



UNIVERSIDAD  
NACIONAL  
AUTÓNOMA  
DE MÉXICO

# PRACTICAL SESSION 3

## CREATING ASYMMETRIC KEYS AND DIGITAL CERTIFICATES

ASIGNATURA:  
Criptografía

GRUPO: 2

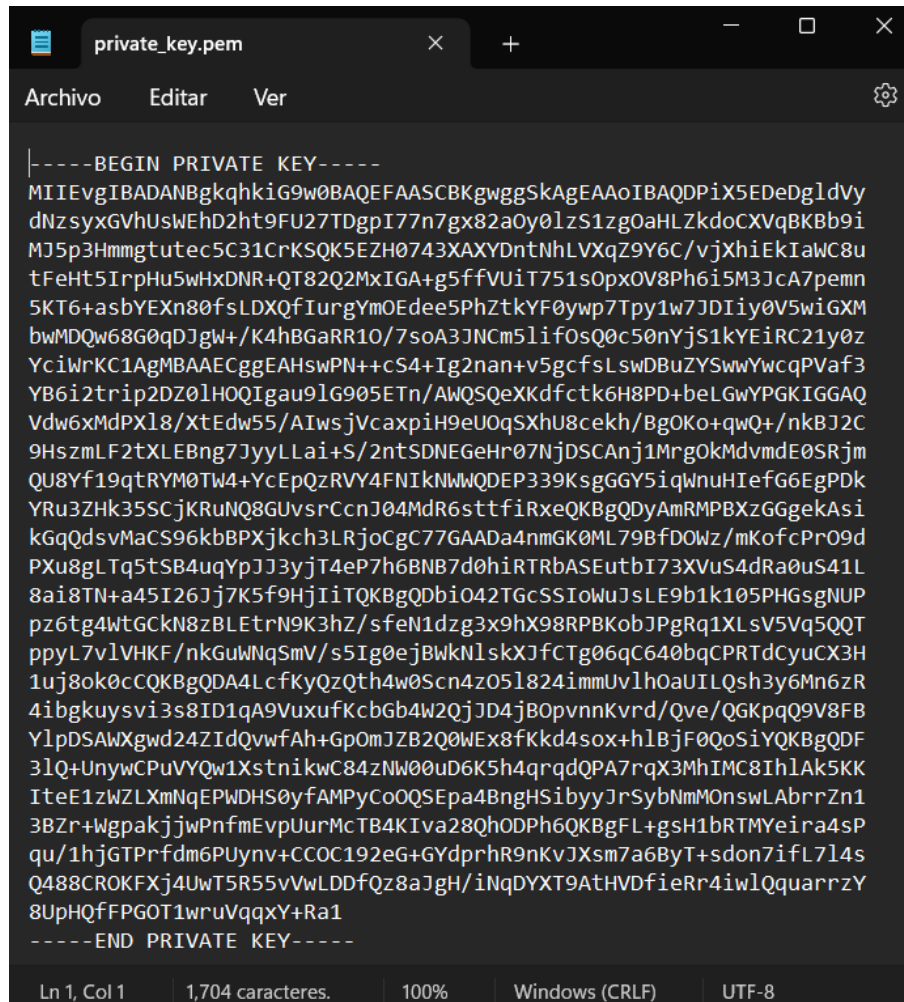
NOMBRE:  
Reyes Mendoza  
Miriam Guadalupe

FECHA: 10/05/24



FACULTAD DE INGENIERÍA

La segunda parte de la salida es una salida del sistema irrelevante o caracteres decorativos que no forman parte del comando real o sus resultados.



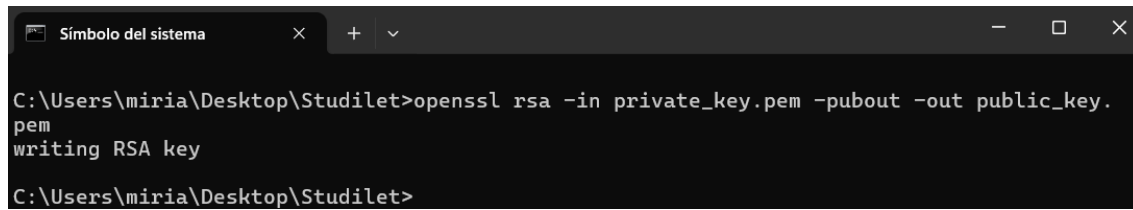
```
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAgEAAoIBAQQDPiX5EDeDgldvy
dNzsyxGVhUsWEhD2ht9FU27TDgpI77n7gx82a0y0lzS1zg0aHLZkdoCXVqBKBb9i
Mj5p3Hmmtutec5C31CrKSQK5EZHO743XAXYDntNhLVXqZ9Y6C/vjXhiEkIawC8u
tFeHt5IrpHu5wHxDNR+QT82Q2MxIGA+g5ffvUit751sOpXOV8Ph6i5M3JcA7pemn
5KT6+asbYEXn80fsLDXQfIurgYmOEde5PhZtkYF0ywp7Tpy1w7JDiiy0V5wiGXM
bwMDQw68G0qDJgw+/K4hBGAARR10/7soA3JNCm5lif0sQ0c50nYjs1kYEiRC21y0z
YciWrKC1AgMBAACggEAHswPN++cS4+Ig2nan+v5gcfsLswDBuZYSwwYwcqPVaf3
YB6i2trip2DZ0LHOQIgau9lG905ETn/AWQSQeXKdfctk6H8PD+beLGWYPGKIGGAQ
Vdw6xMdPx18/XtEdw55/AIwsjVcaxpiH9eU0qSXhU8cekh/BgOKo+qwQ+/nkBJ2C
9HszmLF2tXLEBng7JyyLLai+S/2ntSDNEGEhr07NjDSCAnj1MrgOkMdvmdE0SRjm
QU8Yf19qtrYM0TW4+YcEpQzRVY4FNiKNNWQDEP339KsgGGY5iqWnuHIefG6EgPDK
YRu3ZHK35SCjKRuNQ8GUvsrCcnJ04MdR6sttfiRxeQKBgQDyAmRMPBXzGGgekAsi
kGqQdsVMacS96kbBPXjkch3LRjoCgC77GAADa4nmGK0ML79BfDOWz/mKofcPrO9d
PXu8gLtQ5tSB4uqYpJJ3yjt4eP7h6BNB7d0hiRtRbASeutbI73XVuS4dRa0uS41L
8ai8TN+a45I26Jj7K5f9HjIiIQKBgQDbIO42TGcSSIOWuJsLE9b1k105PHGsgNUP
pz6tg4WtGCKN8zBLEtrN9K3hZ/sfeN1dzg3x9hX98RPBKobJPgRq1XLSV5Vq5QQT
ppyL7v1VHKF/nkGuWNqSmv/s5Ig0ejBwKNlskXJfCTg06qC640bqCPRtdCyuCX3H
1uj8ok0cCQKBgQDA4LcfKyQzQth4w0Scn4z05l824immUv1h0aUILQsh3y6Mn6zR
4ibgkuysvi3s8ID1qA9VuxufKcbGb4W2QjJD4jB0pvnkKvrd/Qve/QGKpqQ9V8FB
Y1pDSAWXgwd24ZIdQvwfAh+GpOmJZB2Q0WEX8fKkd4sox+h1BjF0QoSiyQKBgQDF
3lQ+UnywCPuVYQw1XstnikwC84zNW00uD6K5h4qrqdQPA7rqX3MhIMC8Ih1Ak5KK
IteE1zWZLXmNqEPWDHS0yfAMPyCoOQSEpa4BngHSibyyJrSybNmM0nswLABrrZn1
3BZr+WgpakjjwPnfmEvpUurMcTB4KIva28QhODPh6QKBgFL+gsH1bRTMYeira4sP
qu/1hjGTPrfdm6PUynv+CCOC192eG+GYdprhR9nKvJXsm7a6ByT+sdon7ifL7l4s
Q488CROKFXj4UwT5R55vVwLDDfQz8aJgH/iNqDYXT9AthVDFieRr4iwlQquarrzY
8UpHQFFPGOT1wruVqxxY+Ra1
-----END PRIVATE KEY-----
```

