentiality and integrity of the assets.

3.3.3 T.DisclosureKey

An attacker gets access to the locally stored private keys and/or certificates and allows him to decrypt the symmetric file encryption keys and meta data and/or compromise the authenticity of the assets.

3.3.4 T.DisclosurePW

An attacker attempts to disclose the used passwords protecting private keys in order to get access to primary or secondary assets (e.g. Brute Force attack)

3.3.5 T.DisclosureMKey

An attacker attempts to get access to the assets using the administrative masterkey in order to get access to all stored files to decrypt them.

3.3.6 T.Residual

An attacker attempts to gain access to user data or to TSF data from previous sessions with different users by exploiting residual information from the workstation or its public interfaces.

3.4 Organizational Security Policies

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

The organizational security policies (OSP) for the TOE will be defined in the following manner:

OSP.Name **Short title**

Description of the organizational security policy.

3.4.1 OSP.Recovery

The company administrator creates the company masterkey pair and authorizes users and their devices towards the IDP.

Each company using the Bdrive system shall be able to recover the data that have been transferred to the system in case that a user / device key of any folder and files associated to the company gets lost.

3.4.2 OSP.UserApproval

The company administrator assures only authenticated users get access to the Bdrive System.

3.5 Assumptions

Assumptions need to be taken into account in order to ensure a secure operation of the TOE.

The assumptions for the TOE (A) will be defined in the following manner:

A.Name

Description of the assumption.

3.5.1 A.Installation

It is assumed that all software components belonging to the TOE are properly installed. Especially only authentic certificates have to be enrolled.

3.5.2 A.Credentials

It is assumed that measures are taken to ensure that all authorized users protect their credentials in a way that they may not be disclosed to other individuals.

3.5.3 A.Malware

It is assumed that the Bdrive Client workstation is free from untrusted soft- and hardware which may maliciously affect with the operating system, with the other software or with the hardware.

3.5.4 A.Admin

It is assumed that the administrators of all security-relevant systems in the TOE environment are trustworthy and well-trained in order to be aware of security risks and respective measures to protect the installations against such security risks.

3.5.5 A.User

It is assumed that users of a workstation (including other privileged staff) do not actively or negligently compromise the security of the workstation on which the TOE is installed. E.g. the users

- do not leave such a workstation unattended while in operational state,
- do not modify the TOE program or data files,
- do not modify the hard disk in order to compromise the encryption,
- do not copy or transfer private keys used by the TOE outside the workstation,
- keep secrets used for authentication personal,
- do not place malicious software on the workstation.

3.5.6 A.Physical

The workstation on which the TOE is installed shall not fall under temporary and undetected physical control of an attacker.

3.5.7 A.TrustedBackend

It is assumed that the TOE requires the backend infrastructure to be sufficiently protected by physical and logical security measures against potential attackers.

4 Security Objectives

This chapter describes the security objectives for the TOE and the security objectives for the TOE environment.

The security objectives for the TOE (O) and the security objectives for the operational environment (OE) will be defined in the following manner:

O/OE.Name Short title

Description of the objective.

4.1 Security Objectives for the TOE

The following security objectives address the protection provided by the TOE *independently* of the environment as well as the organizational security policies to be met by the TOE independently of the operational environment.

4.1.1 O.Keygen Key generation

The TOE shall generate secure symmetric keys and private / public key pairs for users and company.

4.1.2 O.Access Access to data

Only authorized users shall have access to the contents of plain files in Bdrive, to user specific meta data or to secondary assets that may reveal these contents.

4.1.3 O.UserAuthentication Authentication of BDrive Users

Before encryption of new data for additional devices and users, the TOE ensures the authenticity of receiving devices and users.

4.1.4 O.Integrity Integrity of file contents and file information

The integrity and confidentiality of the user data in form of plain file content and meta data shall be protected.

4.1.5 O.Authentication Authentication of external entities

The TOE shall verify the authenticity of the external entities Bdrive Server and IDP.

4.1.6 O.Residual Protection of residual information

Residual information that occurs temporarily in the scope of the TOE and that may violate the security protection of one or more assets shall be erased when no longer needed.

4.2 Security Objectives for the Operational Environment

The following security objectives for the operational environment of the TOE are defined:

4.2.1 OE.Installation

All software components of the Bdrive system shall be properly installed according the user guidance documentation.

4.2.2 OE.Credentials

Measures shall be taken to ensure that all authorized users protect their credentials in a way that they may not be disclosed to other individuals.

4.2.3 OE.Malware

The Bdrive Client workstation shall be free from untrusted soft- and hardware which may prevent the operating system, the other software or the hardware from its intended behaviour.

4.2.4 OE.Admin

Administrators shall be trustworthy and well-trained in order to be aware of security risks and respective measures to protect the installations against such security risks.

4.2.5 OE.User

Authorized users shall not actively or negligently compromise the security of the workstation on which the TOE is installed.

If the TOE is used by a company, then the OEs OE.Credentials, OE.Malware, OE.Admin, OE.User may be considered as the result of an effective information security management system in place.

4.2.6 OE.Physical

The workstation on which the TOE is installed shall not fall under temporary and undetected physical control of an attacker.

4.2.7 OE.TrustedBackend

The backend infrastructure required by the TOE shall be sufficiently protected against attackers by physical and logical security measures.

4.2.8 OE.MasterKey

The company that uses the Bdrive system shall have measures in place in order to protect the confidentiality of the private part of the Company Masterkey and the integrity of the public part, e.g. a four-eyes-principle for access to this key.

4.3 Security Objective Rationale

Table 6 provides an overview of security objectives coverage (TOE and its environment) also giving an evidence for sufficiency and necessity of the objectives defined. It shows that all threats and OSPs are addressed by the security objectives. It also shows that all assumptions are addressed by at least one security objective for the environment.

