USB Portable Storage Device: Security Problem Definition Summary

Introduction

The USB Portable Storage Device (hereafter referred to as “the device” or “the TOE”) is a portable storage device that provides a USB interface for connecting to a host computer. The device employs cryptographic means to provide the necessary protection of user data, the strength of which lies in the quality of the cryptographic algorithms, the mode used, the key sizes as well as the entropy of the authorisation factor (e.g., password, passphrase) and cryptographic keys. The device encrypts the user data as it is stored on the device, and decrypts the user data as it leaves the device. All cryptographic functions (encryption/decryption of user data, hashing, random number generation, etc.) are implemented on the device itself. This precludes, for example, encryption/decryption being carried out in a driver on the host computer.

In addition to protecting the user data, the device is also responsible for ensuring the system data (e.g., software, firmware) cannot be modified by untrusted entities through the logical interface.

The device can also provide an optional capability – updating the devices system files (e.g. patches).

The device is dedicated to store user data, which may or may not include removable memory media in the device. This cPP is not suitable for use with more general USB devices that provide access to other forms of storage such as CDs, DVDs or magnetic media, nor for more complex devices that may include memory media (such as smartphone, cameras or media players).

The device, its boundaries, components and connections are depicted in Figure 1.

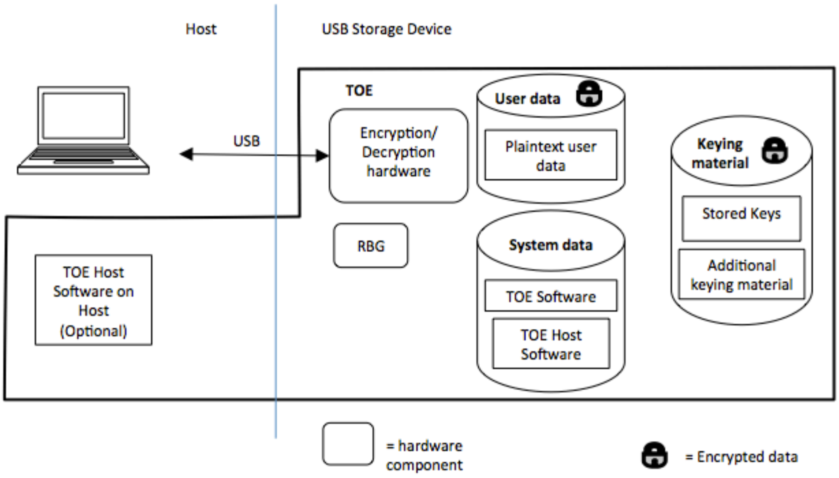

Figure 1: USB device boundaries, components and connections

It is intended to be used for the following scenarios:

- Transfer of sensitive data between two host computers.
- Long term storage of sensitive data.

In this cPP we refer to “authorisation” rather than “authentication” to reflect the fact that the device does not need to hold reference authentication data, or other data that identifies users as individuals. Authorisation data includes passphrases and possibly other data (such as cryptographic salt values), and is used to derive a Key Encryption Key (KEK). The KEK is used to encrypt a Data Encryption Key (DEK) that is generated on the device and used to encrypt and decrypt data stored on the device.

The host computer plays no role in the encryption/decryption of user data. Software running on the host computer could gather the necessary authorisation data but must not perform any other authorization functions. The TOE Host software is then considered as part of the TOE and is subject to requirements to provide certain functionality to users of the device.

In addition to user data, the device may also hold data that determines operation of the device itself (including software that may be downloaded to a host computer, such as driver software). This data is referred to as “system data”. System data may include (but are not limited to) software or firmware, patches, or configuration data. The TOE provides the integrity protection of the system data. The host computer plays no role in protecting the system data that resides on the device.

All data on the device is either user data, system data, keying material or else unused (unused areas of the device may contain old encrypted user data or old encrypted keying material).

Threats

A threat consists of a threat agent, an asset and an adverse action of that threat agent on that asset.

Threat agents

Threat agents are unauthorised users, i.e. they do not possess valid authorisation factors. They are referred to here as “attackers”. Attackers are typically characterised by a number of factors, such as expertise, available resources, and motivation, with motivation being linked directly to the value of the assets at stake. The device is designed to be resistant to an attacker possessing a basic attack potential.

[Note: Defining the meaning of the 'basic attack potential' for the USB Storage Device technology area to make it more precise is a subject of refinement within the USB iTC.]

Assets (resources to be protected)

The assets to be protected by the device are:

- the plaintext user data and keying material requires confidentiality protection, and
- system data requires integrity protection.

In general, some individual data items stored on the device will not be sensitive, but the cPP does not require distinction of security requirements at this more granular level: all user data and keying material is to be protected for confidentiality, and all system data is to be protected for integrity.

