

# TPLOGQINFO(3)

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

# Contents

<a href="#">1</a>	<a href="#">SYNOPSIS</a>	<a href="#">1</a>
<a href="#">2</a>	<a href="#">DESCRIPTION</a>	<a href="#">2</a>
<a href="#">3</a>	<a href="#">RETURN VALUE</a>	<a href="#">3</a>
<a href="#">4</a>	<a href="#">ERRORS</a>	<a href="#">4</a>
<a href="#">5</a>	<a href="#">EXAMPLE</a>	<a href="#">5</a>
<a href="#">6</a>	<a href="#">BUGS</a>	<a href="#">6</a>
<a href="#">7</a>	<a href="#">SEE ALSO</a>	<a href="#">7</a>
<a href="#">8</a>	<a href="#">COPYING</a>	<a href="#">8</a>

## Chapter 1

# SYNOPSIS

```
#include <ndebug.h> #include <nerror.h>
```

```
long tplogqinfo(int lev, long flags);
```

Link with *-lnstd -lpthread -lrt -lm*

## Chapter 2

# DESCRIPTION

Function is used to query information about current thread logging configuration. By default if *lev* is passed and infos are queried for given loggers (passed in *flags*):

```
#define TPLOGQI_GET_NDRX      0x00000001
#define TPLOGQI_GET_UBF      0x00000002
#define TPLOGQI_GET_TP       0x00000004
```

and if *lev* is higher, than currently configured, function will return **0**. If *lev* equals or is lower than currently configured log level, the informations are returned. Returned infos may be forced to be returned by **TPLOGQI\_EVAL\_RETURN** flag. The *iflags* of the loggers can be evaluate, and currently supported operation is **TPLOGQI\_EVAL\_DETAILED** (in flags), if set, the test for **detailed** flag is made. If flag is present in **iflags** of logger config, then return value will contain then **TPLOGQI\_RET\_HAVDETAILED** bit.

## Chapter 3

# RETURN VALUE

On success, **tplogqinfo()** returns following infos in 32bit integer: first 0x0000ffff bits contains the actual logger facility currently in use for matched **TPLOGQI\_GET** logger:

```
#define LOG_FACILITY_NDRX      0x00000001
#define LOG_FACILITY_UBF      0x00000002
#define LOG_FACILITY_TP       0x00000004
#define LOG_FACILITY_TP_THREAD 0x00000008
#define LOG_FACILITY_TP_REQUEST 0x00000010
#define LOG_FACILITY_NDRX_THREAD 0x00000020
#define LOG_FACILITY_UBF_THREAD 0x00000040
#define LOG_FACILITY_NDRX_REQUEST 0x00000080
#define LOG_FACILITY_UBF_REQUEST 0x00000100
```

If **detailed** flag is evaluated positive, then following bit is returned too in return value:

```
#define TPLOGQI_RET_HAVDETAILED 0x00010000
```

The current debug level for given logger is returned in oldest 8 bits of the return value, i.e. extracted in mask 0xff000000.

On error, -1 is returned, with **Error** set to indicate the error.

## Chapter 4

# ERRORS

Note that **Nstrerror()** returns generic error message plus custom message with debug info from last function call.

**NEINVAL** Invalid flags passed to function. None of **TPLOGQI\_GET\_NDRX**, **TPLOGQI\_GET\_UBF** or **TPLOGQI\_GET\_TP** are present.

---

## Chapter 5

# EXAMPLE

See `ubftest/test_nstd_debug.c` for sample code.



## Chapter 6

# BUGS

Report bugs to [support@mavimax.com](mailto:support@mavimax.com)

## Chapter 7

## SEE ALSO

**tplogdump(3)** **tplogdumpdiff(3)** **tplog(3)** **tplogsetreqfile\_direct(3)** **tplogsetreqfile(3)** **ex\_devguide(guides)** **ndrxdebug.conf(5)**  
**tplogconfig(3)**

## Chapter 8

# COPYING

© Mavimax, Ltd