

PROGRAM SOCKET UNTUK MENGIRIM FILE DENGAN VISUAL BASIC PADA SISTEM OPERASI WINDOWS

Hersatoto Listiyono

Fakultas Teknologi Informasi, Universitas Stikubank Semarang

Abstrak :

Terjadinya komunikasi antara dua komputer atau lebih memerlukan aturan dan format yang sama agar komunikasi dapat dimengerti. Aturan ini yang disebut protokol. Penggunaan socket programming memungkinkan adanya komunikasi antara client dan server. Salah satu contoh sederhana penggunaan socket programming adalah pembuatan program untuk chatting. Program tersebut sebenarnya merupakan bentuk aplikasi berupa komunikasi antara client dan server. Ketika seorang user (client) melakukan koneksi ke chat server, program akan membuka koneksi ke port yang diberikan, sehingga server perlu membuka socket pada port tersebut dan “mendengarkan” koneksi yang datang. Socket sendiri merupakan gabungan antara host-address dan port address. Dalam hal ini socket digunakan untuk komunikasi antara client dan server

Kata kunci : *protokol, socket, port*

PENDAHULUAN

Socket adalah interface pada jaringan yang menjadi titik komunikasi antarmesin pada Internet Protocol, dan tentunya tanpa komunikasi ini, tidak akan ada pertukaran data dan informasi jaringan. Socket terdiri dari elemen-elemen utama sebagai berikut: Protokol, Local IP, Local Port, Remote IP, Remote Port.

Sedangkan pemrograman socket adalah bagaimana memberi instruksi kepada Operating System untuk dapat menggunakan komponen/API (Application Programming Interface) socket dan kemudian menggunakannya untuk berkomunikasi antar komputer.

Protokol

Dalam komunikasi antara beberapa pihak, tentunya harus digunakan kesepakatan aturan dan format yang sama agar komunikasi dapat dimengerti. Seperti halnya dua orang yang menggunakan bahasa yang sama, maka bahasa di sini berfungsi sebagai protokol. Protokol yang digunakan dalam socket dapat menggunakan TCP ataupun UDP.

Aplikasi socket umumnya terdiri dari dua kategori berdasarkan pengiriman datanya yaitu: Datagram socket (menggunakan UDP) yaitu stream socket (menggunakan TCP). Walaupun sama-sama berfungsi sebagai protokol pertukaran data, terdapat perlakuan yang berbeda antara UDP dan TCP.

UDP tidak memerlukan proses koneksi terlebih dahulu untuk dapat mengirimkan data, paket-paket data yang dikirimkan UDP bisa

jadi melalui rute yang berbeda-beda, sehingga hasil yang diterima bisa jadi tidak berurutan. Contohnya jika aplikasi socket pengirim mengirimkan berturut-turut pesan 1, pesan 2, dan pesan 3, maka aplikasi socket penerima belum tentu mendapatkan pesan yang berurutan dimulai dari pesan 1, pesan 2, dan terakhir pesan 3. Bisa saja pesan 2 terlebih dulu diterima, menyusul pesan-pesan yang lain, atau berbagai kemungkinan lainnya. Bahkan, dapat terjadi pesan yang dikirimkan tidak sampai ke penerima karena kegagalan pengiriman paket data.

Tidak demikian halnya dengan stream socket yang menggunakan TCP. Jenis ini mengharuskan terjadinya koneksi terlebih dahulu, kemudian mengirimkan paket-paket data secara berurutan, penerima juga dijamin akan menerima data dengan urutan yang benar, dimulai dari data pertama yang dikirimkan hingga data terakhir. TCP dapat menangani data yang hilang, rusak, terpecah, ataupun terduplikasi.

Dari sekilas perbedaan ini, kita dapat menarik kesimpulan bahwa aplikasi socket yang menggunakan TCP memerlukan pertukaran data dua arah yang valid. Sedangkan, aplikasi socket yang menggunakan UDP lebih memprioritaskan pada pengumpulan data.