Ln 1, Col 1 | 1,704 caracteres. | 100% | Windows (CRLF) | UTF-8

El archivo es la representación codificada en base64 de una clave privada. Este formato se usa comúnmente para almacenar y transportar claves privadas de forma segura. El contenido principal es la clave privada en sí, codificada en formato base64. Esta codificación hace que la clave sea legible para los humanos y, al mismo tiempo, mantiene la seguridad. Decodificarla revelaría una cadena compleja de caracteres específica de la clave privada.

## CLAVE PÚBLICA RSA (2048 BITS)

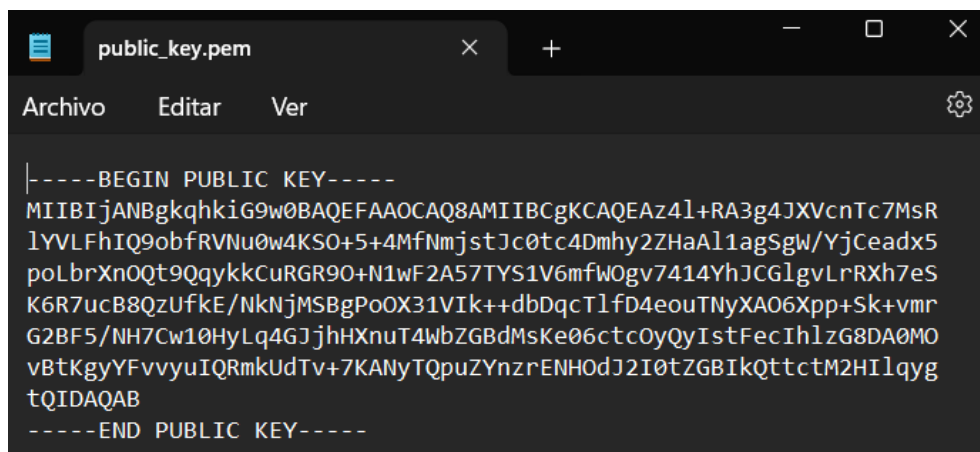
OpenSSL tiene que leer los datos de la clave privada de `private_key.pem`. Después debe extraer la información de la clave pública incrustada dentro de la clave privada. La clave pública extraída se escribe en un nuevo archivo llamado `public_key.pem`.



```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl rsa -in private_key.pem -pubout -out public_key.
pem
writing RSA key
C:\Users\miria\Desktop\Studilet>
```

Una vez que el comando finaliza correctamente, se debe tener un nuevo archivo llamado `public_key.pem` que contiene la clave pública en formato PEM. Luego puede compartir esta clave pública con cualquiera que desee cifrar mensajes o verificar las firmas digitales.

- `rsa`: Este subcomando se ocupa específicamente de las operaciones con claves RSA.
- `-in private_key.pem`: Esta opción especifica el archivo de entrada que contiene la clave privada en formato PEM (`.pem`).
- `-pubout`: Esta opción le dice a OpenSSL que extraiga la clave pública de la clave privada.
- `-out public_key.pem`: Esta opción define el nombre del archivo de salida para la clave pública extraída. También se guardará en formato PEM.

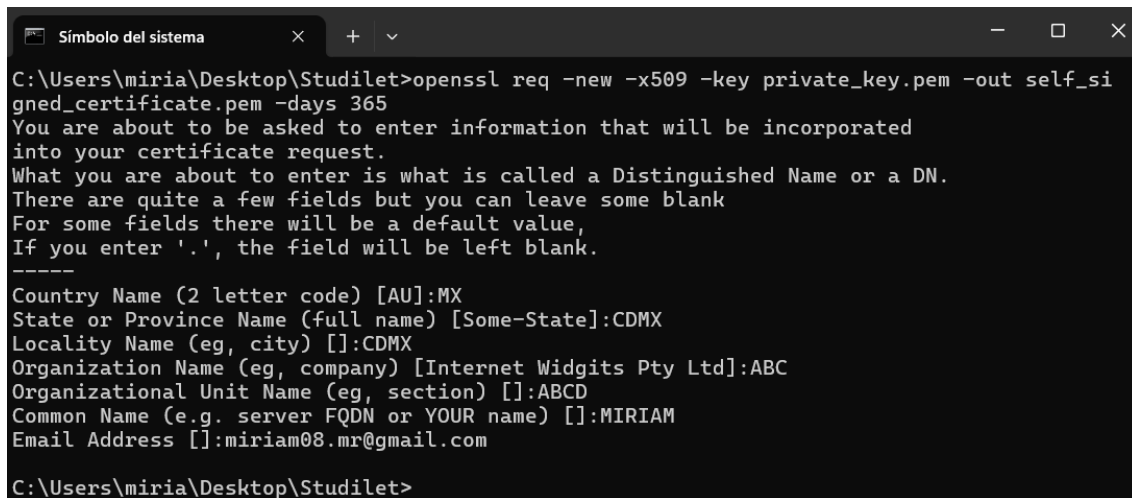


```
public_key.pem
Archivo  Editar  Ver
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAz4l+RA3g4JXVcnTc7MsR
lYVLFhIQ9obfrVNu0w4KSO+5+4MfNmjstJc0tc4Dmhy2ZHaAl1agSgw/YjCeadx5
poLbrXnOQt9QyqkCuRGR90+N1wF2A57TYS1V6mfW0gv7414YhJCGlgvLrRXh7eS
K6R7ucB8QzUfke/NkNjMSBgPoOX31VIk++dbDqcTlfd4eouTNyXA06Xpp+Sk+vmr
G2BF5/NH7Cw10HyLq4GJjhXnuT4WbZGBdMsKe06ctcOyQyIstFecIhlzG8DA0MO
vBtKgyYFvvyuIQRmkUdTv+7KANYTQpuZYnzrENH0dJ2I0tZGBIkQttctM2Hilqyg
tQIDAQAB
-----END PUBLIC KEY-----
```

El archivo que se generó es la representación codificada en base64 de una clave pública, probablemente generada por el comando `openssl rsa` que describió anteriormente. Está encerrado entre los marcadores. Este formato se usa comúnmente para almacenar y distribuir claves públicas de forma segura.

## CERTIFICADO AUTOFIRMADO (PKCS12)

Se debe generar un certificado autofirmado usando la biblioteca OpenSSL. Un certificado autofirmado actúa como una identidad digital que verifica la propiedad de un servidor o firma electrónicamente documentos digitales.



