

DCMTK - Bug #1006

defragmentTCP() does not handle WSAEWOULDBLOCK error on Windows when no data ready from non-blocking socket for read()

2021-08-30 18:34 - Jörg Riesmeier

Status:	Closed	Start date:	2021-08-30
Priority:	Normal	Due date:	
Assignee:	Marco Eichelberg	% Done:	100%
Category:	Library	Estimated time:	2:00 hours
Target version:	3.6.7		
Module:	dcmnet	Compiler:	
Operating System:	Windows		

Description

The following [forum posting](#) from "wrenashe" reports a possible bug and corresponding solution:

We (Bonnie found the issue) have a DICOM message relay on Windows to test, storescu --> our DICOM router(SCP --> SCU, using DCMTK3.6.6) --> external SCP,

We got the error shown below on Windows, We do not have the same problem on Linux with our DICOM router built from the same DCMTK.

```
C:\testbin>storescu -aee-xxxPACS4 -aet CL_SCP LOCALHOST 11110 C:\testbin\data\*
```

```
E: Store Failed, file: C:\testbin\data\CT.1.3.6.1.4.1.-----1-----12.0:
```

```
E: 0006:020e DIMSE Failed to send message
```

```
E: 0006:031d TCP I/O Error (An established connection was aborted by the software in your host machine.) occurred in routine: writeDataPDU
```

```
E: Store SCU Failed: 0006:020e DIMSE Failed to send message
```

```
E: 0006:031d TCP I/O Error (An established connection was aborted by the software in your host machine.) occurred in routine: writeDataPDU
```

```
E: Association Abort Failed: 0006:031d TCP I/O Error (An established connection was aborted by the software in your host machine.) occurred in routine: sendAbortTCP
```

The investigation shows the DICOM router SCP side got an issue about "DIMSE Read PDV failed". In defragmentTCP(), read() in non-blocking mode fails when the socket is not ready to be read. The code is not aware of the situation like bytesRead = -1 and WSAEWOULDBLOCK as > the error, just returns DUL_NETWORKCLOSED and results in the failure.

The modified code diff (also from Bonnie here) fixes this problem on Windows against our testing scenario, just for your reference here, hope it helps.

```
3699a3700,3704
```

```
> #ifdef HAVE_WINSOCK_H
>         if (OFStandard::getLastNetworkErrorCode().value() == WSAEWOULDBLOCK) {
>             Sleep(1);
>         }
> #endif
```

```
3713c3718,3722
```

```
<         } while (bytesRead == -1 && OFStandard::getLastNetworkErrorCode().value() == DCMNET_EINTR);
---
```

```
>         } while ((bytesRead == -1 && OFStandard::getLastNetworkErrorCode().value() == DCMNET_EINTR));
---
```

```
> #ifdef HAVE_WINSOCK_H
```

```
>         || (bytesRead == -1 && (OFStandard::getLastNetworkErrorCode().value() == WSAEWOULDBLOCK))
> #endif
```

```
>         );
```

```
3723a3733,3743
```

```
> #ifdef HAVE_WINSOCK_H
>         DWORD errorCode = GetLastError();
>         LPVOID lpMsgBuf;
>         FormatMessage(
```

```
>         FORMAT_MESSAGE_ALLOCATE_BUFFER | FORMAT_MESSAGE_FROM_SYSTEM,  
>         NULL, errorCode,  
>         MAKELANGID(LANG_NEUTRAL, SUBLANG_DEFAULT),  
>         (LPTSTR)&lpMsgBuf, 0, NULL);  
>     printf("defragmentTCP(): bytesRead=%d errorCode=%d %s\n", bytesRead, errorCode, (LPT  
STR) lpMsgBuf);  
>     LocalFree(lpMsgBuf);  
> #endif
```

History

#1 - 2021-08-30 19:13 - Jörg Riesmeier

- Description updated

#2 - 2022-01-04 16:06 - Marco Eichelberg

- Status changed from New to Closed

- Assignee set to Marco Eichelberg

- % Done changed from 0 to 100

- Estimated time set to 2:00 h

Closed by commit #038487699.