

Kvaser REST API Specification

Copyright 2011-2026 Kvaser AB, Mölndal, Sweden
https://www.kvaser.com

Printed Sunday 31st May, 2026

We believe that the information contained herein was accurate in all respects at the time of printing. Kvaser AB cannot, however, assume any responsibility for errors or omissions in this text. Also note that the information in this document is subject to change without notice and should not be construed as a commitment by Kvaser AB.

(This page is intentionally left blank.)

Contents

1 About this Manual **5**

2 Introduction **6**

3 Connection Functions **7**
3.1 deviceStatus . . . 7
3.2 canInitializeLibrary . . . 7
3.3 canUnloadLibrary . . . 8

4 CANlib Equivalent Functions **9**
4.1 canOpenChannel . . . 9
4.2 canClose . . . 9
4.3 canSetBusParams . . . 10
4.4 canBusOn . . . 10
4.5 canBusOff . . . 11
4.6 canSetBusOutputControl . . . 11
4.7 canRead . . . 11
4.8 canWrite . . . 12
4.9 canIoCtl . . . 13
4.10 canReadTimer . . . 13
4.11 canAddFilter . . . 13
4.12 canClearFilters . . . 15
4.13 kvScriptStatus . . . 15
4.14 kvScriptLoadFile . . . 16
4.15 kvScriptStart . . . 16
4.16 kvScriptStop . . . 17
4.17 kvScriptLoadFileOnDevice . . . 17
4.18 kvScriptUnload . . . 18
4.19 kvScriptSendEvent . . . 18
4.20 kvScriptEnvvarOpen . . . 19
4.21 kvScriptEnvvarClose . . . 19
4.22 kvScriptEnvvarSetInt . . . 19
4.23 kvScriptEnvvarGetInt . . . 20
4.24 kvScriptEnvvarSetFloat . . . 20
4.25 kvScriptEnvvarGetFloat . . . 21
4.26 kvScriptEnvvarSetData . . . 21
4.27 kvScriptEnvvarGetData . . . 22
4.28 kvFlashLeds . . . 22
4.29 kvFileGetCount . . . 23
4.30 kvFileDiskFormat . . . 23
4.31 kvFileGetName . . . 23
4.32 kvFileDelete . . . 24
4.33 kvFileCopyFromDevice . . . 24

4.34 kvFileCopyToDevice 25
4.35 kvScriptRequestText 26
4.36 kvScriptGetText 26

5 Connection flow examples **27**

6 Document Revision History **31**

1 About this Manual

This document specifies the JSON REST API that is available in selected Kvaser CAN interfaces. The reader is assumed to be familiar with Kvaser CANlib.



THIS SPECIFICATION IS SUBJECT TO CHANGE!

2 Introduction

This document specifies the JSON REST API which is available in selected Kvaser CAN interfaces. The API is based upon the Kvaser CANlib, so most function and parameter names are the same. Assuming that the device is connected, has IP address 192.168.1.10, and is listening on port 8080, you can access the API in the form:

`http://192.168.1.10:8080/deviceStatus`

The following rules applies to this API:

- All constants must be specified with its numerical value, e.g. `canBITRATE_1M` should be given as `-1`
- All numbers are decimal
- All calls can take an optional integer parameter, `ident=` which is included in the response

The status constants that are currently used by the JSON REST API are listed in Table 1.

Constant	Value
canOK	0
canERR_PARAM	-1
canERR_NOMSG	-2
canERR_NOMEM	-4
canERR_NOCHANNELS	-5
canERR_TIMEOUT	-7
canERR_INVHANDLE	-10
canERR_HARDWARE	-15
canERR_NOT_IMPLEMENTED	-32
canERR_DEVICE_FILE	-33
canERR_SCRIPT_FAIL	-39
canERR_SCRIPT_WRONG_VERSION	-40
canERR_INVALID_PASSWORD[1]	-128
canERR_NO_SUCH_FUNCTION[1]	-129
canERR_NOT_AUTHORIZED[1]	-130
canERR_INVALID_SESSION[1]	-131

Table 1: Status constants used by the JSON REST API. Note that the last constants are extensions to current Kvaser CANlib.