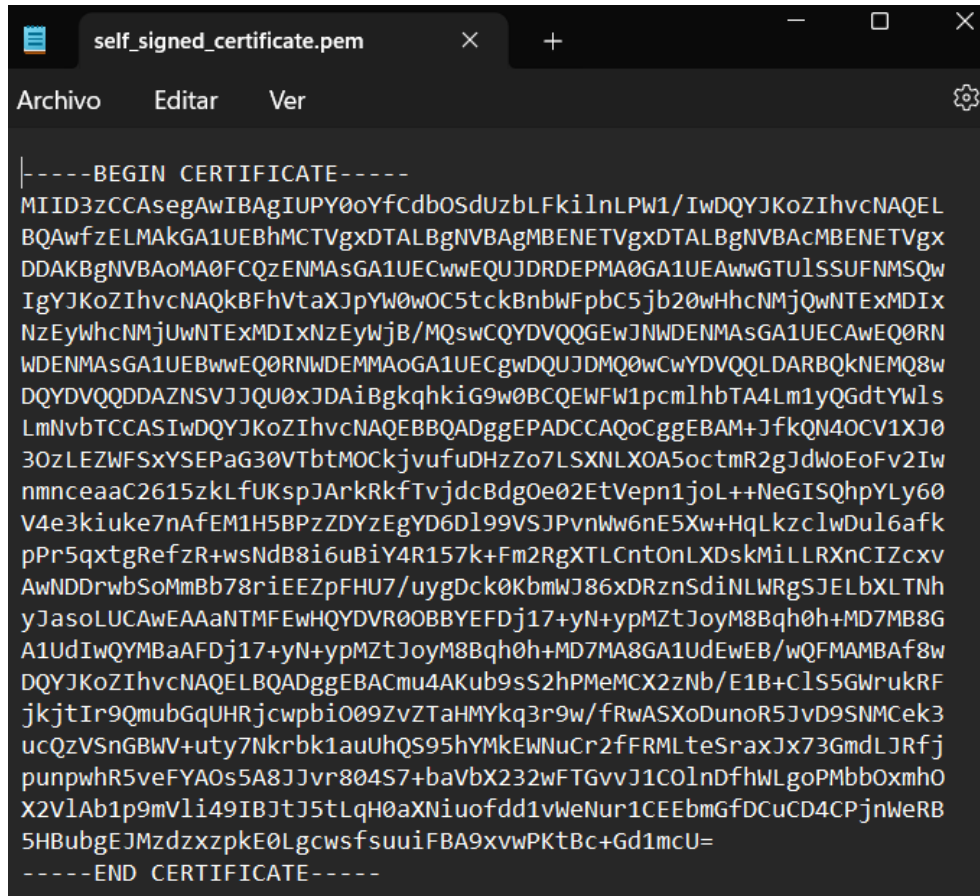
```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl req -new -x509 -key private_key.pem -out self_signed_certificate.pem -days 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:CDMX
Locality Name (eg, city) []:CDMX
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ABC
Organizational Unit Name (eg, section) []:ABCD
Common Name (e.g. server FQDN or YOUR name) []:MIRIAM
Email Address []:miriam08.mr@gmail.com
C:\Users\miria\Desktop\Studilet>
```

OpenSSL debe generar una nueva clave pública/privada si aún no tiene una, en este caso ya se obtuvo en los pasos anteriores. Posteriormente, se solicita información para completar la solicitud de certificado, como el nombre del propietario del certificado, el nombre de la organización y otra información relevante.

OpenSSL firma la solicitud de certificado con la clave privada, creando un certificado autofirmado. El certificado contendrá la clave pública e información sobre la identidad.

- `req`: Es un subcomando de OpenSSL que se ocupa específicamente de las solicitudes de certificado.
- `-new`: Esta opción indica a OpenSSL que genere una nueva solicitud de certificado.
- `-x509`: Esta opción indica a OpenSSL que genere un certificado autofirmado en formato X.509, el formato estándar para certificados digitales.
- `-key private_key.pem`: Esta opción especifica el archivo que contiene la clave privada. El certificado se firmará con esta clave privada.

- `-out self_signed_certificate.pem`: Esta opción define el nombre del archivo de salida para el certificado autofirmado.
- `-days 365`: Esta opción establece la validez del certificado en 365 días (un año). Se puede ajustar este valor según las necesidades.



```

self_signed_certificate.pem
Archivo  Editar  Ver

-----BEGIN CERTIFICATE-----
MIID3ZCCAsEgAwIBAgIUYP0oYfCdbOSdUzbLFkiLnLPw1/IwDQYJKoZIhvcNAQEL
BQAwfzELMAkGA1UEBhMCTVgxDTALBgNVBAGMBENETVgxDTALBgNVBACMBENETVgx
DDAKBgNVBAoMA0FCQzENMASGA1UECwwEQUJDRDEPMA0GA1UEAwGTU1SSUFNMSQw
IgYJKoZIhvcNAQkBFhVtaXJpYW0wOC5tcKbNbWpBc5jb20wHhcNMjQwNTExMDIx
NzEyWWhcNMjQwNTExMDIxNzEyWjB/MQswCQYDVQQGEwJNWDENMASGA1UEAwEQ0RN
WDENMASGA1UEBwwEQ0RNWDENMAoGA1UECgwDQUJDMQ0wCwYDVQQLDARBRQkNEMQ8w
DQYDVQQDDAZNSVJJQU0xJDAiBgkqhkiG9w0BCQEWFW1pcmlhbTA4Lm1yQGdtYWls
LmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM+JfkQN40CV1XJ0
30zLEZWFSxYSEPaG30VTbtMOCKjvufuDHZzo7LSXNLXOA5octmR2gJdWoEoFv2Iw
nmnceaaC2615zkLFUKspJArKrkFtvjdcBdgOe02EtVepn1joL++NeGISQhpYLY60
V4e3kiuke7nAfEM1H5BPzZDYzEgYD6Dl99VSJPvnWw6nE5Xw+HqLkzc1wDul6afk
pPr5qxtgRefzR+wsNdB8i6uBiY4R157k+Fm2RgXTLCntOnLXDskMiLLRXnCIzcxv
AwNDDrwbSoMmBb78riEEZpFHU7/uygDck0KbmWJ86xDRznSdiNLWRGSJELbXLTNh
yJasoLUCAwEAAaNTMFEWtH5BPzZDYzEgYD6Dl99VSJPvnWw6nE5Xw+HqLkzc1wDul6afk
A1UdIwQYMBaAFDj17+yN+ypMZtJoyM8Bqh0h+MD7MA8GA1UdEwEB/wQFMAMBAf8w
DQYJKoZIhvcNAQELBQADggEBACmu4AKub9sS2hPMeMCX2zNb/E1B+ClS5GwrukRF
jkjtIr9QmubGqUHRjcwpbi009ZvZTaHMYkq3r9w/fRwASXoDunoR5JvD9SNMcek3
ucQzVSngBWV+uty7Nkrbk1auUHQs95hYMKEWNUcr2fFRMLteSraxJx73GmdLJRfj
punpwhR5veFYA0s5A8JJvr804S7+baVbX232wFTGvvJ1C0InDfhwLgoPMbb0xmH0
X2VlAb1p9mVli49IBJtJ5tLqH0aXNiuofdd1vWeNur1CEEbmGfDCuCD4CPjnWeRB
5HBubgEJMzdzxpkE0LgcwsfsuuiFBA9xvwPKtBc+Gd1mcU=
-----END CERTIFICATE-----

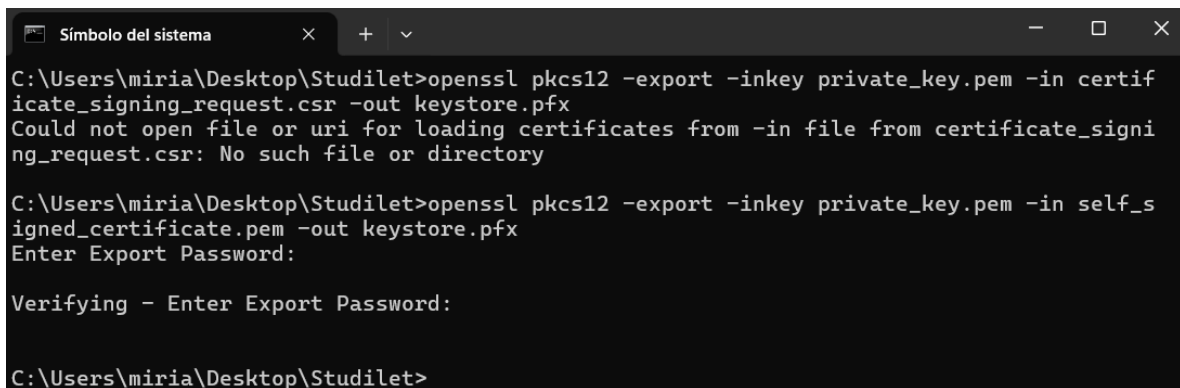
```

