SDTP-0002 - Specification of commitment file format

Version 1.0 from 09-03-2019, initial version not deprecating any previous versions.

This document specifies the design and properties of the binary file format that is used for commitments created with SDTP. It is intended as a reference for the properties as well as a guideline for other software that might also implement it. The document is about version 1.

Purpose of the file format

The purpose of the file format is the storage of transportation for files which are used in relation to applied commitment schemes. This may either be a file that commits a certain statement or one that reveals a previously committed statement. It also contains a subject line that describes the content of the commitment from the beginning (it is therefore known before revelation) and therefore should not tell or give hints about the statement whose revelation is controlled by the owner of the commitment. Another payload data is the scheduled revelation time of the commitment, which is defined by the owner and known to the receiver from the beginning. It may be used to control if the claimed date for revealing the commitment has been satisfied.

Structure

This is the table for files committing a commitment:

Offset	Data type	Short description
0	const uint32_t	Magic value (file identifier)
4	const uint8_t	File format version
5	uint8_t	File type
6	uint8_t	ID
7	uint32_t	Scheduled revelation (SDTP Time)
11	uint8_t	Offset of actual data
12	char[]	Subject of the commitment
>12	uint8_t[32]	Commitment hash value

This is the table for files revealing a commitment:

Offset	Data type	Short description
0	const uint32_t	Magic value (file identifier)
4	const uint8_t	File format version
5	uint8_t	File type
6	uint8_t	ID
7	uint8_t	Offset of actual data
>8	char[]	Subject of the commitment
>9	char[]	Commitment string
>10	uint8_t[12]	Random entropy

All integers consisting of multiple octets to be stored in the network byte order, also called big endian.

Full description of variables contained in the file

In this section, all values that are included in the binary file format for handling commitments are explained in detail. If they only exist under certain conditions or differ in their properties, these behavior is explained further.

Magic value

An unsigned 32 bit value that is always present at the first four octets of the file and shall indicate that an SDTP commitment file is present. Developers of other file formats are asked to avoid confusion by not using this value at the beginning of their binary formats.

The value always is `0x6eb41a5a` - this is a random sequence of four bytes with no further meaning.

File format version

The version of the file format, which is `0x00` for the format described in this specification. Since any implementation shall refuse operation if the version number is higher then the one it knows how to process, this value shall only be incremented if backward compatibility is not possible or undesirable, for example, because a version has fundamental security flaws. If additional information that can be ignored is added, this shall be done by increasing the offset to a suitable number for adding additional data.

File type

Indicates, whether the file is a commitment or a revelation. If the least significant bit is false/unset, it is a commitment; otherwise, if it is true/set, it is a revelation. The other bits of the octet are ignored and might be used as additional flags for backward compatibility preserving further information, similar to the increasing of offset.

ID

A random value whose only required property is to be consistent over commitment and revelation file (any commitment and revelation must have the same ID). ID and subject are used together for unique identification of a commitment.

Scheduled revelation (SDTP time)

An unsigned 32 bit value of the scheduled time the commitment is to be revealed. This only applies to commitment files (File types least significant bit is false), as this property does not make sense for revelations. SDTP time as defined in SDTP-0001 is used.

Offset of actual data

The number of octets from the beginning of the file to the actual begin of the *Subject of the commitment*. This data before the subject can be considered to be the header and its actual size can be modified later to insert additional data while maintaining backward compatibility.

Subject of the commitment

A UTF-8 string with up to 64 bytes that is user defined and shall describe the commitment without revealing it prematurely or giving any hints about the actual commitment. Example: If a commitment commits to a certain birthday present, the subject might be `Birthday present`. This will label the commitment without telling the actual present. The string is NULL-terminated. It shall be equal for any pair of commitment and its revelation.

Hash value (SHA-256)

A series of 32 bytes containing the hash value of the commitment string. This is only applicable and therefore present in the commitment file (not the revelation file). The hash function used is SHA-256 as defined in NIST FIPS PUB 180-4.

Commitment string

A UTF-8 string with the actual commitment, with a length of up to 1024 bytes. It is only present in a revelation file and does not contain a NULL-terminator as the well-defined location (after the NULL-terminator of the *Subject of the commitment*, before the constant number of 12 entropy octets) makes it redundant and memory consumption can be reduced.

Random entropy

Only exists for revelation files. A set of 12 octets which is part of the commitment string when calculating its hash value, either for creating the revelation file or for verifying the commitment, but not when displaying the commitment string to the user under normal circumstances (unlike debugging, for example).

Minimum and maximum file size

A commitment file has a file size of at least 45 bytes. This assumes an empty subject (only the NULL-terminator). The maximum file size is 352 bytes, assuming the maximum offset of 255 bytes and the maximum subject length of 64 bytes.

A revelation file has a file size of at least 22 bytes, assuming both an empty subject and an empty commitment. The maximum file length is 1356 bytes, assuming the maximum offset as well as a maximum subject of 64 bytes and a maximum commitment of 1024 bytes.

Security considerations

There is no known possibility to efficiently find the commitment text with only a file establishing a commitment being given. The file format itself is unprotected as it is intended to be used with other tools such as GnuPG or SDTP (as soon as the entire project is finished) which will provide integrity and secrecy. Especially, it is easy to read the commitments subject with any of the two files and learning the commitment with access to the revelation file. It is also easy to modify the random entropy so third parties can pretend that the owner of the commitment tried to cheat. This can be avoided with digital signatures or operation modes for block ciphers which don’t allow flipping single sections without destroying the readability of the file in general.

An adversary which is able to delay the data transmission might render a commitment useless by delaying the commitment or its revelation for a too long time.

Implementation

The upcoming re-implementation of hcommit which uses binary files will implement this standard. A dedicated library for handling the commitment files is not intended for the simplicity and close integration with the operation of hcommit itself.