A detailed justification required for suitability of the security objectives is given here:

- By O.Keygen, the TOE is required to generate random numbers and keys, respectively, in such a way that it is not possible to reveal or guess any secrets that are used for protection of assets. Hence, O.Keygen counters the threats T.UnauthLocalAccess, T.UnauthRemoteAccess and T.DisclosureKey.

- O.Access requires the TOE to allow access only to authorized users based on their security attributes, according to the specific access rights. In particular, O.Access guarantees that one user may not get access to assets that are not shared with him by another user. Hence, O.Access counters the threat T.UnauthLocalAccess, T.UnauthRemoteAccess, T.DisclosurePW and T.Residual.
- O.UserAuthentication requires the TOE to check and ensure the authenticity of receiving devices and users before encryption of new data for additional devices and users. Hence, this counters threats T.UnauthLocalAccess and T.UnauthRemoteAccess and fulfils OSP.UserApproval.
- O.Integrity ensures that the TOE takes measures such that user data in the scope of the TOE and the Bdrive system may not be manipulated, neither accidentally nor deliberately. Hence, O.Integrity counters T.UnauthLocalAccess and T.UnauthRemoteAccess.
- O.Authentication ensures that the authenticity of external entities involved in the Bdrive system is verified before exposing assets or information on assets at the external interfaces of the TOE. Hence, O.Authentication counters T.UnauthRemoteAccess.
- O.Residual ensures that it is not possible for any user to access assets of a different user by exploiting residual information exposed by the TOE. Therefore, O.Residual counters T.UnauthLocalAccess, T.UnauthRemoteAccess, T.DisclosureKey, T.DisclosurePW and T.Residual.
- OE.Installation requires the person in charge of the initial installation to ensure the proper installation of the TOE. Hence, A.Installation is fulfilled by OE.Installation.
- The collection of OE.Credentials, OE.Malware, OE.Admin, OE.User ensures that the assumptions A.Credentials, A.Malware, A.Admin, and A.User and policy OSP.UserApproval are fulfilled. Furthermore they are supportive in mitigating T.UnauthLocalAccess, T.UnauthRemoteAccess, T.DisclosureKey and T.DisclosurePW by preventing unauthorized access and the disclosure of keys and passwords.
- OE.Physical prevents attackers from having physical access to the workstation where the TOE is running on. Therefore, T.UnauthLocalAccess,, T.DisclosureKey, T.DisclosurePW, T.DisclosureMKey, T.Residual are countered by the objective for the environment and A.Physical is fulfilled.
- OE.TrustedBackend requires the sufficient protection of the backend infrastructure that is required by the TOE. Hence, A.TrustedBackend is fulfilled by OE.TrustedBackend.
- OE.MasterKey ensures that there is no possibility of malicious use of the Company Masterkey. This key is reserved for cases when a user's key gets lost. Therefore, this objective counters the threats T.UnauthLocalAccess, T.UnauthRemoteAccess, T.DisclosureMKey and fulfils OSP.Recovery.

	O.Keygen	O.Access	O.UserAuthentication	O.Integrity	O.Authentication	O.Residual	OE.Installation	OE.Credentials	OE.Malware	OE.Admin	OE.User	OE.Physical	OE.TrustedBackend	OE.MasterKey
T.UnauthLocalAccess	x	x	x	x		x			x	x	x	x		x
T.UnauthRemoteAccess	x	x	x	x	x	x			x	x	x			x
T.DisclosureKey	x					x			x	x	x	x		
T.DisclosurePW		x				x			x	x	x	x		
T.DisclosureMKey												x		x
T.Residual		x				x						x		
OSP.Recovery														x
OSP.UserApproval			x						x	x	x			
A.Installation							x							
A.Credentials								x						
A.Malware									x					
A.Admin										x				
A.User											x			
A.Physical												x		
A.TrustedBackend													x	

Table 6: Security Objectives Rationale

5 Extended Components Definition

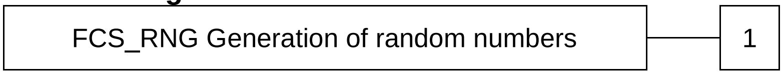
The following family defined in [AIS31/20] is not described in [CC_P2]. It is used as extended component in this Security Target.

5.1 FCS_RNG Generation of random numbers

Family Behaviour

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

Component levelling:



FCS_RNG.1 Generation of random numbers requires that the random number generator implements defined security capabilities and that the random numbers meet a defined quality metric.

Management: FCS_RNG.1

There are no management activities foreseen.

Audit: FCS_RNG.1

There are no actions defined to be auditable.

FCS_RNG.1 Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a [selection: physical, non-physical true, deterministic, hybrid physical, hybrid deterministic] random number generator that implements: [assignment: list of security capabilities].

FCS_RNG.1.2 The TSF shall provide random numbers that meet [assignment: a defined quality metric].

6 Security Requirements

This chapter of the Security Target defines the detailed security requirements that shall be satisfied by the TOE. The statement of TOE security requirements shall define the functional and assurance security requirements that the TOE needs to satisfy in order to meet the security objectives for the TOE.

The CC allows several operations to be performed on security requirements (on the component level); refinement, selection, assignment, and iteration are defined in paragraph 8.1 of Part 1 of the CC [CC1]. These operations are used in this ST.

The **refinement** operation is used to add details to a requirement, and thus further restrict a requirement. Refinements of security requirements are denoted in such a way that added words are in **bold text** and changed words are ~~crossed out~~.

The **selection** operation is used to select one or more options provided by Part 2 resp. Part 3 of the CC [CC2], [CC3] in stating a requirement. Selections having been made are denoted as <u>underlined text.</u>

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made are denoted by showing as *italicized text* .