Tenemos un certificado X.509 codificado en base64 que contiene información sobre el propietario, su período de validez y la clave pública utilizada para la verificación.

Es importante saber que los certificados solo contienen información de clave pública, mientras que los archivos PKCS12 contienen ambas claves, privada y pública. Por lo que hasta este punto el certificado carece del componente de clave privada.

Ya que el comando anterior genera un certificado autofirmado a partir de la clave privada. Ahora, se tendrá que crear un archivo PKCS12 que será llamado `keystore.pfx` y que deberá contener tanto la clave privada como el certificado autofirmado y para lo cual se pedirá que se ingrese una contraseña para proteger el archivo.

OpenSSL leera la clave privada del archivo `private_key.pem`. Posteriormente deberá leer el certificado digital autofirmado del archivo `self_signed_certificate.pem`, combinando de forma segura la clave privada y el certificado en un único archivo PKCS12. Se solicitará que establezca una contraseña para proteger el archivo PKCS12. Es fundamental elegir una contraseña segura y mantenerla confidencial.



```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl pkcs12 -export -inkey private_key.pem -in certificate_signing_request.csr -out keystore.pfx
Could not open file or uri for loading certificates from -in file from certificate_signing_request.csr: No such file or directory

C:\Users\miria\Desktop\Studilet>openssl pkcs12 -export -inkey private_key.pem -in self_signed_certificate.pem -out keystore.pfx
Enter Export Password:

Verifying - Enter Export Password:

C:\Users\miria\Desktop\Studilet>
```

Este comando se utiliza para crear un archivo en formato PKCS12 usando la biblioteca OpenSSL. Un archivo PKCS12 (Personal Information Exchange Syntax Standard #12) combina de forma segura dos la clave privada y el certificado digital asociado (en este caso, un certificado autofirmado).

El archivo PKCS12 suele estar protegido con contraseña, lo que lo convierte en una forma segura de almacenar y transportar conjuntamente la clave privada y el certificado relacionado.

- `pkcs12`: Es un subcomando de OpenSSL específico para la gestión de archivos PKCS12.
- `-export`: Esta opción indica a OpenSSL que cree un archivo PKCS12 a partir de los elementos proporcionados.
- `-inkey private_key.pem`: Esta opción especifica el archivo que contiene la clave privada.
- `-in self_signed_certificate.pem`: Esta opción especifica el archivo que contiene el certificado digital que desea incluir en el archivo PKCS12. En este caso, se supone que es un certificado autofirmado generado previamente.



- `-out keystore.pfx`: Esta opción define el nombre del archivo de salida para el archivo PKCS12 creado. El `.pfx` es una extensión de archivo común para los archivos PKCS12.

Se debe usar un comando que muestre detalles como el emisor, sujeto y validez del certificado contenido en el archivo PFX, pero que no muestra la clave privada ya que generalmente está encriptada y se requeriría una frase de paso adicional.

```

C:\Users\miria\Desktop\Studilet>openssl pkcs12 -info -in keystore.pfx
Enter Import Password:

MAC: sha256, Iteration 2048
MAC length: 32, salt length: 8
PKCS7 Encrypted data: PBES2, PBKDF2, AES-256-CBC, Iteration 2048, PRF hmacWithSHA256
Certificate bag
Bag Attributes
    localKeyID: 98 12 02 79 77 93 02 BF 2C DC 67 F9 F7 CE 62 81 23 E1 0A 29
subject=C=MX, ST=CDMX, L=CDMX, O=ABC, OU=ABCD, CN=MIRIAM, emailAddress=miriam08.mr@gmail.com
issuer=C=MX, ST=CDMX, L=CDMX, O=ABC, OU=ABCD, CN=MIRIAM, emailAddress=miriam08.mr@gmail.com
-----BEGIN CERTIFICATE-----
MIID3zCCAsAgAwIBAgIUPY0oYfCdb0SdUzbLFkIlNLPW1/IwDQYJKoZIhvcNAQEL
BQAwfzELMAkGA1UEBhMCTVgxDTALBgNVBAGMBENETVgxDTALBgNVBACMBENETVgx
DDAKBgNVBAoMA0FCQzENMA5GA1UECwwEQUJDRDEPMA0GA1UEAwwGTUlsSUFNMQw
IgYJKoZIhvcNAQkBFhVtaXJpYW0wOC5tckBnbWVpbC5jb20wHhcNMjQwNTEwMDIx
NzEyWhcNMjQwNTEwMDIxNzEyWjB/MQswCQYDVQQGEwJNWDENMA5GA1UECAwEQ0RN
WDENMA5GA1UEBwwEQ0RNWDENMA5GA1UECgwDQUJDMQ0wCwYDVQQLDARBQkNEMQ8w
DQYDVQQDDAZNSVJJQU0xJDAiBgkqhkiG9w0BCQEWFW1pcmlhbTA4Lm1yQGdtYWls
LmNvbTCCASAwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM+JfkQN40CV1XJ0
30zLEZWFsYSEPaG30VTbtMOckjvufuDHZzo7LSXNLX0A5octmR2gJdWoEoFv2Iw
nmnceaaC2615zkLfUKspJArkRkfTvjdcBdg0e02EtVepn1joL++NeGISQhpYLy60
V4e3kiue7nAfEM1H5BPzZDYzEgYD6Dl99VSJPvnWw6nE5Xw+HqLkzclwDul6afk
pPr5qxtgRefzR+wsNdB8i6uBiY4R157k+Fm2RgXtLCntOnLXdskMiLLRXnCIzcxv
AwNDDrwbSoMmBb78riEEZpFHU7/uygDck0KbmWJ86xDRznSdiNLWRgSJELbXLTNh
yJasoLUCAwEAAaNTMFEwHQYDVR00BBYEFdj17+yN+ypMZtJoyM8Bqh0h+MD7MB8G
A1UdIwQYMBaAFDj17+yN+ypMZtJoyM8Bqh0h+MD7MA8GA1UdEwEB/wQFMAMBAf8w
DQYJKoZIhvcNAQELBQADggEBACmu4AKub9sS2hPMcMX2zNb/E1B+ClS5GWrukRF
jkjtIr9QmubGqUHRjcwpmi009ZvZTaHMYkq3r9w/fRwASXoDunoR5JvD9SNMCek3
ucQzVSngBWV+uty7Nkrbk1auUhQS95hYmkEWNuCr2fFRMLteSraxJx73GmdLJRfj
punpwhR5veFYA0s5A8JJvr804S7+baVbX232wFTGvvJ1C0LnDfhWLgoPMbb0xmh0
X2VlAb1p9mVli49IBJtJ5tLqH0aXNiuofdd1vWeNur1CEEbmGfDCuCD4CPjnWeRB
5HBubgEJMzdzxzkE0LgcwsfsuuiFBA9xvwPKtBc+Gd1mcU=
-----END CERTIFICATE-----
PKCS7 Data
Shrouded Keybag: PBES2, PBKDF2, AES-256-CBC, Iteration 2048, PRF hmacWithSHA256
Bag Attributes
    localKeyID: 98 12 02 79 77 93 02 BF 2C DC 67 F9 F7 CE 62 81 23 E1 0A 29
Key Attributes: <No Attributes>