[1] Extensions to current Kvaser CANlib.

3 Connection Functions

The following functions are used to query and connect to a device. This is done differently than in Kvaser CANlib, e.g. the session concept is new.

3.1 deviceStatus

You may at any time ask a device for its status with the function `deviceStatus`. The device can then respond whether it is free or already connected to some host.

- `uri:/deviceStatus`
- parameters:

  [mode=jsonp] If set, the response will be coded in JSONP, i.e. wrapped with the fixed string 'canlib_callback(...)'.

- returns:

```
"usage" : %u,          # Flags: 0=Free, 1=In use via service,
                       # 2=In use via JSON API,
                       # 4=In use via JSONP API

# The following response codes are only present if the device
# is in use:
"timeout" : %u,        # time left before current session
                       # will end (seconds)
"ip" : "%u.%u.%u.%u"   # IP address of the host the device
                       # is currently connected to.
```

- example:

```
http://192.168.1.10:8080/deviceStatus?mode=jsonp

canlib_callback({"usage":1})

http://192.168.1.10:8080/deviceStatus?mode=jsonp&ident=0001

canlib_callback({"usage":0, "ident":1})

http://192.168.1.10:8080/deviceStatus

{"usage":1}

http://192.168.1.10:8080/deviceStatus

{"usage":3, "timeout":1066, "ip":"192.168.1.12"}
```

3.2 canInitializeLibrary

The function `canInitializeLibrary` sets up a connection and creates a session. Therefore, this routine must be called before any other function that needs a

session. When a session is active, the device will deny any further calls to this function by returning status code -131 (Invalid session).

The returned session must be used when calling other functions, denoted with <session> below. A session is terminated either by an explicit call to `canUnloadLibrary` or after an inactivity of more than 'timeout' seconds.

- `uri:/canInitializeLibrary`
- parameters:

  [password=%s] Access password, the string can be URI encoded if needed.

  [mode=jsonp] When set, the response will be coded as JSONP.

  [timeout=%u] The session timeout in seconds, default is 900.

  [dummy_session=%u] If set to '1', the session returned will be '00000000000000000000000000000000'.

- returns:

  `{"stat":<canOK | canERR_xxx>, "session":"%32x"}`

- example:

  `http://192.168.1.10:8080/canInitializeLibrary?timeout=120`

  `{"stat":0, "session":"1b6ed79f755f0ab94ff9ad62470ad0a0"}`

  `http://192.168.1.10:8080/canInitializeLibrary?timeout=120`

  `{"stat":-131}`

3.3 canUnloadLibrary

The function `canUnloadLibrary` terminates an active session, making the device free.

- `uri:/<session>/canUnloadLibrary`
- parameters: None.
- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  `http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canUnloadLibrary`

  `{"stat":0}`

4 CANlib Equivalent Functions

The following functions mirror functions in Kvaser CANlib. Please refer to the CANlib documentation for further information about these functions and their parameters.

Note that all constants must be specified with their decimal numerical values, e.g. `canBITRATE_1M` should be given as `-1`. All return values will also be returned as decimal integers, e.g. `canOK` will be returned as `0`.

4.1 canOpenChannel

The function `canOpenChannel` returns a handle to the opened channel. This handle should be passed in other functions as the `hnd` parameter as needed.

- `uri:<session>/canOpenChannel`
- parameters:

  channel=%u Channel number on the device.

  flags=%u Flags according to canOPEN_xxx.

