

Re: SetSockOpt with SO_REUSEADDR parameter

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2007-03/msg01890.html>

- *From:* Joseph M. Newcomer <newcomer@xxxxxxxxxxxxx>
 - *Date:* Fri, 23 Mar 2007 01:00:31 -0500
-

While UDP is probably the correct choice, his error is that he is using a TCP article to build his server, and missing most of the key points of how TCP and UDP differ.

joe

On Thu, 22 Mar 2007 05:36:28 -0500, "Mubashir Khan" <mubi@xxxxxxxxxx> wrote:

Yes udp is the correct option for this scenerio for both realtime audio and video. We once worried about the data reliability and we used tcp for VOIP communication in one of our systems the voice data was too much and too rapid for the tcp buffer at target pc and we were having all sorts of problems. Later on we switched to UDP. In heavy traffic time, few packets will be missed and voice at the other end may cut for few moments, in telephony language we called it distortion in line... other wise every thing went fine

interesting article

<http://msdn2.microsoft.com/en-us/library/ms740621.aspx>

It says you may be using Bind after setting socket option.

"mmlab_js" <mmlabjs@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:0F1E014C-41F4-4498-B008-CA1E951B610D@xxxxxxxxxxxxxxxxxxxxx

This concept does not exist. So whatever you are trying to do, it won't work. There is no concept that allows you to create multiple sockets with the same port number. What is happening is that you are throwing away the old socket and replacing it with the new socket.

I write a server application to send real-time video frame to client(s).

Re: SetSockOpt with SO_REUSEADDR parameter

When a client logins into server, I will create a UDP socket to send real-time video frame to it.

So, I think I will use multiple sockets with the same port to send frame.

Is this concept wrong?

NEVER, EVER, UNDER ANY CIRCUMSTANCES
IMAGINABLE, EVER WRITE SOMETHING LIKE
THIS!

THIS IS ALWAYS AN ERROR!!!

It make no sense at all to compare a boolean value to TRUE
or FALSE; it is

as pointless as writing

if(a > 0) == TRUE)

instead of writing

if(a > 0)

since a boolean value always stands for, well, a boolean
value. Comparing

a boolean value

to TRUE is ALWAYS an error, since ANY non-zero value
is non-FALSE, but you

have

erroneously insisted that it be equal to the literal TRUE!

LOSE THIS BAD PROGRAMMING STYLE
IMMEDIATELY!!!!

Thanks for your suggestions. I will remember and change my bad
programming
style.

I'm not sure what you are doing here or why you are needing
to reuse the
address for
multiple connections, because UDP is a connectionless
protocol.

The reason I use mutiple UDP is the beginning of this response.

Because I will send real-time video frame, I use UDP socket.

I will re-describe my idea. The basic architecture is based on the source
[\[http://support.microsoft.com/kb/192570\]](http://support.microsoft.com/kb/192570)

1) Server listen on port 9898

2) Client spawns a thread, which connects to the server

3) Server accepts the connection and spawns a thread to handle the socket
communication and send real-time frame

4) The thread creates a UDP socket with port 9899

5) The thread sends the real-time frame with UDP socket to client.

So, I think I need one UDP socket for one client.

Re: SetSockOpt with SO_REUSEADDR parameter

After I change the following code:

```
=====
void CUdpSendSocket::CreateSocket(UINT nPort)
{
    CString szError;

    BOOL bRet = Create(nPort, SOCK_DGRAM, FD_WRITE);

    if (bRet > 0) {
        szError.Format(_T("Send Socket Create() failed: %d"),
            ::GetLastError());
        AfxMessageBox(szError);
        return;
    }
    m_nPortNum = nPort;

    BOOL reuse = TRUE;
    if (!SetSockOpt(SO_REUSEADDR, &reuse, sizeof(int))) {
        szError.Format(_T("SetSockOpt failed to set SO_REUSEADDR:%d"),
            ::GetLastError());
        AfxMessageBox (szError);
    }
}
=====
```

It still can't work for create UDP sockets with the same port. What shall I do?

Joseph M. Newcomer [MVP]
email: newcomer@xxxxxxxxxxxxx
Web: <http://www.flounder.com>
MVP Tips: http://www.flounder.com/mvp_tips.htm