```



```
Símbolo del sistema
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFNTBfBgkqhkiG9w0BBQ0wUjAxBgkqhkiG9w0BBQwwJAQQYC0qVRvyQ5Edz6Ed
yW3hsgICCAAwDAYIKoZIhvcNAgkFADAdBglgkgBZQMEASoEEAom0yJFg4oKcwlF
4D9+YnUEggTQ64LTi1Pxqk+TB/BSnazRnvkPrpbWqCSrGFnjRN07KUjQxj6g0Vg9
fQhEEbtRhPjjFejmH+hVhsdrckkJ9uL2ERMJohjZpxsUqhZPwx/MTso009Ks3gB5
OGgKa1jj7ZHN9PkhTxJ30CKHWjHgZF28yaVS0XCFgZpV72EuP07h5JSZWZtEmtH
kGY7vYAXKXJvb1+8+kL0q2RIsF/+8nEuVfEeBPaqR7c5wWtDYz5FmcEzYguqu47U
6IrbEAqMT5XVoYJcm/6+xPn7kkWRRnIjJukiQVG05Rc1sUGHqssmk6aiB4CLcF7L
q5WpSgLMqfLiIHp8VrU/WnfKmH9D0JrN5kJBn0d1kRuykL0bVHrMrH9WxOWKfBkt
NpYRjVpTh5daq3LzXc65vJK9B181SDTiEq6gSTn0BoiF6BD411sLLYbA4V9NyGGV
jSeA5h5Tm4vVXVekqpWJyQsL3Z0nHdCZadKbCgb6K3/WPnj8T2j9UUwpBFpF9yva
9GfpznQJVC9ZArDZHuWG0ihKHqmb5017tRiGtbQHK+1J2cAq/MnbZedbG4jeTu0o
Ev69okIcNKs9aWoz08PpJCmBLDnzWPg6Z9AYQoDcsxYM9WKvBWFLop3E3z0WRSht
srScaRFCdAgzWPyCJJ1jHYgBBzLoPNxe7y6Rk8+LvMddIxM/wLSxvFbGklpBIuBb
rV1BB4lu0zrfovJL8z7r2AxE6EpeXCL2FvNaiPgZun0S0+nXNjWTZi2nUvn9CFg
MWAYaoLgaoZp5IglURu306uqr5+8SrILlfYnRWBLmu+hPqsBuwun2ggLLgCu3x
ycb4fSn0zC5GAng7mw+Cyuqr8G0s/3GuLZs6VLGy0PRwyEvvj2ZWqRCc0eKVq4qn
F9/QkPRhGiXzZjNn5qjybcyJO9PxiXA4AonlhgMa+GiQUuN6y9gTrjQqmat92h6Q
tf+nenk6nLTAt/8qFcd1Lz4M8H6VLSamcMG8XYJXbASeEuEwuh/i5nwaZEXb1k2Q
iPnqrSq60IZQ4Cbv0zS4/LexZem4CYor8m0VKEALhW9YAiQoQU8syxDbdZMtVHqv
9xtG0L/WrUHzzFbrd/Vk/Q/oT66dT2j8ajzhaX9EtvfjeXbLbuqzS2rxJAhhWFnf
xRMhcU5ERNwQM5BOC3Pt4YVaE0X/WkVKkwJ0EXdpii7XuyheTQW9lfSULVC1u25E
mQZLJ1E4L4za9mVL7/fkOzP0buRZnqbsb8cM0NCp0toYPoaizy1Wy/n8Bkc21aFY
C9H43wAZ7cFwxMektIFC+Ana/X74nQdPMID+/+ad9orZApXBH8CELCPIYbg56zTM
Mjffj9qLzfuq5dj1kPTanYPPiKnwyWHNq8tSLME2kJS50NMsioPmdNBxxy5kCsm5H
QZD6yVB9ACMEBqY6KyN80z47UiUYPTheLLKjTi+URvhaMQUpwGMEjEnbkeAGnXH
7xKuE5HL6ISBp7aQkzg0+xDj+DPzvLLJ3mYwwLhQ7k/2ndTRqm9Y4/aqqK0vbS9P
vDSV5ZaE6xRTR0EEp8vUpDvgkLhoaADvM8wRt0fE66BNFNryhCKnFcLjNkcnEt9S
twaeFe1LXomYgEkjJdLiKI+XxwWheilMDr2D33WTNHZAipmjWLPHiy8=
-----END ENCRYPTED PRIVATE KEY-----

C:\Users\miria\Desktop\Studilet>
```

Después de ingresar la contraseña correcta, la salida contiene detalles sobre el certificado dentro del archivo PKCS12. La información incluye el país, estado, localidad, organización, unidad organizativa, nombre común y dirección de correo electrónico. Se muestra el certificado Base64 codificado entre los marcadores --- --BEGIN CERTIFICATE----- y -----END CERTIFICATE-----.

La salida indica que la clave privada está encriptada y se necesita una frase de paso PEM separada para acceder a ella. Mostrando los datos de la clave privada cifrada almacenados en formato PEM.

## CERTIFICADO FIRMADO POR UNA AUTORIDAD

Getacert requiere una Solicitud de Firma de Certificado (CSR) durante el proceso de solicitud. Por lo que se puede usar el mismo comando usado anteriormente para generar la CSR.

```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl req -new -key private_key.pem -out miriamyi01.github.io.csr
You are about to be asked to enter information that will be incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:CDMX
Locality Name (eg, city) []:CDMX
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ABC
Organizational Unit Name (eg, section) []:ABCD
Common Name (e.g. server FQDN or YOUR name) []:MIRIAM
Email Address []:miriam08.mr@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:12345
An optional company name []:12345

C:\Users\miria\Desktop\Studilet>
```

Este comando se utiliza con OpenSSL para generar una Solicitud de Firma de Certificado (CSR), que es una parte esencial del proceso para obtener un certificado SSL/TLS para mi sitio web.

- `req`: Este subcomando le indica a OpenSSL que deseas utilizar la funcionalidad de solicitud.
- `-new`: Esta opción especifica que deseas generar una nueva CSR.
- `-key private_key.pem`: Esta opción le dice a OpenSSL que use la clave privada almacenada en el archivo `private_key.pem` al crear la CSR.
- `-out miriamyi01.github.io.csr`: Esta opción especifica el nombre del archivo para el archivo CSR de salida. En este caso, la CSR se guardará como `miriamyi01.github.io.csr`.