- returns:

```
{"stat":<canOK | canERR_xxx>, "hnd":%d}
# hnd is only valid if stat returns canOK
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canOpenChannel?
    channel=0&flags=8

{"stat":0, "hnd":0}
```

4.2 canClose

- `uri:<session>/canClose`
- parameters:

  hnd=%u

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canClose?hnd=0

{"stat":0}
```

4.3 canSetBusParams

- `uri:<session>/canSetBusParams`
- parameters:

  hnd=%u

  freq=%d Bit rate, or one of canBITRATE_xxx. If freq is not any of canBITRATE_xxx, the following parameters must also be set:

  tseg1=%u

  tseg2=%u

  sjw=%u

  noSamp=%u

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  `http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canSetBusParams?hnd=0&freq=-1&ident=1234`

  `{"stat":0, "ident":1234}`

4.4 canBusOn

- `uri:<session>/canBusOn`
- parameters:

  hnd=%u

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  `http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canBusOn?hnd=0`

  `{"stat":0}`

4.5 canBusOff

- `uri:<session>/canBusOff`
- parameters:

  hnd=%u

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  `http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canBusOff?hnd=0`

  `{"stat":0}`

4.6 canSetBusOutputControl

- `uri:<session>/canSetBusOutputControl`
- parameters:

  hnd=%u

  drivertype=%u Driver type according to canDRIVER_xxx.

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  `http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/`
  `canSetBusOutputControl?hnd=0&drivertype=4`

  `{"stat":0}`

4.7 canRead

- `uri:<session>/canRead`
- parameters:

  hnd=%u

  max=%u Max number of messages to be received in the answer, default is 1.

- returns:

```
{"stat":<canOK | canERR_xxx>,
 "msgs:" : [
   {
     "id" : %u,
     "dlc"  : %u,
     "time" : %u,
     "flag" : %u,
     "msg"  : [%u, ...]
   },
   ...
 ]
}
```

- example:

```
192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canRead?hnd=0&max=5

{"stat":0, "msgs":[
  {"id":10, "dlc":4, "msg":[67,12,8,0], "time":3520517614,"flag":2},
  {"id":12, "dlc":3, "msg":[32,12,16], "time":3520517724,"flag":2},
  {"id":14, "dlc":4, "msg":[0,0,24,0], "time":3520517835,"flag":2}]}
```

4.8 canWrite

- `uri:<session>/canWrite`
- parameters:

  hnd=%u

  id=%u

  flag=%u

  dlc=%u

  msg=%u[,%u[...]]

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canWrite?hnd=0&
    id=55&flag=0&msg=99,100,101,102,103,104,105&dlc=7

{"stat":0}
```

4.9 canIoCtl

- `uri:<session>/canIoCtl`
- parameters:

  hnd=%u

  func=%u Function according to canIOCTL_xxx.

  buf=%s Parameter depending on the actual function used.
- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canIoCtl?hnd=0&
    func=10

{"stat":0}
```

4.10 canReadTimer

- `uri:<session>/canReadTimer`
- parameters:

  hnd=%u
- returns:

  `{"stat":<canOK | canERR_xxx>, "time":%lu}`

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canReadTimer?
    hnd=0

{"stat":0, "time":3564353987}
```

4.11 canAddFilter

Filter messages on Id. Filters can be either Pass, Stop or Counting Pass filters. For Counting Pass filters: counter goes from 0 to counterMax-1 and if counter < counterThreshold then the filter is active, otherwise inactive. Counting filters only work on single Id's.

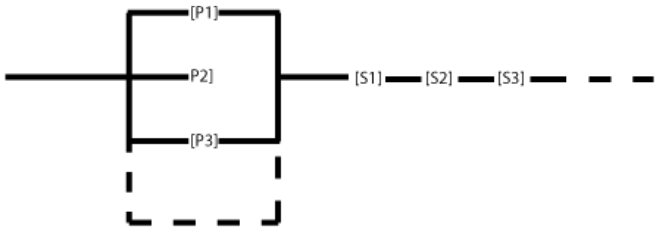

Figure 1: Filter layout

- `uri:<session>/canAddFilter`
- parameters:

  hnd=%u

  type=%u Type can be either PASS [1], STOP [2] or COUNTING_PASS [3].

  idMin=%u <optional> Used when specifying an Id range. Specifies min id in an id range.

  id=%u Specifies id or upper id when used in an Id range.

  flags=%u Can be either canMSG_STD [2] for standard (11-bit) id:s or canMSG_EXT [4] for extended (29-bit) Id's.

  counterThreshold=%u Used with counting pass filters. Specifies the threshold for counting to.

  counterMax=%u Used with counting pass filters. Specifies the max number that the counter counts to.

