Introducción

- En determinadas ocasiones es necesaria una conexión segura:
  - Petición de datos de una tarjeta bancaria en comercio electrónico.
  - Petición del usuario/contraseña en el acceso al correo electrónico.
- Necesidad de comunicaciones cifradas.
- Dos tipos de cifrado:
  - Simétrico.
  - Par de claves pública/privada.
- Los servicios seguros utilizan claves pública/privada.

Creación de un par de claves pública y privada (I)

- Pueden generarse protegidas con contraseña o sin contraseña.

```
openssl genrsa [-des|-des3|-idea] [<tamaño>]
```

- Protegidas con contraseña:

```
openssl genrsa -des3 2048 > clave.key
```

- Protegidas sin contraseña:

```
openssl genrsa 2048 > clave.key
```

- Utilizar contraseña es más seguro, pero requiere su introducción para permitir arrancar el servicio que utilizará el certificado creado con las claves.

Creación de un par de claves pública y privada (II)

- Puede extraerse la parte pública ejecutando el comando:

```
openssl rsa -in clave.key -pubout -out
  publica.key
```

Creación de un certificado

- Dos tipos de certificados:
  - Firmados por una autoridad de certificación (CA).
  - Autofirmados.
- En los firmados por CA, la CA garantiza que ese certificado es autentico, corresponde al ordenador, etc.
- En los autofirmados es el propio ordenador el que “garantiza” su autenticidad.
- Los autofirmados son gratuitos y los firmados por una CA no.

Creación de un certificado para su firma por CA

- Se crea utilizando el comando:

```
openssl req -new -key clave.key -out servidor.csr
```

- Donde:
  - req indica que se solicita un certificado X.509 para ser firmado.
  - clave.pem es el fichero con las claves pública/privada generado con anterioridad.
  - servidor.csr es el fichero que contendrá la salida.
- Se solicita información sobre el país, provincia, localidad, organización, unidad, ordenador y correo del administrador.
- El fichero servidor.csr debe enviarse a una CA para que lo firme y nos devuelva un fichero firmado servidor.crt.

Creación de un certificado autofirmado

- Se crea utilizando el comando:

```
openssl req -new -key clave.key -x509 -days 365
   -out servidor.crt
```

- Donde:
  - req indica que se solicita un certificado X.509 para ser firmado.
  - clave.pem es el fichero con las claves pública/privada generado con anterioridad.
  - x509 indica que firme el certificado como X.509 valido.
  - servidor.crt es el fichero que contiene la salida.
- Se solicita la misma información que antes.

Instalación de un certificado en el servidor

- Tan solo es necesario copiar los ficheros clave.key y servidor.crt en la ubicaciones adecuadas y con los nombres adecuados.
- Ejemplos:
  - Servidor web Apache:
    - clave.key como /etc/pki/tls/private/localhost.key
    - servidor.pem como /etc/pki/tls/certs/localhost.crt
  - Servidor de entrega final de correo (POP3s e IMAPs).
    - clave.key como /etc/pki/dovecot/private/dovecot.pem
    - servidor.pem como /etc/pki/dovecot/dovecot.pem

Instalación de un certificado en el cliente

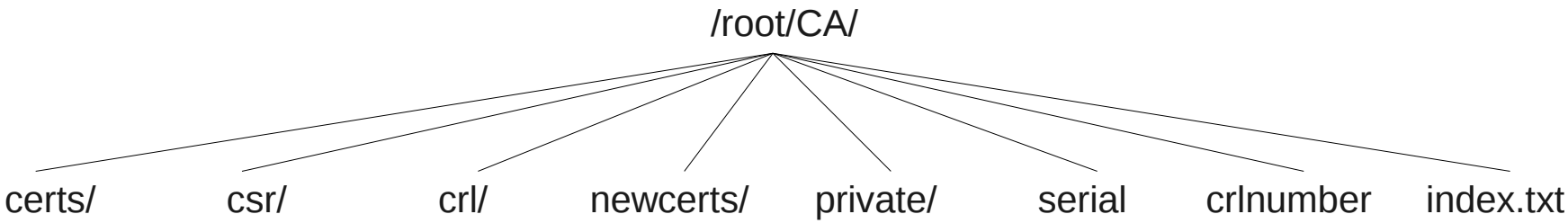
- Si el certificado ha sido firmado por una CA reconocida por nuestro cliente, nuestro certificado será reconocido automáticamente.
- Si el certificado es autofirmado o firmado por una CA no reconocida, deberemos instalarlo.
  - La instalación depende del sistema operativo y programa, pero es sencilla.
  - Una vez instalado, el certificado será valido como si hubiera sido firmado por una CA reconocida.

Creación de una autoridad de certificación (I)

- En ocasiones es necesario crear una autoridad de certificación para:
  - Firmar certificados como una CA pero de forma gratuita.
  - Expedir certificados para varios ordenadores y que los clientes importen solo nuestra CA como CA reconocida y no los certificados de cada ordenador.
- Una autoridad de certificación es cualquiera de quién nos fiemos o nuestros programas se fíen.
- Ejemplos de CA incluidos en el Microsoft Internet Explorer:
  - Autoridad Certificadora del Colegio Nacional de Correduría Pública Mexicana, A.C.
  - Deutsche Telekom Root CA 1
  - http://www.valicert.com
  - Microsoft Root Authority
  - Verisign Trust Network

Creación de una autoridad de certificación (II)

- Es necesario crear el siguiente árbol de directorios y ficheros:



- certs, csr, crl, newcerts y private son directorios.
- serial contiene el número de serie del siguiente certificado que firmemos, inicialmente 01\n.
- crlnumber contiene el número de serie de la siguiente lista de certificados revocados, inicialmente 01\n.
- index.txt es una “base de datos” con los certificados firmados.

