Legacy OpenSSL

Uso de certificados digitales con cifrado 40BitRC2 en OpenSSL 3 y superior

Anonymous · 03/11/2026

Legacy OpenSSL

Desde OpenSSL 3 (y también en OpenSSL 4) ciertos algoritmos que antes eran estándar se consideran inseguros y se encuentran en el proveedor *legacy*, que no está activo por defecto. Entre ellos está el cifrado `40BitRC2` de los archivos PKCS#12 (`.pfx` o `.p12`), que es el que utilizan muchos certificados digitales emitidos para el SII.

Cuando el proveedor *legacy* no está activo, PHP no puede leer esos certificados y falla con un error como:

```
It was not possible to read the digital certificate data.
Unsupported encryption algorithm or method (Error #0308010C).
OpenSSL: error:030000A0:digital envelope routines::unknown cipher
```

Revisar si el certificado es afectado

Ejecutar:

```
openssl pkcs12 -in certificadoDigital.pfx -info -nokeys -nocerts
```

Si en la respuesta está presente `PKCS7 Encrypted data: pbeWithSHA1And40BitRC2-CBC`, el certificado está cifrado con `40BitRC2`. Si el proveedor *legacy* no está activo, el comando termina además con el error `unknown cipher`.

Revisar la versión y la configuración de OpenSSL que usa PHP

PHP puede usar una versión de OpenSSL distinta a la del comando `openssl` del sistema. Para ver la que usa PHP:

```
php -r 'echo OPENSSL_VERSION_TEXT, PHP_EOL;'
```

Para ver el directorio de configuración y el de módulos de la instalación de OpenSSL que usa el comando `openssl`:

```
openssl version -d
openssl version -m
```

El archivo de configuración es `openssl.cnf` dentro del directorio `OPENSSLDIR`. El módulo del proveedor *legacy* es el archivo `legacy.so` (Linux) o `legacy.dylib` (macOS) dentro del directorio `MODULESDIR`. Ese archivo debe existir; en Debian y Ubuntu viene con el paquete `libssl3`.

Ubicaciones habituales del `openssl.cnf`:

Sistema	Archivo
Debian, Ubuntu (y imágenes Docker basadas en ellos)	`/etc/ssl/openssl.cnf`
RHEL, Fedora, Rocky, Alma	`/etc/pki/tls/openssl.cnf`
macOS con Homebrew	`/opt/homebrew/etc/openssl@3/openssl.cnf` o `/opt/homebrew/etc/openssl@4/openssl.cnf`, según la versión con la que se compiló PHP

Activar el proveedor legacy

Hay dos formas de activarlo. Ambas usan la misma configuración.

Opción 1: archivo de configuración propio (recomendada)

Crear un archivo, por ejemplo `openssl-legacy.cnf`, con el contenido:

```
openssl_conf = openssl_init

[openssl_init]
providers = provider_sect

[provider_sect]
default = default_sect
legacy = legacy_sect

[default_sect]
activate = 1

[legacy_sect]
activate = 1
```

Y apuntar a él con la variable de entorno `OPENSSL_CONF` en el proceso que ejecuta PHP:

```
export OPENSSL_CONF=/ruta/a/openssl-legacy.cnf
```

Esta opción no modifica los archivos del sistema, por lo que no se pierde cuando el sistema o Homebrew actualizan OpenSSL. Para otros procesos:

- **PHP-FPM**: definir la variable en el pool con `env[OPENSSL_CONF] = /ruta/a/openssl-legacy.cnf` (y `clear_env = no` si se usa la herencia del entorno).
- **Docker**: definir `ENV OPENSSL_CONF=/ruta/a/openssl-legacy.cnf` en el `Dockerfile` o en `environment:` de `docker-compose.yml`.
- **PHPUnit**: definirla en el entorno antes de ejecutar los tests.

Opción 2: editar el `openssl.cnf` del sistema

Editar el `openssl.cnf` que corresponda (ver la tabla). Estos archivos ya traen las secciones `[openssl_init]`, `[provider_sect]` y `[default_sect]`, por lo que no se debe copiar el bloque completo, ya que duplicaría secciones. Hay que ajustar las existentes:

1. En la sección `[provider_sect]` agregar `legacy = legacy_sect`, dejando `default = default_sect`.

2. En la sección `[default_sect]` dejar `activate = 1` sin comentar.

3. Agregar al final de esos bloques la sección:

   ```
   [legacy_sect]
   activate = 1
   ```

Si la sección `[default_sect]` no tiene `activate = 1`, el proveedor *default* queda inactivo en cuanto se activa otro, y OpenSSL deja de funcionar con los algoritmos normales.

Los archivos del sistema pueden ser sobrescritos al actualizar OpenSSL o Homebrew; después de una actualización hay que revisar que el cambio siga presente.

Verificar

Comprobar que el proveedor está activo:

```
openssl list -providers
```

Deben aparecer `default` y `legacy`. Luego repetir el comando de la sección Revisar si el certificado es afectado: ya no debe terminar con error. Para comprobar PHP, que puede usar una configuración

distinta a la del comando `openssl`, cargar el certificado desde PHP con la misma variable `OPENSSL_CONF` definida.

Referencias

- https://wiki.openssl.org/index.php/OpenSSL_3.0#Providers
- https://stackoverflow.com/a/74416513
- https://stackoverflow.com/a/72600724

Última actualización el 03/11/2026