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example: pass filter

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canAddFilter?
  hnd=0&type=1&idMin=10&id=20&flags=2
{"stat":0}
```

- example: stop filter

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canAddFilter?
  hnd=0&type=2&idMin=15&id=16&flags=2
{"stat":0}
```

- example: counting pass filter, accept 1 in 100 messages

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/canAddFilter?
  hnd=0&type=3&id=1&counterThreshold=1&counterMax=100&flags=2
{"stat":0}
```

4.12 canClearFilters

Clear all filters.

- `uri:<session>/canClearFilters`
- parameters:

  hnd=%u

- returns:

  ```
  {"stat":<canOK | canERR_xxx>}
  ```

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    canClearFilters?hnd=0

  {"stat":0}
  ```

4.13 kvScriptStatus

- `uri:<session>/kvScriptStatus`
- parameters:

  hnd=%u

  slotNo=%u

- returns:

  ```
  {"stat":<canOK | canERR_xxx>, "scriptStatus" : %u}
    # status flag bits 1=kvSCRIPT_STATUS_LOADED
    #                  2=kvSCRIPT_STATUS_RUNNING
  ```

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptStatus?hnd=0&slotNo=0

  {"stat":0, "status":3}
  ```

4.14 kvScriptLoadFile

Uploads a compiled script file specified in the request body to the given script slot on the device.

- `uri:<session>/kvScriptLoadFile`
- parameters:

  hnd=%u

  slotNo=%u
- request headers:

  Content-Type=application/octet-stream

  Content-Length=%u size of file in bytes
- returns:

  `{"stat":<canOK | canERR_xxx>}`
- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptLoadFile?hnd=0&slotNo=0

Excerpt from HTTP POST request:
Content-Length: 1496
Content-Type: application/octet-stream
Media type: application/octet-stream (1496 bytes)

{"stat":0}
```

4.15 kvScriptStart

- `uri:<session>/kvScriptStart`
- parameters:

  hnd=%u

  slotNo=%u
- returns:

  `{"stat":<canOK | canERR_xxx>}`
- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/kvScriptStart
  ?hnd=0&slotNo=0

{"stat":0}
```

4.16 kvScriptStop

- `uri:<session>/kvScriptStop`
- parameters:

  hnd=%u

  slotNo=%u

  mode=%u kvSCRIPT_STOP_NORMAL=0 or kvSCRIPT_STOP_FORCED=-9

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/kvScriptStop?
  hnd=0&slotNo=0&mode=0

{"stat":0}
```

4.17 kvScriptLoadFileOnDevice

- `uri:<session>/kvScriptLoadFileOnDevice`
- parameters:

  hnd=%u

  slotNo=%u

  localFile=%s Filename on the device, shall be prefixed with F:/ for devices with built-in flash memory only

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptLoadFileOnDevice?hnd=0&localFile=F:/myscript.txe&slotNo=0

{"stat":0}
```

4.18 kvScriptUnload

- `uri:<session>/kvScriptUnload`
- parameters:

  hnd=%u

  slotNo=%u

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptUnload?hnd=0&slotNo=0

  {"stat":0}
  ```

4.19 kvScriptSendEvent

- `uri:<session>/kvScriptSendEvent`
- parameters:

  hnd=%u

  slotNo=%u

  eventType=%u only kvEVENT_TYPE_KEY = 1 is implemented

  eventNo=%u

- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example: send key event 'a' to slot 0