Creación de una autoridad de certificación (III)

Directorio	Descripción
/root/CA/certs	Directorio donde se almacenaran los certificados ya firmados y enviados a los clientes.
/root/CA/newcerts	Directorio donde se guardan los certificados que acaban de ser firmados.
/root/CA/crl	(Certificate Revokation List). Directorio donde los certificados revocados son almacenados.
/root/CA/csr	(Certificate Signing Request). Directorio donde son almacenadas las peticiones de certificados pendientes de firmar.
/root/CA/private	Directorio donde se almacenaran la clave privada de la autoridad de certificación, así como las demás claves privadas que sean generadas para los diferentes servicios.

Creación de una autoridad de certificación (IV)

- Es necesario configurar los valores de la CA copiando /etc/pki/tls/openssl.cnf en otro fichero y modificando las secciones [CA_default], [req_distinguished_name] y [req].

Creación de una autoridad de certificación (V)

Sección [CA_default]:

dir	/root/CA	Directorio raíz de la autoridad de certificación.
certs	$dir/certs	Directorio donde se almacenaran los certificados ya firmados.
crl_dir	$dir/crl	Directorio donde los certificados revocados son almacenados.
database	$dir/index.txt	Archivo con la base de datos de los certificados.
new_certs_dir	$dir/newcerts	Directorio donde se guardan los certificados que acaban de ser firmados.
certificate	$dir/irtic.pem	Archivo con la clave pública de la autoridad de certificación.
serial	$dir/serial	Archivo con el número de serie de los certificados.

Creación de una autoridad de certificación (VI)

crlnumber	$dir/crlnumber	Archivo con el número de serie de revocación. Si se desea un funcionamiento como en versiones antiguas de openssl puede comentarse esta línea.
crl	$dir/crl.pem	Lista de los certificados revocados.
private_key	$dir/private/irtic.key	Archivo con la clave privada de la autoridad de certificación.
RANDFILE	$dir/private/.rand	Archivo con el número aleatorio privado.
x509_extensions	usr_cert	Extensiones que han de añadirse al certificado.
name_opt	ca_default	Formato en que se mostrará el nombre del certificado antes de que sea firmado.
cert_opt	ca_default	Formato en que se mostrará un certificado antes de que sea firmado.
default_days	365	Días por defecto para los que se firma el archivo.
default_crl_days	30	Días por defecto en que debe ser actualizada la lista de certificados revocados de esta autoridad de certificación.
default_md	default	Compendio de mensaje utilizado, por defecto es md5 (valor default).
preserve	no	Indica si se ha de mantener o no el orden Domain Name.
policy	policy_match	Política por defecto a aplicar si no se especifica ninguna.

Creación de una autoridad de certificación (VII)

Sección [req_distinguished_name]:

Variable	Valor	Descripción
countryName_default	ES	País de emisión del certificado
stateOrProvinceName_default	Valencia	Estado o provincia de emisión
localityName_default	Paterna	Localidad de emisión del certificado
0.organizationName_default	Universitat de Valencia	Nombre de la organización
organizationalUnitName_default	IRTIC	Nombre de la sección
commonName_default	Autoridad de Certificacion del IRTIC	Nombre de la autoridad
emailAddress_default	webmaster@irtic.uv.es	Dirección de correo del responsable de la autoridad de certificación

Creación de una autoridad de certificación (VIII)

Sección [req]:

Variable	Valor por defecto	Descripción
default_bits	2048	Bits por defecto de la clave privada.
default_md	sha1	Compendio de mensaje usado por defecto.

Creación de una autoridad de certificación (IX)

- Creación del certificado de la CA:

```
openssl req -new -x509 -days 3650 –config
  /root/CA/irtic.cnf -keyout
  /root/CA/private/irtic.key -out /root/CA/irtic.pem
```

- Comprobación del certificado de la CA:

```
openssl rsa -in /root/CA/private/irtic.key –text
openssl x509 -in /root/CA/irtic.pem –text
openssl x509 -in /root/CA/irtic.pem -purpose
```

Distribución de la acreditación de la CA

- Debe hacerse público, mediante un servidor:
  - Web.
  - FTP.
  - Etc.

el fichero irtic.pem.

- En algunos casos (Windows) debe ser llamado irtic.cer para que sea reconocido como certificado.

Firma de un certificado por una CA

- Deseamos firmar el certificado /root/CA/csr/servidor.csr.
- Examinamos sus datos:

```
openssl req -in /root/CA/csr/servidor.csr -text
```

- Si es valido lo firmamos:

```
openssl ca -config /root/CA/irtic.cnf -in
   /root/CA/csr/servidor.csr –verbose
```

- El Common Name debe corresponder con el ordenador para el que se firma.
- Si firmamos un certificado que no es de nuestra organización:

```
openssl ca -config /root/CA/irtic.cnf -in
/root/CA/csr/servidor.csr -verbose -policy policy_anything
```

Revocación de un certificado por una CA

- Deseamos revocar el certificado `/root/CA/certs/servidor.crt.`
- Revocamos el certificado:

```
openssl ca –config /root/CA/irtic.cnf -revoke
  /root/CA/certs/servidor.crt
```

- Actualizamos la lista de certificados revocados:

```
openssl ca –config /root/CA/Robotica.cnf
  -gencrl -out /root/CA/crl/crl.pem
```

- Comprobamos la lista de certificados revocados:

```
openssl crl -in /root/CA/crl/crl.pem -text
```