

RE: call is blocked in recvfrom() and no further proceedings in Win CE

Source:

<http://www.tech-archive.net/Archive/WindowsCE/microsoft.public.windowsce.embedded/2005-11/msg00124.html>

- *From:* "Sudha.v" <Sudhav@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 16 Nov 2005 23:08:05 -0800
-

Thanks for your reply. Here I'm simplifying my problem. Just have a look and let me know if you can find something on it.

I have to create an application which will have two sockets. These two sockets communicates with server through two different ports. For sending request in one port(for ex 5070). and for receiving in another port(5090).

Actually in windows I'm able to do it.

In windows CE, I'm able to send a request but I'm unable to receive it.

Scenario in windows

1. Create socket. (for transmission)
2. Create another socket & bind with server IP address(For reception).

The above steps are fine and application works properly.

Scenario in windows CE.

1. Create socket. (for transmission) – Success
2. Create another socket & bind with server IP address(For reception). – failure.

what I guess is, If I'm able to bind with server IP address, I can achieve the desired result.

So any idea of how to fix this issue with TCP/UDP?

Any help would be greatly appreciated.

Note : In my application, server is a NT service and it will receive request and send response. So, If TCP is used, connect() will be succeeded. But application will be blocked in accept(). Because there is no socket to connect from other end.

And without accept I can't receive the data.

So, UDP would be preferable. Right now I'll check this will emulator. later i'll use GPRS for communication. I wouldn't use ActiveSync.

Sudha

—

Attempting half-heartedly is like expecting failure and achieving it.

"Sudha.v" wrote:

RE: call is blocked in recvfrom() and no further proceedings in Win CE

> In the below code, application simply keeps on waiting for incoming
> data(Means, call is blocked in recvfrom() and no further proceedings).
> Environment : EVC++ 4.0, Smart phone 2003 SDK Emulator.
> Simple intro about the application.
> An NT service is running in my system(PC) which is nothing but the server.
> My program has to send request to service through port 5070(in this port only
> server is receiving the requests). Depending upon request, server will send
> response back to the program in port 5090. and application uses UDP.
> · So In the code, socket s will send request to NT service.
> · And sock will receive the response.
> · Everything seems to be OK. But during debugging the application infinitely
> waiting in recvfrom() call. This call is not at all returning.
> But the same code is working in windows. Only in Windows CE it is not working.
> In order to find whether data is available in the port or not, I have done
> the following.
> · Sent data from Windows CE(i.e., socket is created in EVC++ 4.0 to send
> request to NT service).
> · Received the response in Windows(means, for reception I created socket in
> VC++ 6.0 and data is successfully received).
> · From the above two steps, I m clear that, data is available in the
> specific port (i.e, in 5090).
> So, the issue is in Windows CE, recvfrom() call is infinitely waiting and
> not returning even though the packet is available in the specific net work
> address.
> You may be annoyed of seeing this bulk of code. But I m desperately in need
> of the solution. So, please have a look at the code. Your help is greatly
> appreciated.

```
>
> #include <stdio.h>
> #include <stdlib.h>
> #include <malloc.h>
>
> #include <windows.h>
> #include <windowsx.h>
> #include <winsock2.h>
> #include <ws2tcpip.h>
> #include <string.h>
> #define DEFAULT_FAMILY AF_UNSPEC
> #define DEFAULT_SOCKETTYPE SOCK_DGRAM
> #define DEFAULT_PORT "5090"
> #define BUFFER_SIZE 23 // length of "WinCE Echo
> Test Packet"
> #define TIMEOUT_SECS 2
> #define TIMEOUT_USECS 0
> SOCKET sock = INVALID_SOCKET;
> int nFamily = DEFAULT_FAMILY;
> int nSockType = DEFAULT_SOCKETTYPE;
> char szRemoteName[64] = "INFBA01803";
> char *szPort = DEFAULT_PORT;
> SOCKADDR_STORAGE ssRemoteAddr;
> int cbXfer, cbTotalRecvd, cbRemoteAddrSize;
```