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptSendEvent?hnd=0&eventNo=97&eventType=1&slotNo=0

  {"stat":0}
  ```

4.20 kvScriptEnvvarOpen

- `uri:<session>/kvScriptEnvvarOpen`
- parameters:

  hnd=%u

  envvarName=%s

- returns:

```
{"stat":<canOK | canERR_xxx>, "eHnd":%u, "envvarType": kvENVVAR_TYPE_xxx,
     "envvarSize": %u}
#eHnd is the handle to use in further access to the envvar.
```

- example: open myenvvarint

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarOpen?hnd=0&envvarName=myenvvarint

{"stat":0, "eHnd":0, "envvarType":1, "envvarSize":4}
```

4.21 kvScriptEnvvarClose

- `uri:<session>/kvScriptEnvvarClose`
- parameters:

  eHnd=%u

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarClose?eHnd=0

{"stat":0}
```

4.22 kvScriptEnvvarSetInt

- `uri:<session>/kvScriptEnvvarSetInt`
- parameters:

  eHnd=%u

  val=%u
- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptEnvvarSetInt?eHnd=0&val=42

  {"stat":0}
  ```

4.23 kvScriptEnvvarGetInt

- `uri:<session>/kvScriptEnvvarGetInt`
- parameters:

  eHnd=%u
- returns:

  `{"stat":<canOK | canERR_xxx>, "val": %d}`

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptEnvvarGetInt?eHnd=0

  {"stat":0, "val":42}
  ```

4.24 kvScriptEnvvarSetFloat

- `uri:<session>/kvScriptEnvvarSetFloat`
- parameters:

  eHnd=%u

  val=%f
- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarSetFloat?eHnd=0&val=4.2

{"stat":0}
```

4.25 kvScriptEnvvarGetFloat

- `uri:<session>/kvScriptEnvvarGetFloat`
- parameters:

  eHnd=%u

- returns:

```
{"stat":<canOK | canERR_xxx>, "val": %f}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarGetFloat?eHnd=0

{"stat":0, "val":4.200000}
```

4.26 kvScriptEnvvarSetData

- `uri:<session>/kvScriptEnvvarSetData`
- parameters:

  eHnd=%u

  buf=%x Hexadecimal string of the binary data.

  startIndex=%u Currently only 0 is supported

  dataLen =%u The number of bytes to write. Currently only writing the full envvar is supported.

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example: set envvar to Hello

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarSetData?eHnd=0&buf=48656c6c6f&startIndex=0&dataLen=5

{"stat":0}
```

- example: set envvar to 0x12, 0x34, 0xab, 0xcd,

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarSetData?eHnd=0&buf=1234abcd&startIndex=0&dataLen=4

{"stat":0}
```

4.27 kvScriptEnvvarGetData

- `uri:<session>/kvScriptEnvvarGetData`
- parameters:

  eHnd=%u

  startIndex=%u

  dataLen =%u
- returns:

```
{"stat":<canOK | canERR_xxx>, "buf": "%x"}
```

- example: read an envvar with the value Hello

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvScriptEnvvarGetData?eHnd=0&startIndex=0&dataLen=5

{"stat":0, "buf":"48656c6c6f"}
```

4.28 kvFlashLeds

- `uri:<session>/kvFlashLeds`
- parameters:

  action=%u One of the kvLED_ACTION_xxx constants, defining which LED to turn on or off.

  timeout=%u Specifies the time, in milliseconds, during which the action is to be carried out. When the timeout expires, the LED(s) will return to its ordinary function. If not set, the LED(s) will keep its state until the next update.
- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/kvFlashLeds?
  action=0&timeout=1024

{"stat":0}
```

4.29 kvFileGetCount

- `uri:<session>/kvFileGetCount`
- parameters:

  hnd=%u
- returns:

  `{"stat":<canOK | canERR_xxx>, "count": %u}`

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvFileGetCount?hnd=0

  {"stat":0, "count":1}
  ```

4.30 kvFileDiskFormat

- `uri:<session>/kvFileDiskFormat`
- parameters:

  hnd=%u
- returns:

  `{"stat":<canOK | canERR_xxx>}`

- example:

  ```
  http://192.168.1.10:8080/6090a77b21278bc53fb33a1612580863/
    kvFileDiskFormat?hnd=0

  {"stat":0}
  ```

4.31 kvFileGetName

- `uri:<session>/kvFileGetName`
- parameters:

  hnd=%u

  fileNo=%u
- returns:

  `{"stat":<canOK | canERR_xxx>, "name": "%s"}`

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/kvFileGetName
  ?hnd=0&fileNo=0

{"stat":0, "name":"F:/myscript.txe"}
```

4.32 kvFileDelete

- `uri:<session>/kvFileDelete`
- parameters:

  hnd=%u

  deviceFileName=%s Shall be prefixed with F:/ for devices with built-in flash memory only

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/kvFileDelete?
  hnd=0&deviceFileName=F:/myscript.txe

{"stat":0}
```

4.33 kvFileCopyFromDevice

Downloads a file as an attachment from the device

- `uri:<session>/kvFileCopyFromDevice`
- parameters:

  hnd=%u

  deviceFileName=%s Shall be prefixed with F:/ for devices with built-in flash memory only

- returns response headers:

  Content-Type=application/octet-stream

  Content-Length=%u size of file in bytes

  Content-Disposition=attachment; filename=%s Will be prefixed with F:/ for devices with built-in flash memory

- example: Copy F:/myscript.txe from the device.

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvFileCopyFromDevice?hnd=0&deviceFileName=F:/myscript.txe


Excerpt from HTTP Response header:
Content-Length: 2047
Content-Type: application/octet-stream
Content-Disposition: attachment; filename="F:/myscript.txe"
```

4.34 kvFileCopyToDevice

Uploads a file specified in the request body to the device.

- `uri:<session>/kvFileCopyToDevice`
- parameters:

  hnd=%u

  deviceFileName=%s Shall be prefixed with F:/ for devices with built-in flash memory only

- request headers:

  Content-Type=application/octet-stream

  Content-Length=%u size of file in bytes

- returns:

```
{"stat":<canOK | canERR_xxx>}
```

- example:

```
http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
  kvFileCopyToDevice?hnd=0&deviceFileName=F:/myscript.txe

Excerpt from HTTP POST request:
Content-Length: 2047
Content-Type: application/octet-stream
Media type: application/octet-stream (2047 bytes)

{"stat":0}
```

4.35 kvScriptRequestText

- `uri:<session>/kvScriptRequestText`
- parameters:

  hnd=%u

  slot=%u

  request=%u kvSCRIPT_REQUEST_TEXT_UNSUBSCRIBE=1, kvSCRIPT_REQUEST_TEXT_SUBSCRIBE=2

- returns:

  ```
  {"stat":<canOK | canERR_xxx>}
  ```

- example: Subscribe to printf from script in slot 2

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptRequestText?hnd=0&request=2&slot=2

  {"stat":0}
  ```

4.36 kvScriptGetText

- `uri:<session>/kvScriptGetText`
- parameters:

  hnd=%u

- returns:

  ```
  {"stat":<canOK | canERR_xxx>, "buf":"%s"}
  ```

- example:

  ```
  http://192.168.1.10:8080/1b6ed79f755f0ab94ff9ad62470ad0a0/
    kvScriptGetText?hnd=0

