Crear el par intermedio

Una autoridad de certificación intermedia (CA) es una entidad que puede firmar certificados en nombre de la entidad emisora `root`. La entidad de certificación `root` firma el certificado intermedio, formando una cadena de confianza.

El propósito de usar una CA intermedia es principalmente por seguridad. La clave de `root` puede mantenerse fuera de línea y ser utilizada con la menor frecuencia posible. Si se ve comprometida la clave intermedia, la entidad emisora `root` puede revocar el certificado intermedio y crear un nuevo par criptográfico intermedio.

Preparación del directorio

Los archivos de CA de `root` se mantienen en `/root/ca`. Elija un directorio diferente (`/root/ca/intermediate`) para almacenar los archivos de la CA intermedia.

```
# mkdir /root/ca/intermediate
```

Cree la misma estructura de directorios utilizado para los archivos de la CA `root`. Es conveniente también crear un directorio `csr` para incluir las solicitudes de certificado de firma.

```
# cd /root/ca/intermediate
# mkdir certs crl csr newcerts private
# chmod 700 private
# touch index.txt
# echo 1000 > serial
```

Añada un archivo `crlnumber` en el árbol de directorios de la CA intermedio. `crlnumber` será utilizado para realizar un seguimiento de las listas de revocación de certificados.

```
# echo 1000 > /root/ca/intermediate/crlnumber
```

Copie el archivo de configuración de la CA intermedia del apéndice en `/root/ca/intermediate/openssl.cnf`. Cinco opciones se han cambiado en comparación con el fichero de configuración la CA `root`:

```
[ CA_default ]
dir             = /root/ca/intermediate
private_key     = $dir/private/intermediate.key.pem
certificate     = $dir/certs/intermediate.cert.pem
crl             = $dir/crl/intermediate.crl.pem
policy          = policy_loose
```

Crear la clave intermedia

Cree la clave intermedia (`intermediate.key.pem`). Cifre la clave intermedia con el cifrado AES de 256 bits y una contraseña segura.

```
# cd /root/ca
# openssl genrsa -aes256 \
      -out intermediate/private/intermediate.key.pem 4096

Enter pass phrase for intermediate.key.pem: secretpassword
Verifying - Enter pass phrase for intermediate.key.pem: secretpassword

# chmod 400 intermediate/private/intermediate.key.pem
```

Crear el certificado intermedio

Utilice la clave intermedia para crear una solicitud de firma de certificado (CSR). Los detalles deben adaptarse generalmente a la CA `root`. El nombre común (`Common Name`), sin embargo, tiene que ser diferente.

```
-----------------------------------------------------------------
(¡!) Advertencia

Asegúrese de especificar el archivo de configuración de la CA
intermedia (intermediate/openssl.cnf).
-----------------------------------------------------------------

# cd /root/ca
# openssl req -config intermediate/openssl.cnf -new -sha256 \
      -key intermediate/private/intermediate.key.pem \
      -out intermediate/csr/intermediate.csr.pem

Enter pass phrase for intermediate.key.pem: secretpassword
You are about to be asked to enter information that will be incorporated
into your certificate request.
-----
Country Name (2 letter code) [XX]:GB
State or Province Name []:England
Locality Name []:
Organization Name []:Alice Ltd
Organizational Unit Name []:Alice Ltd Certificate Authority
Common Name []:Alice Ltd Intermediate CA
Email Address []:
```

Para crear un certificado intermedio, utilice la CA de `root` con la extensión `v3_intermediate_ca` para firmar la CSR intermedia. El certificado intermedio debe ser válido por un período más corto que el certificado raíz. Diez años sería razonable.

```
-----------------------------------------------------------------
(¡!) Advertencia
```

```
Esta vez, especifique el fichero de configuración de la CA root
(/root/ca/openssl.cnf).
-----------------------------------------------------------------
```