The **iteration** operation is used when a component is repeated with varying operations. An iteration is denoted by showing a slash “/” and the iteration indicator after the component identifier. In order to trace elements belonging to a component, the same slash “/” with iteration indicator is used behind the elements of a component.

6.1 Security functional policy (SFP)

<u>**Bdrive Access policy**</u>

- subjects: users
- security attributes of users: user IDs, private authentication and encryption keys, authentication and encryption certificates
- objects: encrypted files, encrypted file chunks, encrypted meta data
- security attributes of encrypted files: list of user IDs
- security attributes of encrypted file chunks: Forward Error parameters
- operation: decryption
  Rule: A user can only decrypt an encrypted file if and only if one of his deviceIDs is contained in the list of deviceIDs associated with this encrypted file.
- operation: encryption
  Rule: A user only encrypts plain file with authenticated certificates of devices contained in the list of deviceIDs associated with enclosing root node.

Note:

1. *As stated in chapter 1.4.2, for the sake of availability, the symmetric keys in the general MDA are additionally encrypted by the TOE with the so-called Company Masterkey, which is the public key of an asymmetric key pair generated by a BDrive-Admin of the customer company. In case that a device and therefore the private key of a user of a particular company gets lost, this Company Masterkey can be used in order to recover the symmetric keys that are needed for decryption and integrity check of the encrypted file chunks.*
   Please note that the decryption with the help of this Company Masterkey is not part of the TSF.
2. *Please refer to Table 4: Objects and the explanation of the security attribute "list of user IDs" there.*

6.2 Security Functional Requirements

6.2.1 Class FCS: Cryptographic support

6.2.1.1 Cryptographic key generation (FCS_CKM.1)

Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/SymKeyDerivation

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Key Derivation through Extraction-then-Expansion using HMAC-SHA-256 (cf. FCS_COP.1/SHA) for extraction and expansion* and specified cryptographic key sizes *256 bits* that meet the following: *Recommendations for key derivation as specified in NIST Special Publication 800-56C.*

FCS_CKM.1.1/AsymKeyDerivation

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Key Generation Algorithm for RSA (without CRT)* and specified cryptographic key sizes *4096 bits* that meet the following: *section 3.1 in [PKCS1 V2.2].*

Note: For generation of RSA keys the function generate_rsa_prime of Botan 2.4 is used.

FCS_CKM.1.1/TLS

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *RSA schemes* and specified cryptographic key sizes of *2048-bit or greater* that meet the following: *FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3;*

6.2.1.2 Cryptographic key destruction (FCS_CKM.4)

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *no defined standard*.

6.2.1.3 Cryptographic Operation (FCS_COP.1)

Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/AES

The TSF shall perform *symmetric encryption and decryption of user data* in accordance with a specified cryptographic algorithm *AES-256 in CTR mode of operation and block size 256 bits with zero padding* and cryptographic key sizes *256 bits* that meet the following: *AES standard as specified in FIPS-197 and CTR mode as specified in NIST Special Publication SP800-38a.*

FCS_COP.1.1/TLS.HASH

The TSF shall perform [*cryptographic hashing*] in accordance with a specified cryptographic algorithm *SHA-256, SHA-384, SHA-512* and cryptographic key sizes *256-bit, 384-bit, 512-bit* that meet the following: *FIPS Pub 180-4*.

FCS_COP.1.1/HMAC

The TSF shall perform *file authentication* in accordance with a specified cryptographic algorithm *HMAC using SHA-256* and cryptographic key sizes *256 bits* that meet the following: *FIPS180-4 for SHA, RFC2104 for HMAC.*

FCS_COP.1.1/RSA

The TSF shall perform *asymmetric encryption and decryption of TSF data* in accordance with a specified cryptographic algorithm *RSA* and cryptographic key sizes *4096 bits* that meet the following: *RSAES-OAEP with SHA-256 and MGF.1 as in PKCS #1 v2.2.*

FCS_COP.1.1/Keystore

The TSF shall perform *user private key encryption and decryption* in accordance with a specified cryptographic algorithm *PKCS #5 using PBKDF2 as key derivation function, PBES 2 as encryption scheme and PBMAC1 as message authentication scheme* and cryptographic key sizes *256 bits* that meet the following: *Password-based cryptography as specified in PKCS #5 v2.1.*

FCS_COP.1.1/SHA

The TSF shall perform *hash calculation of TSF data* in accordance with a specified cryptographic algorithm *SHA-256* and cryptographic key sizes *none* that meet the following: *SHA-2 standard as specified in FIPS-180-4.*

FCS_COP.1.1/TLS

The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm: *see Table 7* and cryptographic key sizes *see Table 7* that meet the following*: see Table 7.*

Operation/Purpose	**Algorithms / Cipher Suite**	**Standard**
Authentication	See Table 8	See Table 8
Key Agreement	Diffie-Hellman Group 14	RFC3526
Key Agreement	DH exponent minimum length of 384 bits	--
Confidentiality	Forward secrecy	--

Table 7: Algorithms / Cyphersuite

6.2.1.4 Generation of random numbers (FCS_RNG.1), cf. [AIS31/20]

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 (cf. [AIS31/20], sec. 4.8.1)
The TSF shall provide a *deterministic* random number generator that implements:

> *(DRG.3.1) If initialized with a random seed using internal entropy sources based on temporal and positional properties of user actions, the internal state of the RNG shall have a minimum of 120 bits of minentropy.*
>
> *(DRG.3.2) The RNG provides forward secrecy.*
>
> *(DRG.3.3) The RNG provides backward secrecy even if the current internal state is known.*

FCS_RNG.1.2 (cf. [AIS31/20], sec. 4.8.1)
The TSF shall provide random numbers that meet:

> *(DRG.3.4) The RNG, initialized with a random seed containing at least 180 bits of entropy before the first use and reseeded after 512 invocations, generates output for which 2^{19} strings of bit length 128 are mutually different with probability $1\text{-}2^{-12}$.*
>
> *(DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A.*

Note: The TOE implements an HMAC_DRBG as defined in SP800-90A Chapter 10.1.2 using hash function SHA-256. The initial seed is based on random information like points in time between user actions and the position of the mouse cursor at user actions or the system time and date. For reseeding similar temporal and positional properties of user actions are used. For more information about this, please contact the developer.