Adverse actions

The underlying consequences of all threats considered in this cPP are that an attacker could retrieve plaintext user data or keying material, or could modify system data.

Storage of User Data, Keying material and System Data

[T.UNAUTHORISED_USER_DATA_ACCESS]

The primary threat to be addressed is the unauthorised disclosure of user data stored on a device. Attackers may attempt to use the logical interface to access plaintext data, or may connect the device to a host that provides raw access to the device content (e.g. to specified disk sectors or blocks).

Attackers may gain physical access to the device, and thus bypass the logical interface to obtain access to user data (perhaps by making physical modifications to the device to access its memory via their own physical connections, or the attacker might use equipment appropriate for the attack potential to read the contents of memory locations from their physical state).

Attackers may also access user data that remains to be unprotected due to a failure interrupting correct operation of the device. Attackers may look for unencrypted keying material giving them unauthorised access to user data.

[T.UNAUTHORISED_SYSTEM_DATA_MODIFICATION]

Attackers may modify system data stored on a device. Attackers may attempt to use the logical interface to modify system data, or may connect the device to a host that provides raw access to the device content (e.g. to specified disk sectors or blocks).

[T.KEYING_MATERIAL_COMPROMISE]

Possession of any of the keys, authorisation data, random numbers or any other values that contribute to the creation of keys or authorisation data could allow an attacker to defeat the encryption. As part of a conservative approach to security, gaining access to keying material is considered to be of equal importance to gaining access to plaintext user data or system data itself. Attackers may look for keying material in unencrypted sectors of the drive, including in memory used to support power saving modes (in the TOE).

Alternatively an attacker might determine a key because of insufficient entropy used in its generation.

Authorisation

[T.AUTHORISATION_GUESSING]

Attackers may mount an exhaustive search (brute force) attack against the device to determine authorisation factors to gain unauthorised access to the user data stored on the device.

Updates

[T.UNAUTHORISED_UPDATE]

Attackers may attempt to perform an unauthorised update of the product, which compromises the security features of the device. Poorly chosen update protocols, signature generation/verification algorithms and parameters may allow attackers to install software and/or firmware that bypasses the intended security features and provides them unauthorised access to user data and keying material or allows modification of system data.

This threat includes attempts to make an unauthorised rollback of updates so that they are no longer applied, or to replay an old, valid update message containing a superseded update (which might allow a known vulnerability to be exploited).

Policies

[P.NO_STORE]

It is not possible to reconstruct the keys that protect user data from data persistently stored on the TOE. (This means that complete reference authorisation data is not persistently stored on the TOE.)

[P.RECOVERY]

If the TOE implements any mechanism to enable data recovery in the event of loss of authorisation data[1] then the TOE must allow this mechanism to be disabled, such that it cannot be re-enabled by an unauthorised user.

If no data recovery mechanism is provided by the TOE then this requirement is met. A TOE that provides a data recovery mechanism is required to allow it to be disabled via configuration of the device.

[P.CRYPTO]

The cryptographic algorithms, key lengths and modes used shall be in conformance with the requirements of the National authorised cryptographic authority.

[P.AUTH_CHANGE]

The TOE enables authorised users to change the value of the authorisation data, but only when supplying correct current authorisation data as part of the change operation.

[P.FULL_ENCRYPTION]

Only encrypted data storage shall be available to store user data (i.e. no unencrypted storage is available to users).

[P.NO_BOOT] – Enforced by the Operational environment

It shall not be possible to boot from the TOE in its evaluated configuration.

Assumptions

[A.USER_GUIDANCE]

Users will be instructed in secure use of the device.

[A.LOST_DEVICE]

The device shall be discarded in the event that the device is left unattended, lost or stolen and later recovered, if it may be suspected that an attacker has tampered with the device.

[1] Such a mechanism would typically be based on the generation of additional recovery keys that enable the DEK to be recovered via inputs other than the user's usual authorisation data.

[A.TRUSTED_HOST]

The host computer is trusted, free of malware that could interfere with the correct operation of the device. Only authorised users have access to the host computers.

[A.TRUSTED_CONNECTION]

Communication between host computer and the device is sufficiently protected to prevent disclosure of, or tampering with, user data, keying material or system data.

Glossary

Keying material	A data item that is used in combination with other data in order to derive a cryptographic key (e.g. a passphrase, seed, or each of the values used in an xor combination).
Target of Evaluation	The set of software, firmware and/or hardware, possibly accompanied by guidance, that implements the security requirements expressed in this Protection Profile

Abbreviations

cPP	Collaborative Protection Profile
DEK	Data Encryption Key
KEK	Key Encryption Key
RBG	Random Bit Generator
TOE	Target of Evaluation
USB	Universal Serial Bus