

NAME

xo_syslog, **xo_vsyslog**, **xo_open_log**, **xo_close_log**, **xo_set_logmask** - create SYSLOG (RFC5424) log records using libxo formatting

LIBRARY

Text, XML, JSON, and HTML Output Emission Library (libxo, -lxo)

SYNOPSIS

```
#include <libxo/xo.h>
```

void

```
xo_syslog(int pri, const char *name, const char *fmt, ...);
```

void

```
xo_vsyslog(int pri, const char *name, const char *fmt, va_list vap);
```

void

```
xo_close_log(void);
```

void

```
xo_open_log(const char *ident, int logstat, int logfac);
```

int

```
xo_set_logmask(int pmask);
```

DESCRIPTION

The **xo_syslog**() function creates log entries following the standard defined in RFC5424. These messages are sent to the log syslogd(8) daemon, where they can be filtered, forwarded, and archived. **libxo** format strings are used to create both the message text and the **SD-PARAMS** content, containing name/value pairs that can be parsed by suitable automation software.

Refer to **xo_format**(5) for basic information about formatting strings. **xo_syslog** encodes all value fields at SD-PARAMS within the syslog message. An exception is made for fields with the "{d:}" modifier; such fields appear in the message text only, with fields with the "{e:}" modifier appear as SD-PARAMS, but not in the message text.

xo_vsyslog() accepts a *va_list* for additional flexibility.

xo_open_log(), **xo_close_log**(), and **xo_set_logmask**() are all analogous to their **libs** counterparts, **openlog**(3), **closelog**(3), and **setlogmask**(3). The extra underscores in the names are unfortunate, but

keep consistency in **libxo** function names.

EXAMPLES

```
xo_syslog(LOG_LOCAL4 | LOG_NOTICE, "ID47",
    "{e:iut/%u} An {:event-source} {:event-id/%u} log entry",
    iut, source, id);
```

SEE ALSO

xo_syslog(3), xo_set_syslog_enterprise_id(3), xo_format(5), libxo(3)

HISTORY

The **libxo** library first appeared in FreeBSD 11.0.

AUTHORS

libxo was written by Phil Shafer <phil@freebsd.org>.