6.2.2 Class FDP: User Data Protection

6.2.2.1 Subset access control (FDP_ACC.1)

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1
The TSF shall enforce the *Bdrive Access policy* on *the subjects, objects and operations as listed in sec. 6.1.*

6.2.2.2 Security Attribute Based Access Control (FDP_ACF.1)

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 *Bdrive Access policy* to objects based on the following: *subjects, objects and their security attributes as listed in sec. 6.1.*

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *The rule stated in the Bdrive Access policy in sec. 6.1.*

FDP_ACF.1.3

The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no additional rules.*

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *A user may not decrypt encrypted files if his authentication certificate is revoked or invalid because the TOE is unable to access his own private encryption key from the Bdrive Server without valid authentication*.

6.2.2.3 Basic Data Authentication (FDP_DAU.1)

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

FDP_DAU.1.1

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of *encrypted files*.

FDP_DAU.1.2

The TSF shall provide *users* with the ability to verify evidence of the validity of the indicated information.

Note: This SFR refers to the authentication by means of HMAC of encrypted files.

6.2.2.4 Export of user data with security attributes (FDP_ETC.2)

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]

FDP_ETC.2.1

The TSF shall enforce the *Bdrive Access policy* when exporting user data, controlled under the SFP(s), outside of the TOE.

FDP_ETC.2.2

The TSF shall export the user data with the user data's associated security attributes.

FDP_ETC.2.3

The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.

FDP_ETC.2.4

The TSF shall enforce the following rules when user data is exported from the TOE:
The list of user IDs with access rights to the file contained in the general MDA and transferred to the Bdrive Server has to coincide with the list of users as chosen by the user sharing the file.

Note: This SFR refers to the transfer of the list of user IDs with access rights for a given file from the TOE to the Bdrive server as part of the general meta data. It has to be ensured that this list of user IDs stays associated to the given file and that it is not modified during transfer.

6.2.2.5 Subset residual information protection (FDP_RIP.1)

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

FDP_RIP.1.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the <u>deallocation of the resource from</u> the following objects: *password protecting the private keys.*

6.2.2.6 Basic data exchange confidentiality (FDP_UCT.1)

Hierarchical to:	No other components.
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]

FDP_UCT.1.1

The TSF shall enforce the *Bdrive Access policy* to <u>transmit and receive</u> user data in a manner protected from unauthorised disclosure.

6.2.2.7 Source data exchange recovery (FDP_UIT.2)

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FDP_UIT.1 Data exchange integrity, or FTP_ITC.1 Inter-TSF trusted channel]

FDP_UIT.2.1

The TSF shall enforce the *Bdrive Access policy* to be able to recover from *deletion of less than n-k arbitrary*[3] *encrypted file chunks* with the help of the source trusted IT product.

6.2.3 Class FIA: Identification and Authentication

6.2.3.1 User attribute definition (FIA_ATD.1)

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

FIA_ATD.1.1

[3] See the explanation of the parameters k and n in sec. 1.3.1.

The TSF shall maintain the following list of security attributes belonging to individual users:

- *User ID, deviceID*
- *private authentication and encryption key to the user ID and deviceID.*

6.2.3.2 Verification of secrets (FIA_SOS.1)

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

FIA_SOS.1.1

The TSF shall provide a mechanism to verify that secrets meet *the following password definition rule: Minimum 8 characters with upper and lower case, at least one special character and at least one number*.

6.2.3.3 User identification before any action (FIA_UID.2)

Hierarchical to:	FIA_UID.1 Timing of identification
Dependencies:	No dependencies.

FIA_UID.2.1

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

6.2.3.4 Timing of authentication (FIA_UAU.1)

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

FIA_UAU.1.1

The TSF shall allow *the unlocking procedure of the private part of the authentication key* on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2

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

6.2.4 Class FMT: Security Management

6.2.4.1 Management of security attributes (FMT_MSA.1)

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 *Bdrive Access policy* to restrict the ability to modify the security attribute~~s~~ *list of user IDs with access rights to a file* to *authorized users with access right to this file*.

Note: The TOE offers the possibility to manage the shared folders (including the invitation of new users to a folder). The technical implementation of the access control lists is part of the Bdrive Server backend.

6.2.4.2 Static attribute initialisation (FMT_MSA.3)

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 *Bdrive Access policy* to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The TSF shall allow the *Bdrive server* to specify alternative initial values to override the default values when an object or information is created.

6.2.4.3 Specification of management functions (FMT_SMF.1)

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

FMT_SMF.1.1

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

- *setting the security attribute (=list of users with access rights) of encrypted files according to the user's choice.*

6.2.4.4 Security Roles (FMT_SMR.1)

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

FMT_SMR.1.1

The TSF shall maintain the roles *authenticated user and unknown user*.

Note: The role "Company Administrator" is not known to the TSF, see chapter 3.2.1.

FMT_SMR.1.2

The TSF shall be able to associate users with roles.