RE: call is blocked in recvfrom() and no further proceedings in Win CE

RE: call is blocked in recvfrom() and no further proceedings in Win CE

```
> WSADATA wsaData;
> ADDRINFO Hints, *AddrInfo = NULL, *AI;
> char szRemoteAddrString[128];
> fd_set fdReadSet;
> TIMEVAL timeout = {TIMEOUT_SECS, TIMEOUT_USECS};
> char pRecvBuf[BUFFER_SIZE];
> struct ip_mreq mreq;
> SOCKET sock, SockServ[FD_SETSIZE];
> fd_set fdSockSet;
> int nNumSocks, nNumSocksCount;
> char pBuf[] = "WinCE Echo Test Packet";
> int iReceiveStatus;
>
> //Change these if required
> #define USERNAME "sudhagar" //TODO
> #define PWD "nopass" //TODO
> #define INV_TO "sandeep" //TODO
> #define IP_SERVER "172.29.12.154"//TODO
> #define PORT_SERVER "5070" //TODO
> TIMEVAL ReceiveTimeout;
> int iRC = 0;
> int iSendStatus = 0;
> int SIPlib_main()
> {
> int i=0,j=0,k,infop1;
> char *reg_dest,*inv_dest,*bhr= NULL ,*nnop1;
> int x=0,reg_id=-1;
> char tmp[200],locip[50];
> unsigned int addr2=0;
> osip_proxy_authorization_t *proxy_auth;
> int cbXfer, cbTotalRecvd, cbRemoteAddrSize;
> TCHAR *str;
> SOCKET s,s1;
> struct sockaddr_in addr,addr1;
> char packetbuffer[1200]={{'\0'}},packetbuffer1[1000]={{'\0'}};
> WSADATA dat ;
>
> ///////////////
> SOCKET SIPRTPSOC;
> struct sockaddr_in addr21;
> char nextnon1[50];
> char *buf2;
> buf2=(char*)malloc(8*sizeof(char));
> ///////////////
> //End of variable declarations
>
> if(WSAStartup(MAKEWORD(2,2), &wsaData))
> {
> return 1;
> }
> sock = INVALID_SOCKET;
```

RE: call is blocked in recvfrom() and no further proceedings in Win CE

RE: call is blocked in recvfrom() and no further proceedings in Win CE

```
> for(i = 0; i < FD_SETSIZE; i++)
> SockServ[i] = INVALID_SOCKET;
>
> // Get a list of available addresses to serve on
>
> memset(&Hints, 0, sizeof(Hints));
> Hints.ai_family = nFamily;
> Hints.ai_socktype = nSockType;
> Hints.ai_flags = AI_NUMERICHOST | AI_PASSIVE;
> if(getaddrinfo(NULL, szPort, &Hints, &AddrInfo))
> {
> OutputDebugString(TEXT("ERROR: Couldn't get resolve the server
> name/address!"));
> }
> nNumSocksCount = 0;
> for(AI = AddrInfo; AI != NULL; AI = AI->ai_next)
> {
> if (nNumSocksCount == FD_SETSIZE)
> {
> // getaddrinfo returned more addresses than we could use
> break;
> }
>
> if((AI->ai_family == PF_INET) || (AI->ai_family == PF_INET6)) //
> only want PF_INET or PF_INET6
> {
> SockServ[nNumSocksCount] = socket(AI->ai_family,
> AI->ai_socktype, AI->ai_protocol);
> if (SockServ[nNumSocksCount] != INVALID_SOCKET)
> {
> if (bind(SockServ[nNumSocksCount], AI->ai_addr,
> AI->ai_addrlen) == SOCKET_ERROR)
> closesocket(SockServ[nNumSocksCount]);
> else
> {
> OutputDebugString(
> TEXT("Socket 0x%08x ready for connection with %hs
> family, %hs type, on port %hs\r\n"),
> SockServ[nNumSocksCount],
> (AI->ai_family == AF_INET) ? "AF_INET" :
> ((AI->ai_family == AF_INET6) ? "AF_INET6" : "UNKNOWN"),
> (AI->ai_socktype == SOCK_STREAM) ? "TCP" :
> ((AI->ai_socktype == SOCK_DGRAM) ? "UDP" : "UNKNOWN"),
> szPort);
> nNumSocksCount++;
> }
> }
> }
> }
> }
> }
> }
> freeaddrinfo(AddrInfo);
```