El archivo CSR generado contiene la parte pública del par de claves, que se puede compartir libremente. Se utiliza para cifrar información que solo la clave privada puede descifrar. También, especifica el nombre de dominio para el que se solicita el certificado SSL/TLS. Puede incluir detalles sobre la organización, pero también se puede dejar en blanco según los requisitos de la Autoridad Certificadora (CA). Por último, está la firma digital creada con la clave privada que ayuda a verificar la autenticidad de la CSR.

```
miriamyi01.github.io.csr
Archivo  Editar  Ver

-----BEGIN CERTIFICATE REQUEST-----
MIIC8DCCAdgCAQAwfzELMAkGA1UEBhMCTVgxDTALBgNVBAGMBENETVgxDTALBgNV
BACMBENETVgxDDAKBgNVBAoMA0FCQzENMA5GA1UECwwEQUJRDEPMA0GA1UEAwwG
TU1SSUFNMQwIgYJKoZIhvcNAQkBFhVtaXJpYW0wOC5tckBnbwFpbC5jb20wgGEi
MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQPXiX5EdeDglDVydNzsyxGVhUsw
EhD2ht9FU27TDgpI77n7gx82a0y0lZs1zg0aHLZkdoCXVqBKBb9iMj5p3Hmmtut
ec5C31CrKSQK5EZH0743XAXYDntNhLVXqZ9Y6C/vjXhiEkIawC8utFeHt5IrpHu5
wHxDNR+QT82Q2MxIGA+g5ffVUiT751s0px0V8Ph6i5M3JcA7pemn5KT6+asbYEXn
80fsLDXQfIurgYmOEde5PhZtkYF0ywp7Tpy1w7JDIiy0V5wiGMBwMDQw68G0qD
JgW+/K4hBGaRR10/7soA3JNCm5liF0sQ0c50nYjS1kYEiRC21y0zYciWrKC1AgMB
AAGGLDAUBGkqhkiG9w0BCQIxBwwFMTIzNDUwFAYJKoZIhvcNAQkHMqCMBTEyMzQ1
MA0GCSqGSIb3DQEBQwUAA4IBAQCcvJVGf9rYtBo13TiDeZ7FHshkQ7oCh12dVvpX
RJcoRy44j52VLRx6T++WeFpFwIwReEjuoGwaxCujKcy/LMITOQ6rt6dAr8aTID
K1NWVh1NMGPSMN8jxYLzVwZc8LRqXByeM9hP3V1kHQ+19Hhiv/OREWzHCEprzoDn
f04JZlInxyNg0sUjSxu6IGlCMXNXneXhF4E7k/LeewGILPY0kw7WgTMyuvix/PhR
wB9S9M9YmgFBdKMDr7xsLLoYBAVFCltA0Q0yFuZaVghLLCyGjGHVr13D3yzVUJrJ
VtFverqCj/z1gj0xXHyTiTh+ubPlnDTYt4oxHrSuflo+7mTg
-----END CERTIFICATE REQUEST-----

Ln 1, Col 1  1,094 caracteres.  100%  Windows (CRLF)  UTF-8
```

Una vez que generamos la CSR, normalmente enviaremos a una Autoridad Certificadora (CA). La CA validará la propiedad del nombre de dominio y emitirá un certificado SSL/TLS firmado basado en la información de la CSR. Luego se puede instalar el certificado en nuestro servidor web para habilitar conexiones HTTPS seguras para el sitio web.



Submit your Certificate Signing Request(CSR) to be decoded and signed by getaCert. Here you can submit your CSR and it will be decoded instantly. This tool is useful to verify that your certificate is valid or to display the information held in the CSR.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIC8DCCAdgCAQAwfzELMAkGA1UEBhMCTVgxDTALBgNVBAGMBENETVgxDTALBgNV
BACMBENETVgxDDAKBgNVBAoMA0FCQzENMA5GA1UECwwEQUJRDEPMA0GA1UEAwwG
TU1SSUFNMQwIgYJKoZIhvcNAQkBFhVtaXJpYW0wOC5tckBnbwFpbC5jb20wgGEi
MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQPXiX5EdeDglDVydNzsyxGVhUsw
EhD2ht9FU27TDgpI77n7gx82a0y0lZs1zg0aHLZkdoCXVqBKBb9iMj5p3Hmmtut
ec5C31CrKSQK5EZH0743XAXYDntNhLVXqZ9Y6C/vjXhiEkIawC8utFeHt5IrpHu5
wHxDNR+QT82Q2MxIGA+g5ffVUiT751s0px0V8Ph6i5M3JcA7pemn5KT6+asbYEXn
80fsLDXQfIurgYmOEde5PhZtkYF0ywp7Tpy1w7JDIiy0V5wiGMBwMDQw68G0qD
JgW+/K4hBGaRR10/7soA3JNCm5liF0sQ0c50nYjS1kYEiRC21y0zYciWrKC1AgMB
AAGGLDAUBGkqhkiG9w0BCQIxBwwFMTIzNDUwFAYJKoZIhvcNAQkHMqCMBTEyMzQ1
MA0GCSqGSIb3DQEBQwUAA4IBAQCcvJVGf9rYtBo13TiDeZ7FHshkQ7oCh12dVvpX
RJcoRy44j52VLRx6T++WeFpFwIwReEjuoGwaxCujKcy/LMITOQ6rt6dAr8aTID
K1NWVh1NMGPSMN8jxYLzVwZc8LRqXByeM9hP3V1kHQ+19Hhiv/OREWzHCEprzoDn
f04JZlInxyNg0sUjSxu6IGlCMXNXneXhF4E7k/LeewGILPY0kw7WgTMyuvix/PhR
wB9S9M9YmgFBdKMDr7xsLLoYBAVFCltA0Q0yFuZaVghLLCyGjGHVr13D3yzVUJrJ
VtFverqCj/z1gj0xXHyTiTh+ubPlnDTYt4oxHrSuflo+7mTg
-----END CERTIFICATE REQUEST-----|