6.2.5 Class FTA: TOE Access

6.2.5.1 User-initiated termination (FTA_SSL.4)

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

FTA_SSL.4.1

The TSF shall allow user-initiated termination of the user's own interactive session.

6.2.6 Class FTP: Trusted path/channels

6.2.6.1 Inter-TSF trusted channel (FTP_ITC.1)

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1

The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure **using the cipher suites listed in Table 8:**

a) **Cryptographically-protected communication channel using the external interface to the Bdrive backend system with a combination of the following cipher suites defined there:**
   1 **Symmetric cipher defined in FCS_COP.1/TLS**
   2 **Keyed hash algorithms defined in FCS_COP.1/TLS.HASH as defined in [RFC5246].**

b) **Authenticated communication channel using TLS as defined in [RFC5246] for server authentication.**

Cipher Suite	IANA no	Specified in
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	0xC0,0x23	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	0xC0,0x24	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	0xC0,0x2B	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	0xC0,0x2C	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	0xC0,0xAC	RFC7251
TLS_ECDHE_ECDSA_WITH_AES_256_CCM	0xC0,0xAD	RFC7251
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	0xC0,0x27	RFC5289
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	0xC0,0x28	RFC5289
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	0xC0,0x2F	RFC5289
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	0xC0,0x30	RFC5289
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	0x00,0x40	RFC5246
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256	0x00,0x6A	RFC5246
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	0x00,0xA2	RFC5288
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	0x00,0xA3	RFC5288
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	0x00,0x67	RFC5246
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	0x00,0x6B	RFC5246
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	0x00,0x9E	RFC5288
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	0x00,0x9F	RFC5288
TLS_DHE_RSA_WITH_AES_128_CCM	0xC0,0x9E	RFC6655
TLS_DHE_RSA_WITH_AES_256_CCM	0xC0,0x9F	RFC6655

Table 8: TLS cipher suites

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 *all security functions specified in the ST that interact with remote trusted IT systems and no other conditions or functions*.

6.3 Security Assurance Requirements

The SARs are taken from the package EAL2. In more detail, these security assurance requirements are:

Security Target evaluation (Class ASE)

- Conformance claims ASE_CCL.1
- Extended components definition ASE_ECD.1
- ST introduction ASE_INT.1
- Security objectives ASE_OBJ.2
- Security requirements ASE_REQ.2
- Security problem definition ASE_SPD.1
- TOE summary specification ASE_TSS.1

Development (Class ADV)

- Security architecture ADV_ARC.1
- Functional specification ADV_FSP.2
- TOE design ADV_TDS.1

Guidance documents (Class AGD)

- Operational user guidance AGD_OPE.1
- Preparative procedures AGD_PRE.1

Life cycle support (Class ALC)

- CM capabilities ALC_CMC.2
- CM scope ALC_CMS.2
- Delivery ALC_DEL.1

Tests activities (Class ATE)

- Coverage ATE_COV.1
- Functional tests ATE_FUN.1
- Independent testing ATE_IND.2

Vulnerability assessment (Class AVA)

- Vulnerability analysis AVA_VAN.2

6.4 Security Requirements Rationale

6.4.1 Security Functional Requirements Rationale

This chapter proves that the quantity of security requirements (TOE) is suited to fulfill the security objectives described in sec. 4.1 and that it can be traced back to the security objectives. At least one security objective exists for each security requirement.

The following table provides an overview on how the TOE security functional requirements cover the TOE security objectives.

	O.Keygen Key generation	O.Access Access to data	O.UserAuthentication	O.Integrity Integrity of file contents and file information	O.Authentication Authentication of external entities	O.Residual Protection of residual information
Cryptographic key generation (FCS_CKM.1)	x					
Cryptographic key destruction (FCS_CKM.4)	x					x
Cryptographic Operation (FCS_COP.1)		x		x		
Generation of random numbers (FCS_RNG.1), cf. [AIS31/20]	x					
Subset access control (FDP_ACC.1)		x				
Security Attribute Based Access Control (FDP_ACF.1)		x				
Basic Data Authentication (FDP_DAU.1)				x	x	
Export of user data with security attributes (FDP_ETC.2)		x			x	
Subset residual information protection (FDP_RIP.1)		x				x
Basic data exchange confidentiality (FDP_UCT.1)		x				
Source data exchange recovery (FDP_UIT.2)				x		
User attribute definition (FIA_ATD.1)		x	x	x		
Verification of secrets (FIA_SOS.1)		x	x	x		
User identification before any action (FIA_UID.2)		x	x	x		

	O.Keygen Key generation	O.Access Access to data	O.UserAuthentication	O.Integrity Integrity of file contents and file information	O.Authentication Authentication of external entities	O.Residual Protection of residual information
Timing of authentication (FIA_UAU.1)		x	x	x		
Management of security attributes (FMT_MSA.1)		x	x			
Static attribute initialisation (FMT_MSA.3)		x	x			
Specification of management functions (FMT_SMF.1)		x	x	x		
Security Roles (FMT_SMR.1)		x	x	x		
User-initiated termination (FTA_SSL.4)		x				x
Inter-TSF trusted channel (FTP_ITC.1)		x	x	x		

Table 9: SFRs and security objectives for the TOE

The security objective O.Keygen requires the generation of cryptographic keys. This is addressed by FCS_CKM.1 for key generation, by FCS_RNG.1 for random number generation for key generation and by FCS_CKM.4 for destruction of keys. Therefore, these SFRs are suitable to meet O.Keygen.

