

VBNET Send Data to TCPIP Port.txt

Compliments of Dan @ Highpower Security Products LLC

```
'*** IMPORT STATEMENTS FOR CLASS
```

```
Imports System.IO
Imports System.Net
Imports System.Net.Sockets
```

```
'*** START OF SUBROUTINE IN CLASS
```

```
Function SendCommandTCPIP(ByVal CommandText As String, ByVal IP As String, ByVal
Port As Integer) As String
```

```
    Dim Timeout As Boolean
    Dim CharIn As Integer
```

```
    Try
```

```
        Dim Client As New TcpClient(IP, Port)
```

```
        Dim ns As NetworkStream = Client.GetStream
        Dim sw As New StreamWriter(ns)
        Dim sr As New StreamReader(ns)
```

```
        Dim LineIn As String = ""
        Dim Polling As New DateTime
        Dim CharPolling As New DateTime
        Dim WaitForResponse As DateTime
```

```
        sw.Flush()
```

```
        WaitForResponse = Date.Now.AddMilliseconds(10)
        Do While WaitForResponse >= Date.Now
            If ns.DataAvailable = True Then Exit Do
        Loop
```

```
'SEND A CR RETURN
```

```
    sw.Write(Chr(13))
    sw.Flush()
```

```
'SEND COMMAND
```

```
    sw.Write(CommandText & Chr(13))
    sw.Flush()
```

```
'WAIT FOR A RESPONSE
```

```
    'Wait DEFAULTTIMEOUTINMS For a reponse
    WaitForResponse = Date.Now.AddMilliseconds(DefaultTimeoutInMS)
```

VBNET Send Data to TCPIP Port.txt

```
Do While WaitForResponse >= Date.Now
    If ns.DataAvailable = True Then Exit Do
Loop
```

```
If ns.DataAvailable = False Then
    sw.Close()
    sr.Close()
    ns.Close()
    Client.Close()
    Return ""
End If
```

'IF NO RESPONSE THEN RETURN NOTHING

```
' Read the NetworkStream into a byte buffer.
Dim WaitForChar As DateTime
WaitForChar = Date.Now.AddMilliseconds(DefaultTimeoutInMS)
```

IF A RESPONSE THEN GET THE STREAM AND DUMP IT INTO A VARIABLE

```
check_buffer:
    CharIn = ns.ReadByte

    If CharIn = -1 Then
        If WaitForChar > Date.Now Then GoTo check_buffer Else Timeout = True
    End If

    If Timeout = True Then
        GoTo end_data
    End If

    If CharIn <> 13 Then
        LineIn = LineIn & Chr(CharIn)
        WaitForChar = Date.Now.AddMilliseconds(DefaultTimeoutInMS)
        GoTo check_buffer
    End If
```

'WE ARE DONE, DON'T FORGET TO CLOSE THE CONNECTIONS

```
end_data:    sw.Close()
            sr.Close()
            ns.Close()
            Client.Close()
            Return LineIn
```

'THIS ERROR CATCHER CATCHES ANY COMMUNICATION PROBLEMS

```
Catch ex As Exception
```

```
VBNET Send Data to TCPIP Port.txt
  MsgBox(ex.ToString)
  Return "[" & IP & "]" Stream failure."
End Try
End Function
```