Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'dlclose.3p' command

$ man dlclose.3p

DLCLOSE(3P) POSIX Programmer's Manual DLCLOSE(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

dlclose ? close a symbol table handle

SYNOPSIS

#include <dlfcn.h>

int dlclose(void *handle);

DESCRIPTION

The dlclose() function shall inform the system that the symbol table handle specified by handle is no longer needed by the application.

An application writer may use dlclose() to make a statement of intent on the part of the process, but this statement does not create any re? quirement upon the implementation. When the symbol table handle is closed, the implementation may unload the executable object files that were loaded by dlopen() when the symbol table handle was opened and those that were loaded by dlsym() when using the symbol table handle identified by handle.

Once a symbol table handle has been closed, an application should as? sume that any symbols (function identifiers and data object identi?

fiers) made visible using handle, are no longer available to the process.

Although a dlclose() operation is not required to remove any functions or data objects from the address space, neither is an implementation prohibited from doing so. The only restriction on such a removal is that no function nor data object shall be removed to which references have been relocated, until or unless all such references are removed. For instance, an executable object file that had been loaded with a dlopen() operation specifying the RTLD_GLOBAL flag might provide a tar? get for dynamic relocations performed in the processing of other relo? catable objects?in such environments, an application may assume that no relocation, once made, shall be undone or remade unless the executable object file containing the relocated object has itself been removed.

RETURN VALUE

If the referenced symbol table handle was successfully closed, dl? close() shall return 0. If handle does not refer to an open symbol ta? ble handle or if the symbol table handle could not be closed, dlclose() shall return a non-zero value. More detailed diagnostic information shall be available through dlerror().

ERRORS

No errors are defined.

The following sections are informative.

EXAMPLES

The following example illustrates use of dlopen() and dlclose():

```
#include <dlfcn.h>

int eret;

void *mylib;

...

/* Open a dynamic library and then close it ... */

mylib = dlopen("mylib.so", RTLD_LOCAL | RTLD_LAZY);

...

eret = dlclose(mylib);

...
```

APPLICATION USAGE

A conforming application should employ a symbol table handle returned from a dlopen() invocation only within a given scope bracketed by a dlopen() operation and the corresponding dlclose() operation. Implemen? tations are free to use reference counting or other techniques such that multiple calls to dlopen() referencing the same executable object file may return a pointer to the same data object as the symbol table handle.

Implementations are also free to re-use a handle. For these reasons, the value of a handle must be treated as an opaque data type by the ap? plication, used only in calls to dlsym() and dlclose().

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

dlerror(), dlopen(), dlsym()

The Base Definitions volume of POSIX.1?2017, <dlfcn.h>