

Enviar un mensajes cifrado con Cryptographic Message Syntax (CMS) entre un cliente y un servidor mediante UDP desde PowerShell

Definición

El Cryptographic Message Syntax (CMS) es un estándar de la IETF para proteger mensajes criptográficamente. En él se definen las sintaxis para firmar digitalmente, realizar hash, realizar mac o para cifrar cualquier tipo de dato. CMS se basa en la sintaxis de PKCS#7, que a su vez se basa en el estándar Privacy-Enhanced Mail. La versión más moderna de CMS se define en la RFC 5652 (ver también la RFC 5911 de módulos ASN.1 acordes al estándar ASN.1 2002).

Información

Enviar mensajes UDP

- <https://www.jesusninoc.com/01/23/enviar-varios-mensajes-entre-un-cliente-y-un-servidor-utilizando-el-mismo-canal-de-comunicacion-por-udp-en-powershell/>

Cifrar un mensaje con Cryptographic Message Syntax (CMS) usando certificados

- <https://www.jesusninoc.com/11/18/exports-a-certificate-to-a-personal-information-exchange-pfx-file/>

Importar un certificado para descifrar un mensaje cifrado con Cryptographic Message Syntax (CMS)

- <https://www.jesusninoc.com/11/20/imports-certificates-and-private-keys-from-a-personal-information-exchange-pfx-file-to-the-destination-store/>

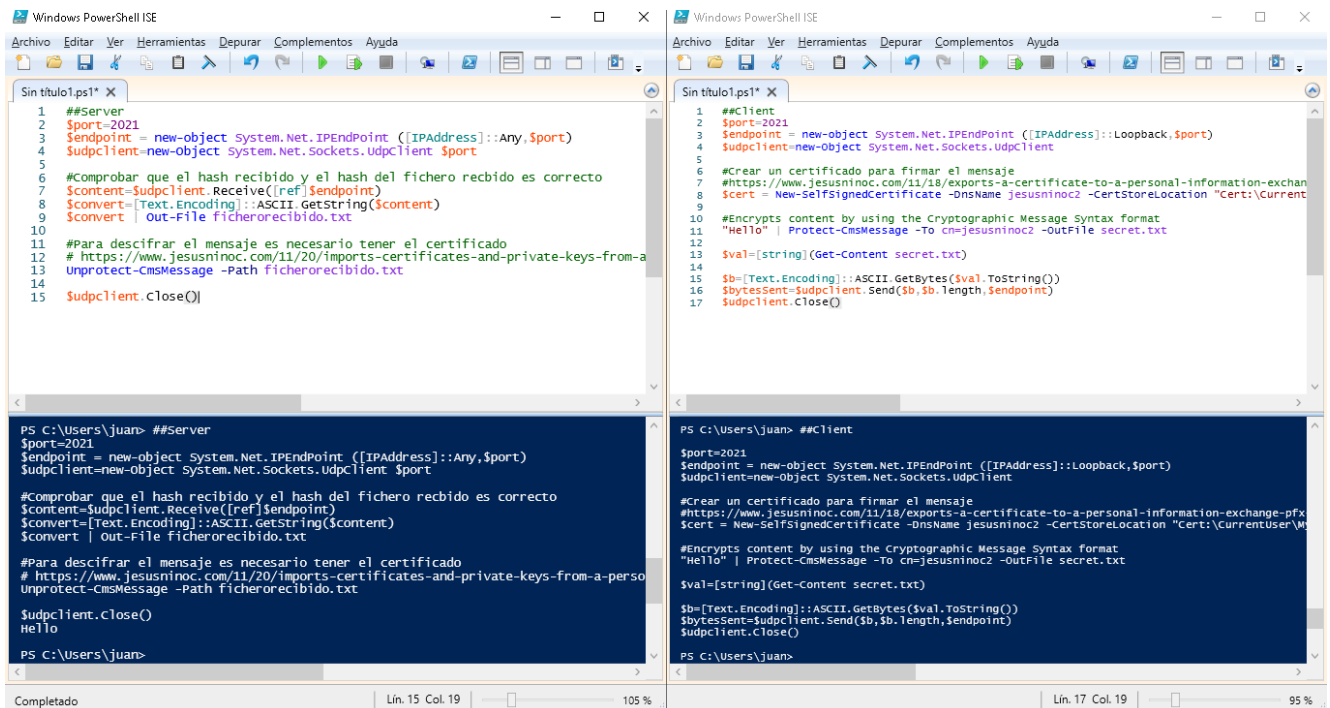
Funcionamiento

Servidor

[crayon-6a9d60822043f843030688/]

Cliente

[crayon-6a9d608220446516694800/]



```
1 ##Server
2 $port=2021
3 $endpoint = new-object System.Net.IPEndPoint ([IPAddress]::Any,$port)
4 $udpcClient=new-object System.Net.Sockets.UdpClient $port
5
6 #Comprobar que el hash recibido y el hash del fichero recibido es correcto
7 $content=$udpcClient.Receive([ref]$endpoint)
8 $convert=[Text.Encoding]::ASCII.GetString($content)
9 $convert | Out-File fichero-recibido.txt
10
11 #Para descifrar el mensaje es necesario tener el certificado
12 # https://www.jesusninoc.com/11/20/imports-certificates-and-private-keys-from-a-personal-information-exchange-pfx-file-to-the-destination-store/
13 Unprotect-CmsMessage -Path fichero-recibido.txt
14
15 $udpcClient.Close()
```

```
1 ##Client
2 $port=2021
3 $endpoint = new-object System.Net.IPEndPoint ([IPAddress]::Loopback,$port)
4 $udpcClient=new-object System.Net.Sockets.UdpClient
5
6 #Crear un certificado para firmar el mensaje
7 #https://www.jesusninoc.com/11/18/exports-a-certificate-to-a-personal-information-exchange-pfx-file-to-the-destination-store/
8 $cert = New-SelfSignedCertificate -DnsName jesusninoc2 -CertStoreLocation "Cert:\CurrentUser\My"
9
10 #Encrypts content by using the Cryptographic Message Syntax format
11 "Hello" | Protect-CmsMessage -To cn=jesusninoc2 -OutFile secret.txt
12
13 $val=[string](Get-Content secret.txt)
14
15 $b=[Text.Encoding]::ASCII.GetBytes($val.ToString())
16 $bytesSent=$udpcClient.Send($b,$b.Length,$endpoint)
17 $udpcClient.Close()
```