Karena itu aplikasi socket dengan TCP sering diterapkan untuk aplikasi chat, transfer file, ataupun transaksi-transaksi penting. Sedangkan aplikasi socket dengan UDP cocok diterapkan untuk aplikasi monitoring jaringan, game online, dan aplikasi-aplikasi broadcast.

IP address

IP Address adalah sebuah alamat virtual pada setiap computer untuk mengirimkan data , contoh dalam kehidupan nyata adalah setiap kita mengirim surat kita harus menyertakan alamat pengirim dan alamat tujuan , jika tidak ada maka paket tidak bisa dikirim karena tidak jelas alamat pengirim dan alamat penerima. Contoh IP Address adalah 192.168.10.121

Port

Salah satu elemen penting yang digunakan dalam aplikasi socket adalah port. Port merupakan sebuah koneksi data virtual yang digunakan aplikasi untuk bertukar data secara langsung. Terdapat banyak port di dalam sebuah sistem komputer dengan fungsinya masing-masing. Sebagai contoh, dalam mengirim e-mail digunakan service SMTP yang umumnya menggunakan port 25. Sementara service POP3 untuk menerima e-mail menggunakan port 110, port 80 digunakan untuk HTTP, port 443 digunakan untuk HTTPS, dan seterusnya. Nomor-nomor port dikategorikan dalam tiga jenis sebagai berikut: *well-known* ports, yaitu merupakan port yang telah digunakan secara internal oleh sistem Windows, misalnya port untuk koneksi Internet, service FTP, dan seterusnya. Port yang telah digunakan ini adalah port 0 sampai dengan port 1023. Yang kedua adalah registered ports, yaitu port yang dapat digunakan dalam aplikasi yang kita buat, range-nya adalah port 1024 hingga port 49151. Ketiga adalah dynamic/private ports yaitu port 49152 sampai dengan port 65535.

Winsock

Untuk pemrograman aplikasi socket berbasis Windows, maka komponen API yang sering digunakan adalah Winsock (Windows Socket API) yang mendukung interface standar TCP/IP, yang merupakan protokol jaringan paling populer saat ini.

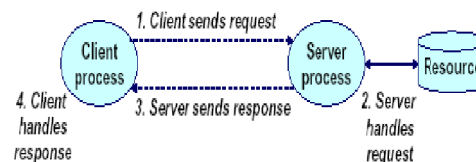
Jadi windows socket adalah perpaduan IP Address dan port number , contoh bila kita mengakses <http://www.google.com> , kita misalkan google.com memiliki IP Address 66.25.12.25 , sama saja anda mengakses dengan <http://www.google.com:80> atau <http://66.25.12.25:80> atau untuk layanan chatting seperti MIRC menggunakan port :6666 jadi cara kerja untuk mengakses MIRC adalah alamatserver:portnumber contoh kita anggap IP Address untuk alamatserver chatting

misalkan DALNET adalah: 121.122.204.12 maka prosesnya adalah 121.122.204.12:6666

Diperlukan jaringan dengan minimal dua komputer yang saling terkoneksi untuk mencoba pemrograman socket. Tapi dapat juga menggunakan satu komputer untuk percobaan, yaitu dengan memanfaatkan alamat localhost(127.0.0.1) yang mengizinkan dua aplikasi berjalan pada satu mesin komputer dan berkomunikasi satu sama lain.

Client-server

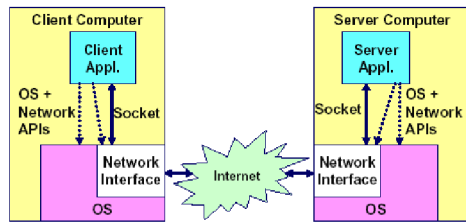
Setiap aplikasi di jaringan, transaksinya didasarkan pada konsep client-server. Sebuah server dan sebuah client atau beberapa client yang meminta/request pelayanan ke server. Fungsi server sebagai pengatur resource yang ada, yang menyediakan pelayanan dengan memanfaatkan resource yang untuk kebutuhan client. Proses ini (proses client-server) dijalankan pada satu komputer berfungsi sebagai server dan sebuah atau beberapa komputer berfungsi sebagai client. Hubungan tersebut dapat digambarkan sebagai berikut :