Submit CSR
```



### Decoded CSR - Summary

Attribute	Value
C	MX
ST	CDMX
L	CDMX
O	ABC
OU	ABCD
CN	MIRIAM
emailAddress	miriam08.mr@gmail.com

your signed public certificate(.cer) : [MIRIAM-2024-05-11-033328.cer](#)

Open your signed public certificate(.cer) +

in pem format(.pem) : [MIRIAM-2024-05-11-033328.pem](#)

Open in pem format(.pem) +

getaCert's public certificate/key(.cer) : [getacert.cer](#)

Open getaCert's public certificate/key(.cer) +

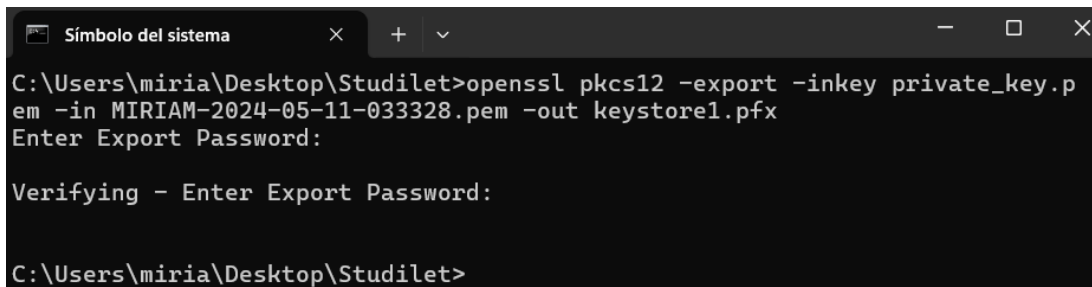
This certificate is signed for 60 days. If you would like a 10 year certificate please Donate and we will send you a 10 year CSR signing page

Getacert proporciona dos certificados esenciales para proteger la conexión:

- **Certificado público firmado (.cer o .pem):** Este es el certificado más importante para el sitio web. Contiene:
  - Nombre de dominio.
  - La información de la clave pública utilizada para cifrar los datos enviados al sitio web.
  - La firma digital de Getacert que verifica la legitimidad del certificado.
  - Los formatos DER (.cer) y PEM (.pem). Ambos se usan ampliamente para la instalación de certificados, y se puede elegir cualquiera de los dos.
- **Certificado público de Getacert (.cer):** Este certificado pertenece a Getacert, la Autoridad Certificadora (CA) que emitió el certificado público firmado. Contiene la información de la clave pública de Getacert. Este certificado podría ser necesario durante la instalación para establecer una cadena de confianza. El navegador web confía en Getacert y Getacert firma el certificado. Esta cadena verifica la autenticidad del certificado.

El período de validez de 60 días se aplica al certificado público firmado. Esto significa se deberá renovar antes de que expire para mantener una conexión segura.

Una vez que se tiene el certificado descargado, se utiliza OpenSSL para crear un almacén de claves PKCS12 que contenga la clave privada y el certificado firmado.



```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl pkcs12 -export -inkey private_key.p
em -in MIRIAM-2024-05-11-033328.pem -out keystore1.pfx
Enter Export Password:

Verifying - Enter Export Password:

C:\Users\miria\Desktop\Studilet>
```

Este comando se utiliza con OpenSSL para crear un archivo de almacén de claves PKCS12. Un almacén de claves PKCS12 es un formato de archivo único que almacena de forma segura la clave privada y certificado juntos. Se suele utilizar para facilitar la implementación y la administración de los certificados SSL/TLS en servidores web.

- `pkcs12`: Este subcomando le indica a OpenSSL que desea trabajar con almacenes de claves PKCS12.
- `-export`: Esta opción especifica que desea exportar la clave y certificado a un archivo PKCS12.
- `-inkey private_key.pem`: Esta opción le indica a OpenSSL que use la clave privada almacenada en el archivo `private_key.pem`. Esta clave privada es el componente esencial para descifrar datos cifrados con la clave pública.
- `-in MIRIAM-2024-05-11-033328.pem`: Esta opción especifica el archivo que contiene el certificado público firmado, por Getacert en este caso.
- `-out keystore1.pfx`: Esta opción especifica el nombre del archivo para el archivo de almacén de claves PKCS12 de salida. En este caso, el almacén de claves se guardará como `keystore.pfx`.

Enter Export Password y Verifying - Enter Export Password con indicaciones que piden establecer una contraseña para el almacén de claves PKCS12. Esta contraseña protege la clave privada y el certificado almacenados dentro del archivo. Es crucial elegir una contraseña segura y única, y mantenerla confidencial.

## VERIFICACIÓN

keystore1.pfx

10/05/2024 09:57 p. m.

Intercambio de información personal

3 KB

Se utiliza un comando para extraer y mostrar información sobre el certificado y la clave privada almacenados dentro de un archivo de almacén de claves PKCS12. En este caso específico, analiza el archivo `keystore1.pfx`. Proporciona información sobre el contenido del almacén de claves sin necesidad de descifrar la clave privada.

El resultado muestra la información extraída, recordando que en un archivo PKCS12, la información de la llave privada, como sus atributos, se protege mediante cifrado y no se incluye directamente en la salida del comando. Esto se hace por razones de seguridad para evitar que la información confidencial de la llave sea expuesta accidentalmente. Este comando descifró y mostró exitosamente detalles acerca del certificado y la llave privada almacenados dentro del archivo `keystore1.pfx`.

```
Símbolo del sistema
C:\Users\miria\Desktop\Studilet>openssl pkcs12 -info -in keystore1.pfx
Enter Import Password:

MAC: sha256, Iteration 2048
MAC length: 32, salt length: 8
PKCS7 Encrypted data: PBES2, PBKDF2, AES-256-CBC, Iteration 2048, PRF hmacW
ithSHA256
Certificate bag
Bag Attributes
    localKeyID: 4D B4 C2 53 ED 71 28 A3 03 FF 78 9E 6A C8 23 0B 30 36 2B 7A

subject=C=MX, ST=CDMX, L=CDMX, O=ABC, OU=ABCD, CN=MIRIAM, emailAddress=miri
am08.mr@gmail.com
issuer=C=US, ST=Washington, L=Seattle, O=getaCert - www.getacert.com
-----BEGIN CERTIFICATE-----
MIIDoTCCAomgAwIBAgICCC+owDQYJKoZIhvcNAQELBQAwWjELMAkGA1UEBhMCVVMx
EzARBgNVBAGTCldhc2hpbmd0b24xEDAOBgNVBAcTB1NlYXR0bGUxJDAiBgNVBAoT
G2dlldGFDXJ0IC0gd3d3LmdldGFjZjZJ0LmNvbTAeFw0yNDA1MTEwMzZmMjhaFw0y
NDA3MTAwMzZmMjhaMH8xCzAJBgNVBAYTAk1YMjQwCwYDVQIDARE1YMjQwCwYD
VQQHDARE1YMjQwCwYDVQKDBQKxDTALBgNVBAsMBEFCQ0QxZDZANBgNVBAMM
Bk1JUKlBTTEkMCIGCSqGSIb3DQEJARYVbWlyYWVtMDgubXJAZ21haWwY29tMIIB
IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAz4L+RA3g4JXVcnTc7MsRLYVL
FhIQ9obfRVNu0w4KS0+5+4MfNmjstJc0tc4Dmhy2ZHaAl1agSgW/YjCeadx5poLb
rXn0Qt9QqykCuRGR90+N1wF2A57TYS1V6mfW0gv7414YhJCGlgvLrRXh7eSK6R7
ucB8QzUfke/NkNjMSBgPoOX31Vik++dbDqcTLfD4eouTNYXA06Xpp+Sk+vmrG2BF
5/NH7Cw10HyLq4GJjhXnuT4WbZGBdMsKe06ctc0yQyIstFecIhLzG8DA0M0vBtk
gyYFvvyuIQRmkUdTv+7KANYTQpuZYnzrENH0dJ2I0tZGBIkQtctcM2HI1qygtQID
AQABo0wwSjAJBgNVHRMEAjaAMBEGCWSGAGG+EIBAQQEAEIE8DALBgNVHQ8EBAMC
BAAwHQYDVROlBBYwFAYIKwYBBQUHAWIGCCsGAQUFBwMBMA0GCsGSIb3DQEBCwUA
A4IBAQB7Z9KFPbMzph6ISFVR30gINHsvi0B4575mkX68t+irHUCanBx/GEV08NgK
SxmklFIlcKQN4aNdWLR2gUuxTV0ikwhdEB+7dcmTA6gE0qASVekIGNCAnD/HHiK0
4pZRqJfK7uPlcIDmXJWJ+ZSx57QIIjCwXJ1XMf1WYBAjUKtqo+8zdLZZeVnDUC
osFSaSDe/LC1uo4ZEmKALw8Hz1+/tfWdV6VjF5Rsb4dScwCszKbPd6xNI/zkrom
1Nc7BciYuCWKElARdW16LCNYC+dLDXDrshWVfnJ1GCPXhWteDivS85U59e+aXYj8
s2pBHMx3gJxUCUfIqcN10IA+0ie
-----END CERTIFICATE-----
PKCS7 Data
Shrouded Keybag: PBES2, PBKDF2, AES-256-CBC, Iteration 2048, PRF hmacWithSH
A256
Bag Attributes
```



