All about LD_PRELOAD

Finnbarr P. Murphy

(fpm@fpmurphy.com)

Normally the Linux dynamic loader *ld-linux* (see *ld-linux(8)* man page) finds and loads the shared libraries needed by a program, prepare the program to run, and then run it. The shared libraries (shared objects) are loaded in whatever order the loader needs them in order to resolve symbols.

LD_PRELOAD is an optional environmental variable containing one or more paths to shared libraries, or shared objects, that the loader will load before any other shared library including the C runtime library (*libc.so*) This is called preloading a library.

Preloading a library means that its functions will be used before others of the same name in later libraries. This enables library functions to be intercepted and replaced (overwritten.) As a result program behavior can be non-invasively modified, i.e. a recompile is not necessary.

For example, you could write a library which implements alternative *malloc* and *free* functionality. By preloading the new library using *LD_PRELOAD* the new *malloc* and *free* functions will be used rather than the corresponding standard *libc* functions.

Shared library paths, if there is more than one, may be separated by either colons (preferred) or spaces. Entries in LD_PRELOAD containing '/'are treated as pathnames whereas entries not containing '/'are searched for as usual. Obviously this only affects dynamically linked – not statically linked – applications.

To avoid this mechanism being using as an attack vector for *suid/sgid* executable binaries, the loader ignores *LD_PRELOAD* if *ruid != euid*. For such binaries, only libraries in standard paths that are also *suid/sgid* will be preloaded.

Some users use *LD_PRELOAD* to specify libraries in nonstandard locations, but the *LD_LIBRARY_PATH* environmental variable is a better solution

Note that shared libraries specified in */etc/ld.so.preload* are loaded before libraries specified by *LD_PRELOAD*. The *libc* library checks for the existing of */etc/ld.so.preload* (see *elf/rtld.c*) and, if found, loads the listed shared libraries just as setting the environment variable *LD_PRELOAD* would do. The advantage of using */etc/ld.so.preload* is that these shared libraries are implicitly trusted and hence the *ruid != euid* test does not apply. Thus the loader will load the shared objects listed in */etc/ld.so.preload* even for *suid/sgid* executable binaries.