  {"stat":0, "buf":"Hello world"}
  ```

5 Connection flow examples

This capter contains a number of flow examples.

- Successful log in, Figure 2.
- Unsuccessful log in, Figure 3.
- Log in, send a CAN message and clean up, Figure 4 on Page 28.
- Invalid channel, Figure 5 on Page 29.
- Read CAN messages (messages available), Figure 6 on Page 29.
- Read CAN messages (no messages available), Figure 7 on Page 30.
- Error examples, Figure 8 on Page 30.

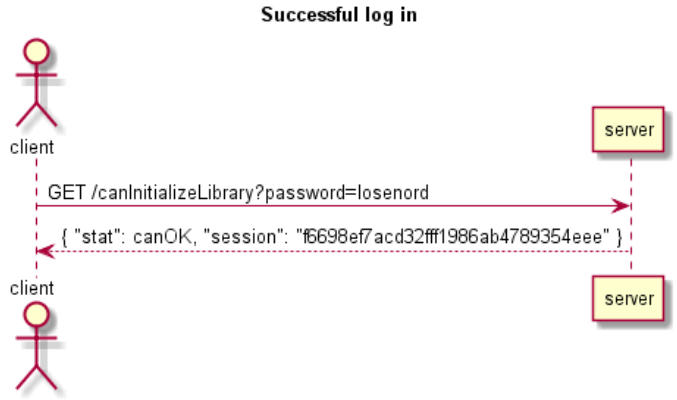

Figure 2: Successful log in

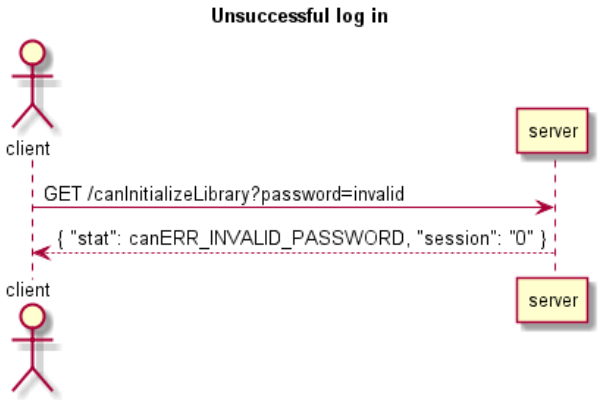

Figure 3: Unsuccessful log in



Figure 4: Log in, send a CAN message and clean up



Figure 5: Invalid channel

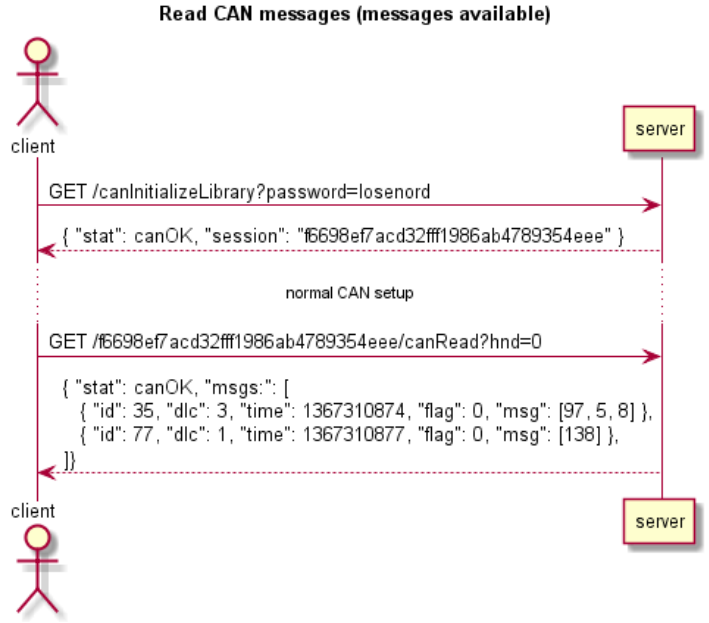