```

Símbolo del sistema
Bag Attributes
  localKeyID: 4D B4 C2 53 ED 71 28 A3 03 FF 78 9E 6A C8 23 0B 30 36 2B 7A

Key Attributes: <No Attributes>
Enter PEM pass phrase:

Verifying - Enter PEM pass phrase:

-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFNTBfBgkqhkiG9w0BBQ0wUjAxBgkqhkiG9w0BBQwwJAQQWSUBhkMGqFX6sGVf
dGmgWAICCAAwDAYIKoZIhvcNAgkFADAdBglgghkgBZQMEASoEE0AAbboQToCg5PWV
AHIE3GsEggTQ5hckxbuP27k80fUYTKetF8li5+ld+JV6JIV+zmbvPnybl49eeere
HDxSf2JIfoNP26106fbhNXoLaXNZBAqcDUUI4kev5wbEmNCMF34wJJwFwoksmu
aYYz9S1qb0VdZczWhIhbTjL0ZdNdyh8fuWr7hnIjr50/eQIF0i3KnNY7FzVni8hw
0FGaesCv5Lswl2EecCQSRCvySwEwyeLbvfgo43LRLY0WtbMiYQigTy3i0Z7N5LpG
hMe/+02GLt7bZOX64i5ECq2o9brPVYwOLSQz9T1A/RDSr0n5LJp50ahTo8L0L8wC
00Epoac3hr934tPbShhSSqRGIEZ5GFLEKpCyhx2eZ5YNKajNr25GUjTm3ci89mnW
3gUdLbccb7Bv8KNo4XB18/dDl6QZMYPXb0cLtxd0dQDHCEV+EEtr2ZOvJVH3gz89
pVjNNFIRrwSm/A48p+72uyfGbLjCkKdUqCGZfDFTV5vhhP1soX2CLNI0yVTwE/WR
fpGgMH2K604Hv4UhfKJ0UML0U/6LctNCWg16jWCa6snoC7nP4gx4YZMytILVXYz0
Wx6SDkRn0SOKw/KzK77UP6XwXPdtfnmAAb8SF0rjkkGL0x62L64/rXKwXhWpQeKd
p7sD2zUEkaLNP5wfmvgrcUk3RLzqxAkuLEMNq2eDKP574ZjJ7aVkp+zC+qqqth9N
ZAPj0yUtkmQEKpkY+nhh7gaNFAp3yDXBuqJ97zkhQnU4MmW9U9h7Jtt/TTv3LJqX
VJHCzy0q6XIw+ZymnBqYOYRILqHKsyJYC6DcJghZ40H1Xr4fPICZH/wWSRPt9D7p
Zu1mAjUgkuEKourIDxQ4LyqAnGeQeZ9ew4/zgQVNTU5CMK5XoasgFHY2QFUNMuKT
kj41hwvyBXb1/Yqs3edHep5ciKCYhItqfGjo43ei5jfLEI35QXn39dpCYwu1TrSG
s+Fdn5aVHS/GTFvQXy3zJVTZgT00qVzQF89+BZz57W45w0ip193Tv38/OrSdo3Zk
jIsQjysLsr6aWAXvX8yEC+FRTZhrVAo4i+o9hSg7NDWT2KcAa7z0taIfTLuR+urK
v0aBN8r72L/umR0W11iBHyaXQ8e7wIEA7L/jnySy/zWfrHJ7s4aVQGe55MkSVZ/u
KeTyiHHvUQh+y5AATUWgJpr4Vvk0jRmEKsubKQKwa9L3/IXTbiFPJZgvBwD1H8PZ1
ysjZCOMBqDYsyvWczFhQGRTVy9ekkkTXLPxfgIvLYpeu6JG61xHauGkqyz+Ffam
ZIO8mp/dCE0PxxGfHGUqhGna3XrW9MY9eTD8V//1TSPS6LMyrDuVW6sGhCg00M38d
MEHu/EdgzYWGDTQTPEB5c/o8g9nzdxH0Nm6nIfc600Z9w9cayp3mVdGZKif+rB
RKVXu75ICm//mVeJ+0oMGLMqIcZ543MIuQIEJGzQeewKZ8MTALIKvYy+iEgTxQwk
vVfcYgRsrp0ugxxSsaTes/d8mgLSpfRGUjrkJJDK/7vze9TuUXYy1ft4y/A5/y4j
ovjt52FoC31Ej52FEWBzYt0PbsLm1f86y0h4IwnkiqGcKl5gxJI0UhmkgZ89Afe
TSiYgwc2UctNbq2+hbtmzhpbGfIwJLB/oZU+L70QfgkAgu40mLvL5sA=
-----END ENCRYPTED PRIVATE KEY-----

C:\Users\miria\Desktop\Studilet>

```

## CONCLUSIÓN

La creación de claves asimétricas y certificados digitales mediante OpenSSL es un paso fundamental para establecer comunicaciones seguras en entornos digitales. Este proceso comienza con la generación de un par de claves RSA de 2048 bits, asegurando un alto nivel de seguridad para aplicaciones modernas. La clave pública, derivada de la clave privada, facilita el intercambio seguro de información, mientras que los certificados autofirmados y firmados por una autoridad certificadora garantizan la autenticidad de los servicios web.

OpenSSL simplifica significativamente el proceso de obtención de certificados SSL/TLS, ofreciendo una solución eficiente para la generación de certificados autofirmados y la creación de solicitudes de firma de certificados (CSR). Esto añade una capa adicional de seguridad a los servidores web, asegurando que las comunicaciones sean tanto seguras como confiables.



La creación de almacenes de claves PKCS12 agiliza la gestión de claves privadas y certificados, permitiendo su almacenamiento seguro y fácil implementación en entornos de producción. La elección de contraseñas seguras para proteger tanto las claves privadas como los archivos de almacén de claves PKCS12 es crucial, ya que garantiza la integridad y la confidencialidad de la información sensible.

Finalmente, la verificación de la información contenida en los certificados y archivos de almacén de claves proporciona una garantía adicional de que se han generado y almacenado correctamente, asegurando un entorno digital confiable y seguro. En resumen, la implementación de claves asimétricas y certificados digitales con OpenSSL es esencial para establecer y mantener comunicaciones seguras en entornos digitales, garantizando la autenticidad, la integridad y la confidencialidad de las transacciones en línea.

## REFERENCIAS

- OpenSSL Wiki. *Command Line Utilities*. [https://wiki.openssl.org/index.php/Command Line Utilities](https://wiki.openssl.org/index.php/Command_Line_Uutilities)
- Getacert. *Sign a Certificate*. <https://getacert.com/signacert.html>