RE: call is blocked in recvfrom() and no further proceedings in Win CE

RE: call is blocked in recvfrom() and no further proceedings in Win CE

```
>
> if (nNumSocksCount == 0)
> {
> OutputDebugString(TEXT("ERROR: Unable to serve on any address. Error
> = %d\r\n"), WSAGetLastError());
> goto Cleanup;
> }
> //Initializing socket s for transmission
> s = socket(PF_INET,SOCK_DGRAM,IPPROTO_UDP);
> memset(&addr,0,sizeof(struct sockaddr_in));
> addr.sin_family = PF_INET;
> addr.sin_addr.s_addr = inet_addr(IP_SERVER);//Server IP
> addr.sin_port = htons((short)atoi(PORT_SERVER));//(5 0 7 0); Server Listen
> Port
> //1st Transmission ( REGISTER )
> i=sendto(s,reg_dest,strlen(reg_dest),0,(struct sockaddr
> *)&addr,sizeof(struct sockaddr_in));
> NKDbgPrintfW(L"\nSent : %d bytes\n\n",i);
> if(i != strlen(reg_dest))
> {
> OutputDebugString(TEXT("ERROR: Couldn't send the data! error =
> %d\r\n"), WSAGetLastError());
> }
> // Wait for incoming data/connections
> nNumSocks = nNumSocksCount;
> FD_ZERO(&fdSockSet);
> for (i = 0; i < nNumSocks; i++) // want to check all available sockets
> FD_SET(SockServ[i], &fdSockSet);
> if (select(nNumSocks, &fdSockSet, 0, 0, NULL) == SOCKET_ERROR)
> {
> OutputDebugString(TEXT("ERROR: select() failed with error =
> %d\r\n"), WSAGetLastError());
> goto Cleanup;
> }
> for (i = 0; i < nNumSocks; i++) // check which socket is ready to process
> {
> if (FD_ISSET(SockServ[i], &fdSockSet)) // proceed for connected
> socket
> {
> FD_CLR(SockServ[i], &fdSockSet);
> if(nSockType == SOCK_STREAM)
> {
> cbRemoteAddrSize = sizeof(ssRemoteAddr);
> sock = accept(SockServ[i], (SOCKADDR*)&ssRemoteAddr,
> &cbRemoteAddrSize);
> if(sock == INVALID_SOCKET)
> {
> OutputDebugString(TEXT("ERROR: accept() failed with
> error = %d\r\n"), WSAGetLastError());
> goto Cleanup;
> }
> }
```

RE: call is blocked in recvfrom() and no further proceedings in Win CE

RE: call is blocked in recvfrom() and no further proceedings in Win CE

```
> OutputDebugString(TEXT("Accepted TCP connection from socket
> 0x%08x\r\n"), sock);
> }
> else
> {
> sock = SockServ[i];
> OutputDebugString(TEXT("UDP data available on socket
> 0x%08x\r\n"), sock);
> }
> break; // Only need one socket
> }
> }
> // Receive data from a client
>
> cbTotalRecvd = 0;
> do
> {
> cbRemoteAddrSize = sizeof(ssRemoteAddr);
> cbXfer = recvfrom(sock, packetbuffer + cbTotalRecvd,
> sizeof(packetbuffer) - cbTotalRecvd, 0,
> (SOCKADDR *)&ssRemoteAddr, &cbRemoteAddrSize);
> cbTotalRecvd += cbXfer;
> } while(cbXfer > 0 && cbTotalRecvd < sizeof(packetbuffer));
>
> OutputDebugString(TEXT("SUCCESS - Received %d bytes from client
> %hs\r\n"), cbTotalRecvd, szRemoteAddrString);
> Cleanup:
> for(i = 0; i < nNumSocks && SockServ[i] != INVALID_SOCKET; i++)
> closesocket(SockServ[i]);
> if(sock != INVALID_SOCKET)
> {
> shutdown(sock, SD_BOTH);
> closesocket(sock);
> }
> WSACleanup();
> return 0;
>
> --
> Attempting half-heartedly is like expecting failure and achieving it.
```

• *Follow-Ups:*

◆ *Re: call is blocked in recvfrom() and no further proceedings in Win CE*

◇ *From:* Paul G. Tobey [eMVP]

• *References:*

◆ *call is blocked in recvfrom() and no further proceedings in Win CE*

◇ *From:* Sudha.v

RE: call is blocked in recvfrom() and no further proceedings in Win CE

- Prev by Date: ***Re: NIC and NDIS problem in Windows Mobile 5.0***
- Next by Date: ***Re: What is the requirement to buy WinCE.Net Runtime ?***
- Previous by thread: ***Re: call is blocked in recvfrom() and no further proceedings in Wi***
- Next by thread: ***Re: call is blocked in recvfrom() and no further proceedings in Win CE***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***