The security objective O.Access requires that only authorized users have access to plain files in Bdrive, to user specific meta data or to secondary assets that may reveal these contents. This access policy is defined in FDP_ACC.1 and FDP_ACF.1. Export of user data is addressed by FDP_ETC.2. Residual information protection is addressed by FDP_RIP.1. FDP_UCT.1 enforces the access policy on transmitted and received data. FTP_ITC.1 ensures that transmitted and received data are encrypted via using a trusted channel. Identification and authentication of users is defined in FIA_ATD.1, FIA_SOS.1, FIA_UID.2 and FIA_UAU.1. Security management of the access policy is defined in FMT_MSA.1, FMT_MSA.3, FMT_SMF.1 and FMT_SMR.1. FTA_SSL.1 defines that the user can terminate his own session. Encryption and decryption of data is addressed by FCS_COP.1. Therefore, these SFRs are suitable to meet O.Access.

The security objective O.UserAuthentication requires the TOE to check and ensure the authenticity of receiving devices and users before encryption of new data for additional devices and users. This is defined in the SFRs for identification and authentication FIA_ATD.1, FIA_SOS.1, FIA_UID.2 and FIA_UAU.1. Security management of the access policy is defined in FMT_MSA.1, FMT_MSA.3, FMT_SMF.1 and FMT_SMR.1. Furthermore FTP_ITC.1 ensures the secure transmission of data via the trusted channel.

The security objective O.Integrity requires that integrity and authenticity of the user data in form of plain file content and meta data is protected. The cryptographic mechanisms for integrity and authenticity protection are defined in FCS_COP.1. The data authentication is defined in FDP_DAU.1. FDP_UIT.2 addresses recovery of data. Identification and authentication of users is defined in FIA_ATD.1, FIA_SOS.1, FIA_UID.2 and FIA_UAU.1. Security management is specified in FMT_SMF.1 with regard to the management functions and in FMT_SMR.1 with regard to the user roles. Furthermore FTP_ITC.1 ensures the secure transmission of data via the trusted channel. Therefore, these SFRs are suitable to meet O.Integrity.

The security objective O.Authentication requires that the authenticity of the external entities Bdrive Server and IDP is verified. The data authentication is defined in FDP_DAU.1. Export of user data is addressed by FDP_ETC.2. Therefore, these SFRs are suitable to meet O.Authentication.

The security objective O.Residual requires that residual information that occurs temporarily in the scope of the TOE and that may violate the security protection of one or more assets is erased when no longer needed. Residual information protection of the password protecting the private keys is addressed by FDP_RIP.1. Destruction of cryptographic keys is covered by FCS_CKM.4. FTA_SSL.1 defines that the user can terminate his own session. Therefore, these SFRs are suitable to meet O.Authentication.

6.4.2 Rationale for SFR dependencies

Component	Dependencies	Dependency fulfilled by
Cryptographic key generation Sym. FCS_CKM.1.1/ SymKeyDerivation	[FCS_CKM.2, or FCS_COP.1]	Cryptographic Operation (FCS_COP.1.1/AES)
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic key generation Asym. FCS_CKM.1.1/ AsymKeyDerivation	[FCS_CKM.2, or FCS_COP.1]	Cryptographic Operation (FCS_COP.1.1/RSA)
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic key generation TLS FCS_CKM.1.1/TLS	[FCS_CKM.2, or FCS_COP.1]	Cryptographic Operation (FCS_COP.1.1/TLS)
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic key destruction (FCS_CKM.4)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	Cryptographic key generation (FCS_CKM.1)
Cryptographic Operation AES (FCS_COP.1.1/AES)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	FCS_CKM.1.1/SymKeyDerivation
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic Operation TLS-HASH (FCS_COP.1.1/TLS.HASH)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	As this is only a hashing function, no cryptographic keys are required.
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic Operation HMAC (FCS_COP.1.1/HMAC)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	FCS_CKM.1.1/SymKeyDerivation
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic Operation RSA (FCS_COP.1.1/RSA)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	FCS_CKM.1.1/AsymKeyDerivation
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic Operation Keystore (FCS_COP.1.1/Keystore)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	The password for *PBKDF2* is enforced by FIA_SOS.1.
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	As this is only a hashing function, no cryptographic keys are required.

Component	Dependencies	Dependency fulfilled by
Cryptographic Operation Hash (FCS_COP.1.1/SHA)	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Cryptographic Operation TLS (FCS_COP.1.1/TLS)	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	FCS_CKM.1.1/TLS
	FCS_CKM.4	Cryptographic key destruction (FCS_CKM.4)
Generation of random numbers (FCS_RNG.1), cf. [AIS31/20]	No dependencies.	-
Subset access control (FDP_ACC.1)	FDP_ACF.1	Security Attribute Based Access Control (FDP_ACF.1)
Security Attribute Based Access Control (FDP_ACF.1)	FDP_ACC.1	Subset access control (FDP_ACC.1)
	FMT_MSA.3	Static attribute initialisation (FMT_MSA.3)
Basic Data Authentication (FDP_DAU.1)	No dependencies.	-
Export of user data with security attributes (FDP_ETC.2)	[FDP_ACC.1, or FDP_IFC.1]	Subset access control (FDP_ACC.1)
Subset residual information protection (FDP_RIP.1)	No dependencies.	-
Basic data exchange confidentiality (FDP_UCT.1)	[FTP_ITC.1, or FTP_TRP.1]	Inter-TSF trusted channel (FTP_ITC.1)
	[FDP_ACC.1, or FDP_IFC.1]	Subset access control (FDP_ACC.1)
Source data exchange recovery (FDP_UIT.2)	[FDP_ACC.1, or FDP_IFC.1]	Subset access control (FDP_ACC.1)
	[FTP_UIT.1, or FTP_ITC.1]	Inter-TSF trusted channel (FTP_ITC.1)
User attribute definition (FIA_ATD.1)	No dependencies.	-
Verification of secrets (FIA_SOS.1)	No dependencies.	-
User identification before any action (FIA_UID.2)	No dependencies.	-