gambar 1 Transaksi Client – server

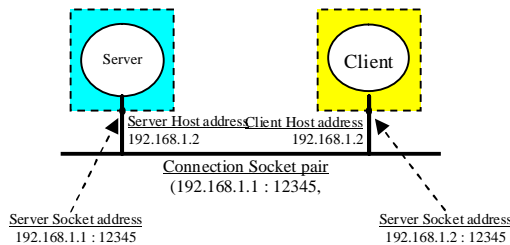
Aplikasi client-server menggunakan protokol transport untuk saling berinteraksi. Ketika proses interaksi terjadi, suatu aplikasi harus memberikan informasi secara detail tentang : informasi tentang apakah dia client atau server, pengirim memberikan informasi tentang data yang dikirim, penerima memberikan informasi tentang dimana data diletakkan, dll.

Antarmuka (interface) antara program aplikasi dengan protokol komunikasi pada suatu sistem operasi disebut Application Program Interface (API). API didefinisikan sebagai suatu kumpulan instruksi yang mendukung proses interaksi antara suatu perangkat lunak dengan suatu protokol yang digunakan.



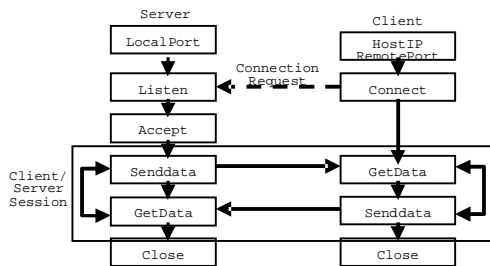
gambar 2 Aplikasi Jaringan

Pada saat suatu aplikasi berkomunikasi, awalnya aplikasi membuat socket baru, maka pada aplikasi tersebut akan diberikan nomer yang digunakan sebagai referensi socket. Jika ada suatu sistem yang menggunakan nomer referensi socket tersebut, maka akan terjalin suatu jaringan komunikasi antar komputer sebaik transfer data lokal.



gambar 3 komunikasi antar komputer

Untuk berkomunikasi antara server dan client harus tahu alamat kelas IP server apakah A, B atau C, seperti contoh di atas dengan menggunakan kelas C, yaitu 192.168.1.1 untuk server dan 192.168.1.2 untuk client, begitu juga nomor port server yang dituju haruslah sama. Disini memakai port 12345.



gambar 4 Ilustrasi Interface socket

Secara garis besar langkah – langkah yang dilakukan pada client dan server adalah sebagai berikut :

1. Langkah – langkah dasar di server :
 - a. Membuat socket dengan perintah Winsock.Localport
 - b. Menyiapkan socket untuk menerima koneksi yang masuk dengan perintah Winsock.Listen

- c. Menerima koneksi yang masuk ke server dengan perintah Winsock.ConnectionRequest
 - d. Memastikan permintaan koneksi diterima dengan Winsock.Accept requestID
 - e. Melakukan komunikasi penerimaan data dengan perintah Winsock.GetData
 - f. Melakukan komunikasi pengiriman data dengan Winsock.SendData
2. Langkah – langkah dasar di client :
 - a. Meminta koneksi client ke server, yang di dalamnya adalah :
 - Melakukan pengalamatan ke server dengan perintah Winsock.RemoteHost atau Winsock.HostIp.
 - Melakukan koneksi port dengan perintah Winsock.RemotePort.
 - Menghubungi server dengan Winsock.connect
 - b. Melakukan komunikasi penerimaan data dengan perintah Winsock.GetData
 - c. Melakukan komunikasi pengiriman data dengan Winsock.SendData

Fungsi-fungsi pemrograman jaringan dalam visual basic antara lain :

