
Subject: Missing data in the socket

Posted by [sinpeople](#) on Sun, 24 Jan 2021 14:04:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi folks,

I am re-writing a program C, which listens to a socket port to wait for the instructions to come from some other program B which I don't have access to its source code.

Fortunately, I have the source code of program C. Though it is very primitive, but its socket communication worked.

I tried to use Urr in the package to simplifier the program C. The program is attached below. It did receive something, but it has far less words than I have expected. Am I using the split function correctly or Is there anything wrong with the way I am using the Urr? It seems several bytes (not sure how long) is missing in the code below. It should be 23 tokens, but this program only get 13 after split function

Though it looks a little bit strange, the original program C can receive the message correctly. I have attached this portion of the code below. Though I have its source code, it cannot be compiled in VC++ environment any more. Otherwise I would have stepped into it.

The source code of the legacy socket handling is also attached

Thank you very much for your suggestions for me to trouble shoot this problem.

Best Regards

David

```
// New Code, which gets less words from the socket
UrrServer urr;
urr.Create(local_cfg.nUdpSigStatusPort); //7593<==
//Cout() << "URR Ping server\n";
//while(false)
for(;)
{
    UrrRequest r;
    if(urr.Accept(r))
    {
        String signalStatus = ~r;
        Vector<String> words = Split(signalStatus, [](int c) { return c == ' ' || c == ',' || c == '\n' ? 1 : 0; });

        SignalStatusMessage sigStatus;
        int nCmdID = atoi(~(words[0]));
        int No_SG = atoi(~(words[4]));
        int nTime = atoi(~(words[2]));

        if(nCmdID == 30)
```

```

{
for(int i=0; i<No_SG; i++)
{
    SignalStatus sig_temp;

    sig_temp.sg_ID = atoi(~(words[3*i +5]));
    sig_temp.sg_Color = atoi(~(words[3*i +6]));
    sig_temp.sg_Elapsed = atoi(~(words[3*i +7]));

    sigStatus.status.push_back(sig_temp);
}
sigStatus.nTime = nTime;
}

}

//Sleep(200);
}

//Original code to receive the string and split tokens
while (true) {

try {
    memset(inSignalStatusBuf, 0, 300);
    recvMsgSize = signalStatusReceiveSocket.recvFrom(inSignalStatusBuf, 300, sourceAddress,
sourcePort);

    if (recvMsgSize > 0)
    {
        char*p;
        vector<string> tokens;
        double time = 0.0;

        p = strtok(inSignalStatusBuf, MSG_SEPARATOR);
        while (p != NULL)
        {
            tokens.push_back(string(p));
            p = strtok(NULL, MSG_SEPARATOR);
        }

        if (atoi(tokens[0].c_str()) == 30) {
            int No_SG = atoi(tokens[4].c_str());
            time = atof(tokens[2].c_str());
            signalStatus.clear();
        }
    }
}

```

```
for (size_t i = 0; i < No_SG; i++)  
{  
    SignalStatus ss_temp;  
    ss_temp.sg_ID = atoi(tokens[3 * i + 5].c_str());  
    ss_temp.sg_Color = atoi(tokens[3 * i + 6].c_str());  
    ss_temp.sg_Elapsed = atoi(tokens[3 * i + 7].c_str());  
    signalStatus.push_back(ss_temp);  
}  
cout << "New Signal Status Received at time " << time << endl;  
}  
}  
}  
}  
catch (SocketException &e) {  
}
```

File Attachments

- 1) [PracticalSocket.cpp](#), downloaded 229 times
- 2) [PracticalSocket.h](#), downloaded 214 times