

Enviar la posición del ratón constantemente entre dos equipos utilizando el protocolo UDP desde PowerShell

Equipo 1 (servidor)

```
##Server
```

```
$ip = New-Object System.Net.IPEndPoint ([IPAddress]::Any,0)
$udp = New-Object System.Net.Sockets.UdpClient 2020

do{
    try
    {
        $content = $udp.Receive([ref]$ip)
        $posicion = [Text.Encoding]::ASCII.GetString($content)
        $posicion
    }
    catch
    {
    }
}while($posicion.split(",")[0] -ne 0)

$udp.Close()
```

Equipo 2 (cliente)

```
## Cliente
```

```
$ip = New-Object System.Net.IPEndPoint ([IPAddress]::Loopback,2020)
$udp = New-Object System.Net.Sockets.UdpClient
```

```

do{
    try
    {
        $posicion =
[String]([System.Windows.Forms.Cursor]::Position.X) `
        + " , " +
[String]([System.Windows.Forms.Cursor]::Position.Y)
        $posicion
        $mensaje = [Text.Encoding]::ASCII.GetBytes($posicion)
        $udp.Send($mensaje,$mensaje.length,$ip) | Out-Null
    }
    catch
    {
    }
}while($posicion.split(",")[0] -ne 0)

$udp.Close()

```

Resultado gráfico

The image shows two side-by-side screenshots of Windows PowerShell ISE windows. The left window, titled 'Sin título1.ps1', contains a server script. The right window, titled 'Sin título1.ps1', contains a client script. Both windows show the script code and the output in the console.

Left Window (Server Script):

```

1 ##Server
2
3 $ip = New-Object System.Net.IPEndPoint ([IPAddress]::Any,0)
4 $udp = New-Object System.Net.Sockets.UdpClient 2020
5
6 do{
7     try
8     {
9         $content = $udp.Receive([ref]$ip)
10        $posicion = [Text.Encoding]::ASCII.GetString($content)
11        $posicion
12    }
13    catch
14    {
15    }
16 }while($posicion.split(",")[0] -ne 0)
17
18 $udp.Close()

```

Right Window (Client Script):

```

1 ## Cliente
2
3 $ip = New-Object System.Net.IPEndPoint ([IPAddress]::Loopback,2020)
4 $udp = New-Object System.Net.Sockets.UdpClient
5
6 do{
7     try
8     {
9         $posicion = [String]([System.Windows.Forms.Cursor]::Position.X) `
10        + " , " + [String]([System.Windows.Forms.Cursor]::Position.Y)
11        $posicion
12        $mensaje = [Text.Encoding]::ASCII.GetBytes($posicion)
13        $udp.Send($mensaje,$mensaje.Length,$ip) | Out-Null
14    }
15    catch
16    {
17    }
18 }while($posicion.split(",")[0] -ne 0)
19
20 $udp.Close()

```

Both windows show the output in the console, which is a list of coordinates (X, Y) separated by a comma and a space, such as '74,141', '47,98', '20,58', and '0,21'.