1. Winsock.Listen, fungsi ini membuat winsock berstatus menunggu status dari local port.
2. Winsock.LocalPort, fungsi ini menggunakan lokal port pada komputer yang bersangkutan untuk jalur komunikasi data, termasuk permintaan sambungan/ koneksi.
3. Winsock.Connect [RemoteHost/HostIP],[RemotePort], fungsi ini digunakan untuk koneksi melalui HostIP (penggunaan IP Address) sebagai koneksi ke Host, beserta remoteport computer yang bersangkutan, dimana komputer server dalam model status listen atau siap untuk dikoneksi.
4. Winsock.Accept [requestID],fungsi ini digunakan untuk menerima koneksi dari client dalam fungsi Winsock_ConnectionRequest. Dan perintah berikutnya permintaan diterima harus dilaksanakan dengan perintah Winsock.Accept requestID.
5. Winsock.SendData [Data],fungsi ini digunakan untuk mengirim. Data yang dikirim dapat berbentuk string atau byte array, tapi disarankan dalam bentuk string.
6. Winsock.GetData [Data],[Type],[MaxLen], fungsi ini digunakan untuk menerima

kiriman data dari Winsock control, dimana Data dapat berbentuk variable, and Type = vbString,vbInteger etc. MaxLen bersifat opsional.

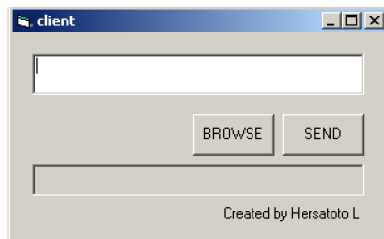
7. Winsock.RemoteHostIP [IPAddress], fungsi ini digunakan untuk cara koneksi ke server dengan menggunakan metode IP address.

Penjelasan Program :

Untuk Client :

Objects yang diperlukan : MSWINSCK.OCX, comdlg32.OCX

Controls: wsTCP (Winsock), cmdSend (CommandButton), dlg (CommonDialog), cmdBrowse (CommandButton), txtFile (TextBox), lblStatus (Label)



gambar 5 Aplikasi client

Source Code untuk Client:

```
Dim buffer() As Byte
Dim lBytes As Long
Dim temp As String

Private Sub cmdBrowse_Click()
    dlg.ShowOpen
    txtfile = dlg.FileName
End Sub

Private Sub cmdSend_Click()
    Cmdsend.Enabled = False
    lBytes = 0
    ReDim buffer(FileLen(dlg.FileName) - 1)
    Open dlg.FileName For Binary As 1
    Get #1, 1, buffer
    Close #1
    Load Wstcp(1)
    Wstcp(1).RemoteHost = "localhost"
    Wstcp(1).RemotePort = 25000
    Wstcp(1).Connect
    Lblstatus = "Connecting..."
End Sub

Private Sub Wstcp_Close(index As Integer)
    Lblstatus = "Connection closed"
    Unload Wstcp(1)
End Sub

Private Sub Wstcp_Connect(index As Integer)
    Lblstatus = "Connected"
    Wstcp(1).SendData dlg.FileTitle & vbCrLf
End Sub

Private Sub Wstcp_DataArrival(index As Integer, ByVal bytesTotal As Long)
    Wstcp(1).GetData temp
    If InStr(temp, vbCrLf) <> 0 Then temp = Left(temp, InStr(temp, vbCrLf) - 1)
```

```
If temp = "OK" Then
    Wstcp(1).SendData buffer
Else
    Lblstatus = "Something wrong"
    Unload Wstcp(1)
    Cmdsend.Enabled = True
End If
End Sub

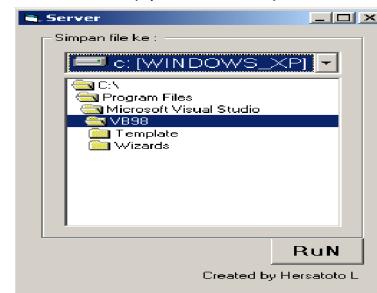
Private Sub Wstcp_SendComplete(index As Integer)
If temp = "OK" Then
    Lblstatus = "Send complete"
    temp = ""
    Unload Wstcp(1)
    Cmdsend.Enabled = True
End If
End Sub

Private Sub Wstcp_SendProgress(index As Integer, ByVal bytesSent As Long, ByVal bytesRemaining As Long)
If temp = "OK" Then
    lBytes = lBytes + bytesSent
    Lblstatus = lBytes & " out of " & UBound(buffer) & " bytes sent"
End If
End Sub
```