Hemos introducido:

```
-subj "/C=TheCountry/CN=theCommonName/ST=theState/O=theOrganization"
```

en la orden siguiente, atendiendo a la sugerencia del blog END POINT y su post OpenSSL CSR with Alternative Names one-line de acuerdo con las abreviaturas para las opciones:

```
C => Country
ST => State
L => City
O => Organization
OU => Organization Unit
CN => Common Name (eg: the main domain the certificate should cover)
emailAddress => main administrative point of contact for the certificate
```

las cuales vienen enumeradas en el post. Entonces el modelo de la orden correspondiente sería:

```
# cd /root/ca
# openssl ca -config openssl.cnf -extensions v3_intermediate_ca \
      -days 3650 -notext -md sha256 \
      -in intermediate/csr/intermediate.csr.pem \
      -out intermediate/certs/intermediate.cert.pem \
      -subj "/C=TheCountry/CN=theCommonName/ST=theState/O=theOrganization"

Enter pass phrase for ca.key.pem: secretpassword
Sign the certificate? [y/n]: y
```

En el ejemplo que acarreamos podemos ejecutar:

```
openssl ca -config openssl.cnf -extensions v3_intermediate_ca \
    -days 3650 -notext -md sha256 \
    -in intermediate/csr/intermediate.csr.pem \
    -out intermediate/certs/intermediate.cert.pem \
    -subj "/C=GB/CN=Alice/ST=England/O=Alice Ltd"
```

y obtendremos el siguiente diálogo

```
Using configuration from openssl.cnf
Enter pass phrase for /root/ca/private/ca.key.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
    Serial Number: 4096 (0x1000)
```

```
    Validity
        Not Before: Dec 12 10:47:23 2017 GMT
        Not After : Dec 10 10:47:23 2027 GMT
    Subject:
        countryName               = GB
        stateOrProvinceName       = England
        organizationName          = Alice Ltd
        commonName                = Alice
    X509v3 extensions:
        X509v3 Subject Key Identifier:
            C8:40:2A:BF:80:27:FA:61:D6:CD:23:71:88:44:96:35:76:20:AB:14
        X509v3 Authority Key Identifier:
            keyid:E3:D0:30:7C:FE:EF:79:BA:DC:2D:4A:8A:93:72:D7:69:33:42:84:36

        X509v3 Basic Constraints: critical
            CA:TRUE, pathlen:0
        X509v3 Key Usage: critical
            Digital Signature, Certificate Sign, CRL Sign

Certificate is to be certified until Dec 10 10:47:23 2027 GMT (3650 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

Ahora hemos de actuar sobre los permisos del fichero recién creado:

```
# chmod 444 intermediate/certs/intermediate.cert.pem
```

El archivo `index.txt` es donde la herramienta CA de OpenSSL almacena la base de datos de certificados. No eliminar o editar este archivo manualmente. Ahora debería contener una línea que hace referencia al certificado intermedio.

```
V 250408122707Z 1000 unknown ... /CN=Alice Ltd Intermediate CA
```

y serial debería tener el siguiente contenido

```
1001
```

Comprobar el certificado intermedio

Como lo hicimos para el certificado `root`, compruebe que los detalles del certificado intermedio son correctos.

```
# openssl x509 -noout -text \
      -in intermediate/certs/intermediate.cert.pem
```

Comprobar el certificado intermedio contra el certificado de `root`. Un `OK` indica que la cadena de confianza está intacta.

```
# openssl verify -CAfile certs/ca.cert.pem \
      intermediate/certs/intermediate.cert.pem

intermediate.cert.pem: OK
```

Crear el archivo de cadena de certificados

Cuando una aplicación (por ejemplo, un navegador web) intenta comprobar un certificado firmado por la CA intermedia, debe verificar el certificado intermedio contra el certificado `root`. Para completar la cadena de confianza, cree una cadena de certificados de CA para presentarla a la aplicación.

Para crear la cadena de certificados CA, concatenar los certificados intermedio y `root` juntos. Vamos a utilizar este archivo más adelante para verificar los certificados firmados por la CA intermedia.

```
# cat intermediate/certs/intermediate.cert.pem \
      certs/ca.cert.pem > intermediate/certs/ca-chain.cert.pem
# chmod 444 intermediate/certs/ca-chain.cert.pem

-----------------------------------------------------------------------
(¡!) Nota

Nuestro archivo de cadena de certificado debe incluir el certificado
de root porque ninguna aplicación cliente lo sabe todavía. Una opción
mejor, especialmente si usted está administrando una intranet, es
instalar el certificado root en cada cliente que necesite
conectarse. En ese caso, el archivo de la cadena sólo tiene que
contener su certificado intermedio.
-----------------------------------------------------------------------
```