Component	Dependencies	Dependency fulfilled by
Timing of authentication (FIA_UAU.1)	FIA_UID.1	User identification before any action (FIA_UID.2)
Management of security attributes (FMT_MSA.1)	[FDP_ACC.1, or FDP_IFC.1]	Subset access control (FDP_ACC.1)
	FMT_SMR.1	Security Roles (FMT_SMR.1)
	FMT_SMF.1	Specification of management functions (FMT_SMF.1)
Static attribute initialisation (FMT_MSA.3)	FMT_MSA.1	Management of security attributes (FMT_MSA.1)
	FMT_SMR.1	Security Roles (FMT_SMR.1)
Specification of management functions (FMT_SMF.1)	No dependencies.	-
Security Roles (FMT_SMR.1)	FIA_UID.1	User identification before any action (FIA_UID.2)
User-initiated termination (FTA_SSL.4)	No dependencies.	-
Inter-TSF trusted channel (FTP_ITC.1)	No dependencies.	-

Table 10: Rationale for SFR dependencies

6.4.3 Security Assurance Requirements Rationale

The assurance package for this ST is chosen to be EAL2. EAL2 is usually applicable in the situations, where users require moderate level of independently assured security, without additional effort from the developer side, other than is consistent with good commercial practice. EAL2 is the lowest level, which includes the vulnerability analysis with the penetration testing, which gives the assurance that the TOE is resistant to attackers.

As such, EAL2 is appropriate for the desktop client of a system solution for storing and sharing for files of all types.

7 TOE Summary Specification

This chapter describes how the TOE addresses the SFRs defined in sec. 6.1.

7.1 Login

When starting the Desktop Client, the unidentified user has to authenticate via the following authentication method on every login:

- authentication with authentication certificate.

This realizes User identification before any action (FIA_UID.2). Only after successful verification of the user's credentials (certificate + protection password) at the IDP, the user is allowed to perform any further action, which realizes Timing of authentication (FIA_UAU.1). In particular, the TOE maintains the user ID as well as the private part of the authentication key for this login procedure which is unlocked by entering the correct device password (User attribute definition (FIA_ATD.1) and Cryptographic Operation (FCS_COP.1), Iteration FCS_COP.1/Keystore).

More precisely, upon successful verification of the user's credentials, the IDP sends a cryptographic token to the Client (see OpenID Connect). The user credentials need to follow the password policy which is technically enforced (FIA_SOS.1). Together with this token, the Client sends a login request to the Bdrive Server. The Bdrive Server validates this token at IDP, the IDP marks this particular token as "used", such that it is not possible to start a second login request with the same token. After successful validation of the token at IDP, the Bdrive Server generates a signed token. The TOE stores the signed token for future requests, on each request the Bdrive server checks the token signature. Then, the user is changed from "unknown user" to authenticated user, which realizes Security Roles (FMT_SMR.1).

7.2 Logout

The user decides upon the termination of the session (User-initiated termination (FTA_SSL.4)). The private parts of the authentication and encryption keys need to be protected upon session termination (Subset residual information protection (FDP_RIP.1) and Cryptographic Operation (FCS_COP.1), Iteration FCS_COP.1/Keystore).

7.3 Management of access rights to files and folders

Every group and every user owns a unique virtual folder called *root node*. A user may be a member of several groups. All subfolders and their files refer to their root node. In order to determine all users who are allowed to access a given folder/file, its root node has to be identified. Hence, a user has access to a given file if

- this file is contained in his own root node, or
- this file is contained in a root node that belongs to one of his groups.

Initially, when a new user starts the usage phase of the Bdrive Client, no files are shared with him. Consequently, his own root node is empty. The user may now create objects by moving them to his root node or to specific subfolders shared with a group of other users. This realizes Static attribute initialisation (FMT_MSA.3).

As explained above, all files that a given user may access are contained in his own root node or in the root node of one of his groups. The user may change the access rights for exactly these files, simply by moving them to other shared folders. This realizes Management of security attributes (FMT_MSA.1) and Specification of Management Functions (FMT_SMF.1).

Users can create new groups by transforming a directory in their private root node into a root node of a new group. Furthermore, every member of a group can invite or remove users.

7.4 Generation of user-specific meta data

The user-specific meta data as detailed in sec. 1.4.2 are generated when a file is put under control of the TOE. The user-specific meta-data contains a checksum (cf. FCS_COP.1/SHA) of the plaintext file. These user-specific meta data are encrypted, authenticated and then transferred to the Bdrive server. The plain file gets encrypted, authenticated, fragmented and transferred to the Cloud storage servers (cf. FDP_UIT.2) . This transfer to the Cloud Storage realizes FDP_UCT.1.

7.5 Generation of general meta data

The general meta data as detailed in sec. 1.4.2 are generated when a file is decided to be transferred to the Bdrive system (user's choice). The list of users who have access rights to the particular file is determined via the corresponding root node, cf. sec. Management of access rights to files and folders. This list makes part of these general meta data, which are transmitted to the Bdrive Server together with the storage locations of this file (Export of user data with security attributes (FDP_ETC.2)).

7.6 Key generation

When a file is decided to be transferred to the Bdrive system (user's choice), two symmetric keys (in the sequel called K_enc and K_auth) are generated by the TOE on the basis of a seed (Generation of random numbers (FCS_RNG.1), cf. [AIS31/20], Cryptographic key generation (FCS_CKM.1) Iteration FCS_CKM.1/SymKeyDerivation).