Untuk Server :

Objects yang diperlukan : MSWINSCK.OCX

Controls: Frame1 (Frame), Dir1 (DirListBox), Drive1 (DriveListBox), cmdRun (CommandButton), wsTCP (Winsock)



gambar 6 Aplikasi server

Source code untuk Server:

```
Dim lPos As Long
Dim bOK As Boolean
Dim fname As String

Private Sub cmdRun_Click()
    If Cmdrun.Caption = "Run" Then
        Cmdrun.Caption = "Stop"
        wsTCP(0).LocalPort = 25000
        wsTCP(0).Listen
    Else
        wsTCP(0).Close
        Cmdrun.Caption = "Run"
    End If
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive
End Sub
```

```

Private Sub wsTCP_Close(index As Integer)
    Close #1
    Unload wsTCP(1)
    bOK = False
End Sub

Private Sub wsTCP_ConnectionRequest(index As Integer, ByVal requestID As Long)
    Load wsTCP(1)
    wsTCP(1).Accept requestID
End Sub

Private Sub wsTCP_DataArrival(index As Integer, ByVal bytesTotal As Long)
    If Not bOK Then
        wsTCP(1).GetData fname
        If InStr(fname, vbCrLf) <> 0 Then fname = Left(fname, InStr(fname, vbCrLf)-1)
        bOK = True
        If Dir(Dir1.Path & "\" & fname) <> "" Then Kill Dir1.Path & "\" & fname
        Open Dir1.Path & "\" & fname For Binary As 1
        lPos = 1
        wsTCP(1).SendData "OK" & vbCrLf
    Else
        Dim buffer() As Byte
        wsTCP(1).GetData buffer
        Put #1, lPos, buffer
        lPos = lPos + UBound(buffer) + 1
    End If
End Sub

```

KESIMPULAN

1. Aplikasi pada jaringan komputer, transaksinya didasarkan pada konsep client-server.
2. Aplikasi client-server menggunakan protokol tranport untuk saling berinteraksi.
3. Protokol yang digunakan pada program pengiriman file ini adalah protokol TCP. Sdangkan socketnya adalah Winsock (Windows Socket API) untuk mendukung interface standar TCP/IP
4. Windows socket adalah perpaduan IP Address dan port number ,
5. Objects yang diperlukan pada program pengiriman file adalah MSWINSCK.OCX, comdlg32.OCX

DAFTAR PUSTAKA

docs.python.org/library/**socket**.html
 forum.vb-bego.com/viewtopic.php
 http://www.aswin.web.ugm.ac.id/content/view/51/2
 ilmukomputer.org/2006/09/27/pemrograman-**socket**-dengan-c
 inf.uajy.ac.id/~kusnadi/wp-content/uploads/tutorial1.pdf
 j3ck3y.wordpress.com/2008/06/05/**pemrograman-socket-jaringan**
 java.sun.com/j2se/1.4.2/docs/api/java/net/**Socket**.html

lecturer.ukdw.ac.id/anton/download/**PEMROGRAMAN%20JARINGAN-Modul6.pdf**
 mti.ugm.ac.id/~yudhistira/ResourceMTI/Tutorial/Cawu01/Computer%20Network/**SOCKET.ppt**
 olm.limnologi.lipi.go.id/masagus/document/index.php?action=download&filename=**Pemrograman%20Jaringan%20TCP..pdf**
 pk.datacrux.org/module.php?module=userpage&menu=104&page_id=5
 sidecode.blogspot.com/2008/12/simple-**programing-socket-on-visual.html**
 www.opengroup.org/onlinepubs/000095399/functions/**socket.html**
 www.lowtek.com/**sockets**
 www.troubleshooters.com/codcorn/**sockets**
 www.troubleshooters.com/codcorn/**sockets**
 www.benpinter.net/article.php?story=20040811203905729
 www.uwo.ca/its/doc/courses/notes/**socket/ -45k**
 www.prasannatech.net/2008/07/**socket-programming-tutorial.html**
 www.kangmaman.com/node/147
 www.osix.net/modules/article/?id=25