NAME

gttlvundump - A tool for constructing TLV file from human-readable input (see **gttlvdump**(1)).

SYNOPSIS

gttlvundump [-h] [-v] [*file...***]**

DESCRIPTION

gttlvundump is a tool that constructs a binary TLV file from input in human-readable format defined by **gttlvdump**. If input file is not specified, input is read from *stdin*. When multiple input files are specified, the result is concatenated starting from the first (left-most) file. It must be noted that only default output format of **gttlvdump** is accepted as input. The output is written to the *stdout*.

It is possible to use functions for composing TLV files. For detailed description of supported functions see **FUNCTIONS** section. To use a function it has to be preceded by '**$**' sign, i.e. **$FUNC**(*arg*). Function arguments are delimited with '**|**' sign.

OPTIONS

-h Print help text.

-v Print TLV utility version.

FUNCTIONS

HMAC(*version*|*algorithm*|*key*|*pattern*)

Compute HMAC value on a set of TLV elements.

version HMAC computation version. Valid versions:
v1 - computation is performed for each TLV element over the concatenation of their header and value.
v2 - computation is performed for the whole set of TLV elements over the TLV set header, each TLV element header and value in the order in which they appear within the TLV file, and the header and the hash function ID of the MAC element itself.

algorithm
Hash algorithm to be used for computation. Use **-h** to get the list of supported hash algorithms.

key Secret cryptographic key for computing HMAC.

pattern TLV pattern describing TLVs to be included into computation (valid with *v1*). Pattern format as defined by **gttlvgrep**.

EXIT STATUS

0 **Exit success.** Returned if everything is OK.

1 **Exit failure.** A general failure occurred.

3 **Invalid command-line parameter.** The content or format of a command-line parameter is invalid or a parameter is missing.

4 **Invalid format.** Input data can not be parsed or data format is invalid.

9 **Input/output error.** Unable to read or write file or stream.

10 **Cryptographic error.** Cryptographic operation could not be performed. Likely causes are unsupported or unknown cryptographic algorithms during HMAC calculation.

13 **System out of memory.**

EXAMPLES

1 Generate a TLV file from the given input file and dump the result with **gttlvdump**:

File *test.tlv* contains following TLV structure description.

```
TLV[1000]:
 TLV[01]:
  TLV[01]:0a
```

TLV[01]:0b

gttlvundump *test.tlv* | gttlvdump

2 Concatenate two TLVs described in *a.tlvdump* and *b.tlvdump* and print the result in human-readable fromat:

gttlvundump *a.tlv b.tlv* | gttlvdump

3 Calculate HMAC value on a set of TLVs:

1. *v1* HMAC calculation.

```
TLV[0300]:
  TLV[01]:
    TLV[01]:616E6F6E00
  TLV[0301]:
    TLV[01]:01
    TLV[02]:54D9D6E7
    TLV[03]:54D9D6E7
  TLV[1f]:$HMAC(v1|sha256|anon|300.01,301)
```

2. *v2* HMAC calculation.

```
TLV[0300]:
  TLV[01]:
    TLV[01]:616E6F6E00
  TLV[0301]:
    TLV[01]:01
    TLV[02]:54D9D6E7
    TLV[03]:54D9D6E7
  TLV[1f]:$HMAC(v2|sha256|anon)
```

AUTHOR

Guardtime AS, http://www.guardtime.com/

SEE ALSO

gttlvdump(1), **gttlvgrep**(1), **gttlvwrap**(1), **tlv**(5), **tlv-desc**(5)