Figure 6: Read CAN messages (messages are available)

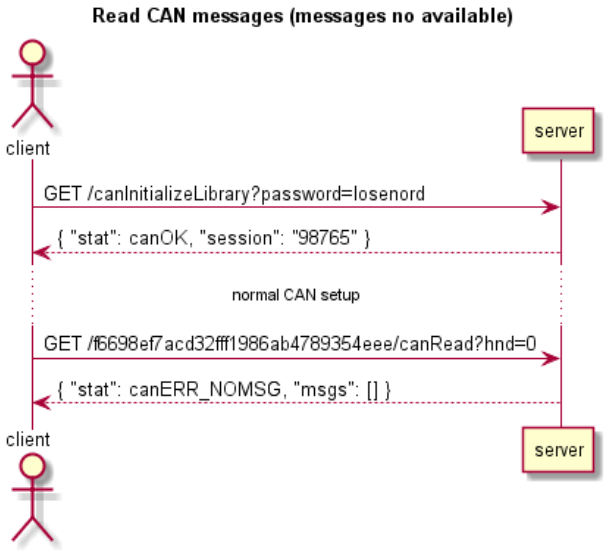

Figure 7: Read CAN messages (no messages are available)

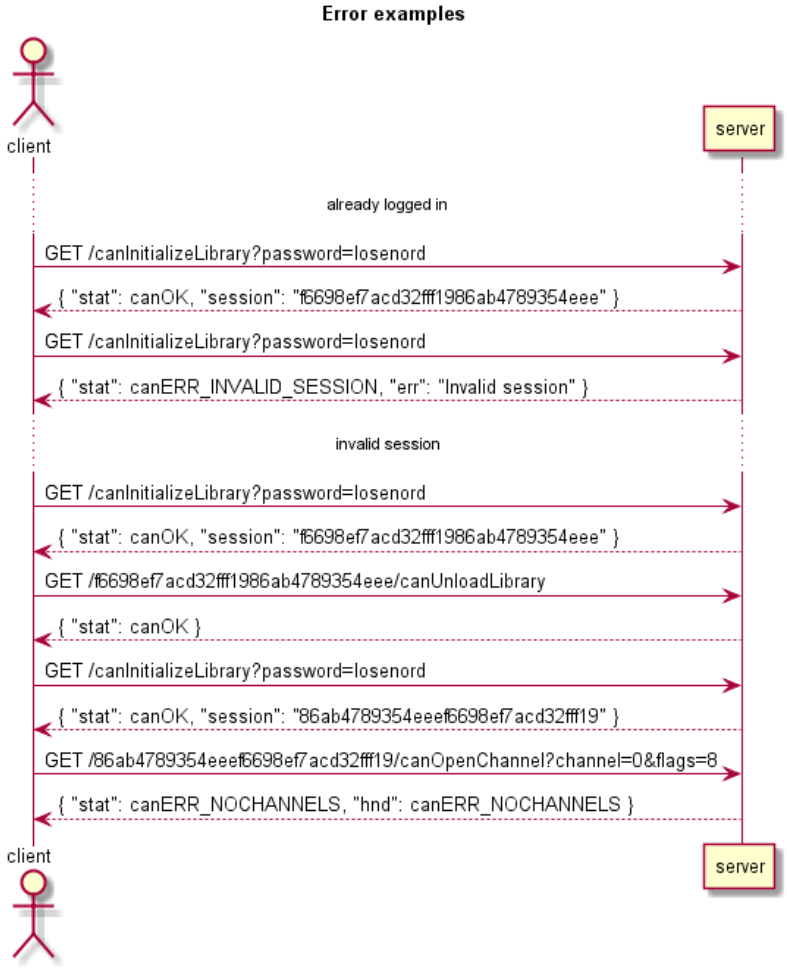

Figure 8: Error examples

6 Document Revision History

Version history for document IN_98151_kvaser_rest_api_specification:

Revision	Date	Changes
-	2013-09-23	Initial version.
1	2014-08-19	Minor updates. First public version. Changed layout of references, figures.
1.1	2014-10-02	Added description of dummy_session.
1.2	2015-12-11	Added sections about CAN filtering.
2.0	2017-11-10	Added sections for kvScript* and kvFile* functions. canAddFilter parameters changed to camelCase.
2.1	2019-08-09	Url protocol updated.
2.2	2024-03-15	Updated Kvaser logo.