

DCMTK - Bug #600

OFStandard::searchDirectoryRecursively() relies on undefined HAVE_READDIR_R

2014-06-26 14:04 - Jörg Riesmeier

Status:	Closed	Start date:	2014-06-26
Priority:	Normal	Due date:	
Assignee:		% Done:	100%
Category:	Library	Estimated time:	0:00 hour
Target version:		Compiler:	
Module:	ofstd, config?		
Operating System:	Linux		
Description OFStandard::searchDirectoryRecursively() checks HAVE_READDIR_R in order to determine whether to use readdir() or the thread-safe readdir_r(). However, HAVE_READDIR_R is never defined because it is neither checked by the GNU Autoconf nor the CMake configure mechanism. What is defined and actually checked by the module "dcmwlm" is HAVE_OLD_READDIR_R (in combination with _REENTRANT).			
Related issues: Related to DCMTK - Feature #161: Notizen zum Thema DCMTK und Multithreading Closed			

History

#1 - 2017-03-27 19:31 - Jan Schlamelcher

- Status changed from New to Closed

Closed by commit #a4a69b0167a08.

#2 - 2017-03-28 12:01 - Jan Schlamelcher

- % Done changed from 0 to 100

#3 - 2017-12-22 14:40 - Jörg Riesmeier

- Status changed from Closed to Reopened

- Target version changed from 3.6.2 to 3.6.6

- Operating System set to Linux

It seems that readdir_r() is deprecated and will be removed from glibc:

<https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=859202>
https://www.gnu.org/software/libc/manual/html_node/Reading_002fClosing-Directory.html
<https://lwn.net/Articles/696474/>

#4 - 2018-02-23 15:32 - Marco Eichelberg

Apparently, the glibc version in which readdir_r was first marked as deprecated is 2.24. In earlier versions of glibc and on other libc implementations we have to assume that readdir() is thread unsafe, because Posix does not require it to be thread safe.

The presence of glibc can be determined by the __GLIBC__ macro being present (after inclusion e.g. of <stdio.h> or <stdlib.h>). The version number of glibc is available at compile time with the __GLIBC__ and __GLIBC_MINOR__ macros, which resolve to the major and minor version number. The following pseudocode could be used to avoid the deprecation warning on glibc systems:

```
#if defined(__GLIBC__) && (((__GLIBC__ == 2) && (__GLIBC_MINOR__ >= 24)) || (__GLIBC__ >= 3))
#define READDIR_IS_THREADSAFE
#endif

#if defined(HAVE_READDIR_R) && !defined(READDIR_IS_THREADSAFE)
// use readdir_r()
#else
// use readdir()
#endif
```

#5 - 2018-02-23 16:07 - Marco Eichelberg

In a few DCMTK modules, `readdir()/readdir_r()` are still used directly. These should be migrated to `OFStandard::searchDirectoryRecursively()` first, so that the system dependent code is only needed in one place. Affected are:

- `private/dcmpps/libsrc/ppsmgram.cc`
- `public/dcmpstat/apps/dcmprscu.cc`
- `public/dcmwlm/libsrc/wlfsim.cc`

#6 - 2018-03-09 12:57 - Michael Onken

This bug is also listed in [Debian as bug #859202](#).

#7 - 2018-03-09 13:49 - Jörg Riesmeier

Was already mentioned (see [Note #3](#)).

#8 - 2018-03-11 14:49 - Marco Eichelberg

Fixed by commit #f421bce for public modules; commit #03a4f49 private modules.

#9 - 2018-03-27 12:49 - Marco Eichelberg

- *Status changed from Reopened to Closed*

#10 - 2020-05-25 13:28 - Michael Onken

- *Target version deleted (3.6.6)*