7.7 File encryption

Files transferred to the Bdrive system are encrypted by AES-256 in CTR mode using the K_enc as mentioned in Sec. 7.6. Big files are split into blocks with a maximum size of 20 MiB which are encrypted blockwise. This realizes Cryptographic Operation (FCS_COP.1), Iteration FCS_COP.1/AES.

7.8 File authentication

The encrypted object as described in sec. 7.7 is authenticated by means of HMAC using the key K_auth, which realizes (FCS_COP.1), Iteration FCS_COP.1/HMAC. In case of big files, each of the blocks (cf. sec. 7.7) is authenticated separately -- the byte offset of each block is appended to the data prior to authentication. Each block gets fragmented (cf. 7.9) and uploaded to the cloud storage servers. The message authentication code (MAC) is stored on the Bdrive Server along with the general meta data (cf. 7.5). This process contributes to Basic Data Authentication (FDP_DAU.1), Basic data exchange confidentiality (FDP_UCT.1) and Source data exchange recovery (FDP_UIT.2.1)..

7.9 File fragmentation

After blockwise data encryption (cf. 7.7) and authentication (cf 7.8), the encrypted block is fragmented using a Reed-Solomon-Cauchy scheme into a number of data chunks and parity chunks. Those are then uploaded to independent cloud storage servers facilitating Source data exchange recovery (FDP_UIT.2.1). The exact number of data and parity chunks is added to the general meta data (cf. 7.5)

7.10 Key encryption & decryption

The keys K_enc and K_auth are asymmetrically encrypted using the device-specific encryption certificates of all authorized users' devices. The TOE determines these certificates from the access list associated with the root node that will contain the previously encrypted file. All encryption certificates are validated by the TOE before usage.

This realizes Cryptographic Operation (FCS_COP.1), Iteration FCS_COP.1/RSA and contrbutes to Subset access control (FDP_ACC.1) and Security Attribute Based Access Control (FDP_ACF.1). Furthermore it is supported by Cryptographic Key Generation (FCS_CKM.1/AsymKeyDerivation).

Afterwards, the symmetric keys are overwritten (Cryptographic key destruction (FCS_CKM.4)).

7.11 Secure channels to other trusted IT products

The TOE connects to the IDP via TLS v1.2. In any case a mutually authenticated TLS handshake takes place. Hence, the channel is established with authentication of both end points. The secure channel to the Bdrive Server is inherited from the secure channel between TOE and IDP.

This realizes Inter-TSF trusted channel (FTP_ITC.1) and is supported by the Cryptographic Key Generation (FCS_CKM.1/TLS) and Cryptographic Operations (FCS_COP.1/TLS and FCS_COP.1/TLS.HASH).

7.12 Certificate Validation

The TOE performs a certificate path validation of authorized encryption certificates before usage for key encryption (cf. 7.9). The path validation traces the certificates' issuer up to a self-signed root CA that is known a priori and shipped with the TOE. Additionally the revocation status of all encryption certificates is checked via OCSP (see identification/authentication via certificates in FIA_UAU.1 and FIA_UID.2).

Appendix

Acronyms

Term	Definition
BSI	Bundesamt für Sicherheit in der Informationstechnik
Botan	Botan is a C++ cryptography library released under the permissive Simplified BSD license and offers a range of tools such as TLS protocol, password hashing, and post quantum crypto schemes. The TOE uses the release 2.12.1.
CC	Common Criteria for IT Security Evaluation
CEM	Common Methodology for Information Technology Security Evaluation
EAL	Evaluation Assurance Level
ECC	Elliptic Curve Cryptography
IDP	Identity Provider
OCSP	Online Certificate Status Protocol
OpenID Connect	OpenID Connect is a simple identity layer on top of the OAuth 2.0 protocol. It allows Clients to verify the identity of the End-User based on the authentication performed by an Authorization Server, as well as to obtain basic profile information about the End-User in an interoperable and REST-like manner.
PKI	Zertifizierungsinfrastruktur / Public Key Infrastructure
SAR	Security Assurance Requirement
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
ST	Security Target
TOE	Target Of Evaluation
TR	Technische Richtlinie
TSF	TOE Security Functionality
MDA	meta data array
SCM	Certificate Service Manager
ACL	Access Control List
FEAK	file encryption and authentication keys, see p. 15.
CSP	Cloud Storage Provider

Bibliography

CC documents	
[CC_P1]	Common Criteria, Part 1: Introduction and General Model; Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
[CC_P2]	Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
[CC_P3]	Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003
[CEM]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004
[AIS31/20]	Bundesamt für Sicherheit in der Informationstechnik, Anwendungshinweise und Interpretationen zum Schema (AIS), AIS 31, A proposal for Functionality classes for random number generators Version 2.0 vom 18.09.2011, Bundesamt für Sicherheit in der Informationstechnik (BSI)
Protection Profiles and Technical Guidelines	
[TR-02102-1]	Technische Richtlinie, Kryptographische Verfahren: Empfehlungen und Schlüssellängen, 2017-01, Stand: 08. Februar 2017
[TR-03111]	Technical Guideline TR-03111: Elliptic Curve Cryptography, Version 2.0, Bundesamt für Sicherheit in der Informationstechnik (BSI), 2012-06-28
[TR-03119]	Technical Guideline BSI TR-03119: Requirements for Smart Card Readers Supporting eID and eSign Based on Extended Access Control, Bundesamt für Sicherheit in der Informationstechnik (BSI), Version 1.3, 2013-03-22
[TR-03224-1]	Technical Guideline TR-03124-1: eID-Client – Part 1: Specifications, Bundesamt für Sicherheit in der Informationstechnik (BSI), Version 1.2, 2015-02-24
[PP-0026]	Protection Profile "Machine Readable Travel Document with "ICAO Application", Extended Access Control, BSI-PP-0026, Version 1.2, 19.11.2007