

PowerShell Server and client (Sockets TCP)

Server

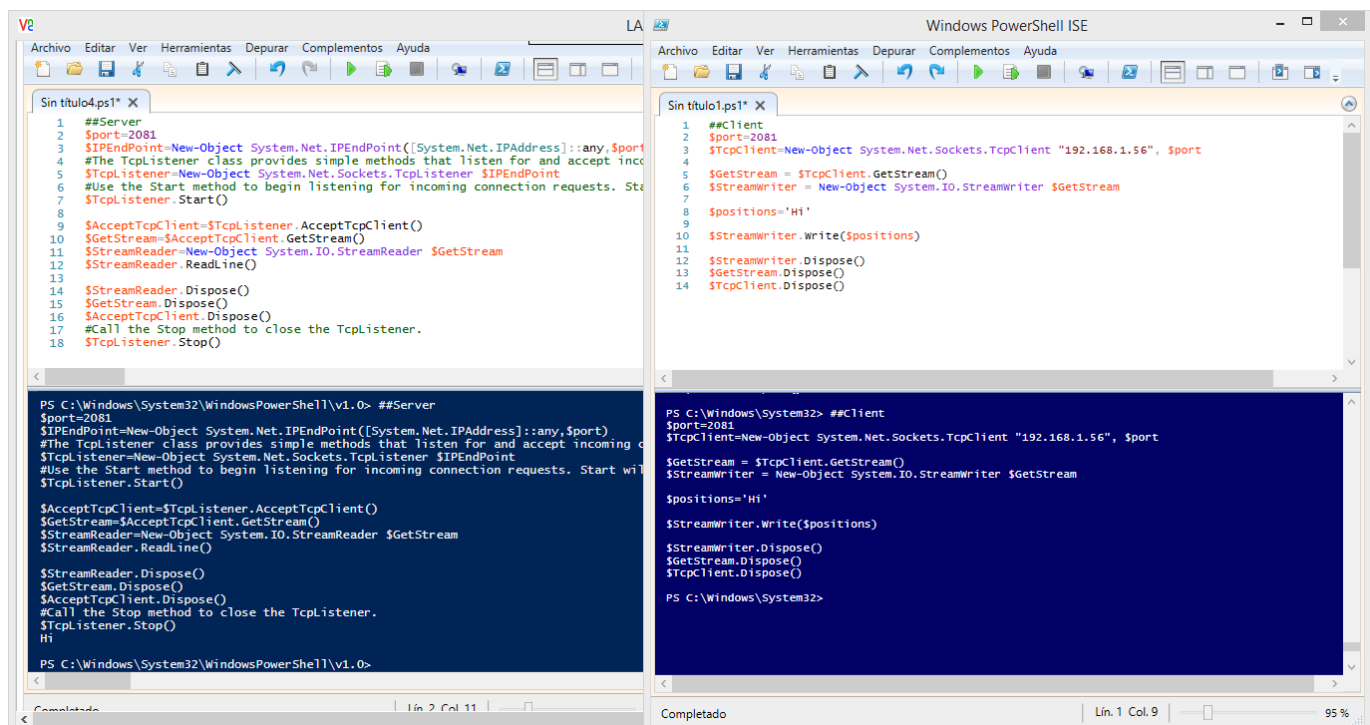
[crayon-6a9d5fcd1b187079512944/]

Client (localhost)

[crayon-6a9d5fcd1b18d252090718/]

Client (remote host)

[crayon-6a9d5fcd1b190058409078/]



The image displays two side-by-side screenshots of the Windows PowerShell ISE (Integrated Scripting Environment) interface. The left window, titled 'Sin título4.ps1', contains a PowerShell script for a TCP server. The script defines a function 'Server' that listens on port 2081 for incoming connections. It uses 'System.Net.Sockets.TcpListener' to create a listener, 'AcceptTcpClient()' to accept connections, and 'GetStream()' to retrieve the stream. It then creates a 'StreamReader' to read data from the stream and outputs the received message. The script concludes by disposing of the stream, listener, and accepting the next connection. The right window, titled 'Sin título1.ps1', contains a PowerShell script for a TCP client. It defines a function 'Client' that connects to a remote host '192.168.1.56' on port 2081. It uses 'System.Net.Sockets.TcpClient' to create a client, 'GetStream()' to retrieve the stream, and 'StreamWriter' to write the message 'Hi' to the stream. The script then disposes of the stream, writer, and client. Both windows show the script content in the editor and the execution output in the console pane at the bottom.

```
##Server
$port=2081
$IPEndPoint=New-Object System.Net.IPEndPoint([System.Net.IPAddress]::any,$port)
#The TcpListener class provides simple methods that listen for and accept incoming connections.
$TcpListener=New-Object System.Net.Sockets.TcpListener $IPEndPoint
#Use the Start method to begin listening for incoming connection requests. Start with $TcpListener.Start()
$TcpListener.Start()
$AcceptTcpClient=$TcpListener.AcceptTcpClient()
$GetStream=$AcceptTcpClient.GetStream()
$StreamReader=New-Object System.IO.StreamReader $GetStream
$StreamReader.ReadLine()
$StreamReader.Dispose()
$GetStream.Dispose()
$AcceptTcpClient.Dispose()
#Call the Stop method to close the TcpListener.
$TcpListener.Stop()

PS C:\Windows\System32\WindowsPowerShell\v1.0> ##Server
$port=2081
$IPEndPoint=New-Object System.Net.IPEndPoint([System.Net.IPAddress]::any,$port)
#The TcpListener class provides simple methods that listen for and accept incoming connections.
$TcpListener=New-Object System.Net.Sockets.TcpListener $IPEndPoint
#Use the Start method to begin listening for incoming connection requests. Start with $TcpListener.Start()
$TcpListener.Start()
$AcceptTcpClient=$TcpListener.AcceptTcpClient()
$GetStream=$AcceptTcpClient.GetStream()
$StreamReader=New-Object System.IO.StreamReader $GetStream
$StreamReader.ReadLine()
$StreamReader.Dispose()
$GetStream.Dispose()
$AcceptTcpClient.Dispose()
#Call the Stop method to close the TcpListener.
$TcpListener.Stop()
Hi
PS C:\Windows\System32\WindowsPowerShell\v1.0>
```

```
##Client
$port=2081
$TcpClient=New-Object System.Net.Sockets.TcpClient "192.168.1.56", $port
$GetStream = $TcpClient.GetStream()
$StreamWriter = New-Object System.IO.StreamWriter $GetStream
$positions='Hi'
$StreamWriter.Write($positions)
$StreamWriter.Dispose()
$GetStream.Dispose()
$TcpClient.Dispose()

PS C:\Windows\System32>

PS C:\Windows\System32> ##Client
$port=2081
$TcpClient=New-Object System.Net.Sockets.TcpClient "192.168.1.56", $port
$GetStream = $TcpClient.GetStream()
$StreamWriter = New-Object System.IO.StreamWriter $GetStream
$positions='Hi'
$StreamWriter.Write($positions)
$StreamWriter.Dispose()
$GetStream.Dispose()
$TcpClient.Dispose()
PS C:\Windows\System32>
```