Recomendaciones para realizar implementaciones de sistemas con los certificados electrónicos emitidos por la Dirección Nacional de Firma Electrónica de Panamá

Autor: Javier Batista
Versión del documento: 0.1

I. Introducción

La Firma Electrónica Calificada emitida por la Dirección Nacional de Firma Electrónica de Panamá cuenta con validez legal inmediata gracias a la Ley Nº 82 del 9 de Noviembre de 2012, la cual le otorga al Registro Público de Panamá atribuciones de Autoridad Registradora y Certificadora Raíz de firma electrónica para la República de Panamá; por lo tanto, el uso de los certificados electrónicos emitidos por la Dirección para firmar la documentación procesada por sistemas de terceros, es con el objetivo de brindarle valor legal a dichos documentos digitales independiente de su formato.

La presenta guía cuenta con recomendaciones básicas dirigidas principalmente al personal técnico de las Instituciones del Estado y las empresas privadas (programadores y soporte técnico) para que puedan emplear la Firma Electrónica Calificada de Panamá en sus plataformas. Se debe comprender que la Dirección Nacional de Firma Electrónica sólo brinda los certificados electrónicos como tal y no proporciona actualmente, un software propio de firma que pueda ser instalado por los usuarios; por ese motivo, se han elaborado guías de configuración en el sitio web de la Dirección para brindarle ayuda a aquellos usuarios que no tienen acceso a plataformas de firma, con el objetivo de que puedan emplear algún programa gratuito de un tercero (tal como el Adobe Reader) para que puedan firmar sus documentos. En el caso de la Firma Electrónica en la Nube (la cual cuenta con un API que será mencionado posteriormente en esta guía), le proporcionamos a nuestros usuarios un portal web para que puedan firmar sus documentos, principalmente para aquellas personas que no cuentan con la posibilidad de realizar implementaciones en sistemas de terceros ni tendrán acceso a plataformas similares. Aquellas Instituciones del Estado o empresas privadas que adquieran nuestros certificados electrónicos son libres de emplearlos en sus sistemas, donde el desarrollo necesario cae por cuenta de los programadores de dichas organizaciones.

II. Desarrollos utilizando las tarjetas inteligentes.

Cualquier desarrollo que se realice para una plataforma en particular requerirá la compatibilidad para leer los certificados electrónicos que están en el chip de las tarjetas junto a sus claves públicas y privadas, es decir, que debería ser compatible con el estándar PKCS #11. Los drivers para leer las tarjetas y algunos manuales para configurar programas como el Adobe Reader, los podrán revisar en esta dirección:
https://www.firmaelectronica.gob.pa/configuracion-firma-electronica.html

Actualmente no contamos con manuales específicos para programadores que realicen desarrollos con las tarjetas, ya que, al ser un estándar criptográfico utilizado en todo el mundo, dependería del lenguaje de programación y del conocimiento de los desarrolladores para así realizar implementaciones propias. La información que necesitarían de nuestra parte serían las rutas de los certificados raíces, CRL, OCSP y TSA, las cuales se pueden obtener de los manuales de configuración mencionados o preferiblemente desde los mismos certificados que están en las tarjetas y que se pueden exportar con el driver que proporcionamos (TokenAdmin), por lo tanto, recomendaríamos que los programadores adquirieran una tarjeta de firma electrónica con el perfil deseado (tal como se detalla en la sección de requisitos: https://www.firmaelectronica.gob.pa/requisitos.html en caso de que no tengan una) para que hagan las pruebas técnicas necesarias.

Como referencia les comparto los datos específicos que necesitarían los desarrolladores:

Ruta de los certificados raíces (los cuales se pueden descargar en la siguiente dirección: http://www.pki.gob.pa/cert.htm)

http://www.pki.gob.pa/cacerts/caraiz.crt

http://www.pki.gob.pa/cacerts/cagob.crt

http://www.pki.gob.pa/cacerts/capc2.crt

La lista de revocación de certificados CRL (http://www.pki.gob.pa/crl.htm):

http://www.pki.gob.pa/crls/caraiz.crl

http://www.pki.gob.pa/crls/cagob.crl

http://www.pki.gob.pa/crls/capc2.crl

Dirección de la Autoridad de Validación (Protocolo de Verificación de Certificados en Línea):

http://ocsp.pki.gob.pa/

Dirección de la Autoridad de Sellado de Tiempo:

http://tsp.pki.gob.pa/

Si van a firmar documentos PDF (que es lo más común), es recomendable que le incluyan el sellado de tiempo (timestamping) para que la hora de la firma venga de un servidor de tiempo real y no de la computadora del usuario. También debería tener LTV (Validación a largo plazo) que ayudaría a que el documento no se muestre como inválido cuando la firma del usuario expire o se revoque. Existen diversas librerías de programación que se pueden emplear en lenguajes como C# o Java y en el caso particular, PDFBox https://pdfbox.apache.org/ en Java serviría para manipular documentos PDF, agregar la firma con timestamping y LTV usando PKCS #11 para leer de las tarjetas, ya que se apoya en otras librerías como Bouncy Castle (https://www.bouncycastle.org/), donde pueden revisar específicamente estos ejemplos:
https://svn.apache.org/viewvc/pdfbox/trunk/examples/src/main/java/org/apache/pdfbox/examples/signature/

Para más información sobre la programación en Java con PKCS#11, pueden consultar el siguiente enlace oficial de Oracle:

https://docs.oracle.com/javase/8/docs/technotes/guides/security/p11guide.html

La ruta del DLL para leer de las tarjetas una vez que se instala el driver de SafeSign con el programa TokenAdmin, la cual puede ser usada a nivel de programación, es la misma que se muestra en los manuales del Adobe Reader que tenemos en nuestra página web:

Windows: C:\Windows\System32\aetpkss1.dll
macOS: /Applications/tokenadmin.app/Contents/Frameworks/libaetpkss.dylib.3.8.4.1

En estos momentos, no contamos con versiones actualizadas del driver para Linux, por lo tanto, el uso de las tarjetas de firma se limitaría a Windows (donde se incluye también la versión 11 de ese sistema operativo) y macOS.

III. Programación de la Firma Electrónica en la Nube:

El uso de la Firma Electrónica en la Nube se encuentra reglamentado con los perfiles de persona natural y funcionario público, tal como se puede apreciar en el enlace de la Gaceta Oficial https://www.gacetaoficial.gob.pa/pdfTemp/29715/96498.pdf y en la sección de Normativa Aplicable de la página web de la Dirección (Resolución No. DG-033-2023 del 26 de enero de 2023): https://www.firmaelectronica.gob.pa/legislacion-nacional.html.

Actualmente estamos en una fase inicial de pruebas para utilizar la firma en la nube en temas de desarrollo con sistemas de terceros como una alternativa legal a las tarjetas (no las reemplaza y si un usuario tiene tarjeta, tendría que adquirir un certificado electrónico en la nube que tiene el mismo costo de $50 dólares por dos años). Los usuarios que adquieren los certificados de la Nube y que no tienen la posibilidad de realizar implementaciones en

sistemas de terceros ni tendrán acceso a dichas plataformas, podrán emplear de manera gratuita nuestro sistema web para firmar documentos PDF.

Dirección del Portal Web **Firma PDF** para Personas Naturales:
https://firma.pki.gob.pa/ciudadano/

Dirección del Portal Web **Firma PDF** para Funcionarios Públicos:
https://firma.pki.gob.pa/gobierno/

En el siguiente enlace podrán descargar la documentación oficial con algunos ejemplos para la integración de la firma electrónica en la nube con el API que ofrecemos:

Java:

https://www.firmaelectronica.gob.pa/api/IntegracionSafecertRuntime.zip

Dentro del archivo ZIP podrán encontrar el siguiente contenido:

- IntegracionSafecertRuntime_2022Diciembre.pdf
  PDF con la presentación.

- SafeCert 2.5.02 - API Pasarela:
  - Documentación - API_Gateway.
    Javadoc del API Pasarela.
  - Documentación – Manuales.
    Manual de integración del API de Pasarela.
  - Safecert-Gateway – API.
    Librerías del API Pasarela y dependencias.

- SIAVAL 7.4.00 - API StandAlone:
  - Documentación - API_StandAlone.
    Javadoc del API StandAlone.
  - Documentación - Manuales.
    Manual de integración del API StandAlone.
  - TspStandAlone.
    Librerías del API Standalone y dependencias.

- Ejemplos:
  - Ejemplos varios de integraciones con nuestras plataformas de firma. El ejemplo de firma centralizada utilizando el API de Pasarela y Standalone es:
    es.sia.example.signature.optimized.GatewayOptimizedSignature

Para configurar el proyecto que está en la carpeta **Ejemplos**, se puede utilizar la última versión de Eclipse e importar dicho directorio. Los ejemplos fueron ejecutados con la versión más reciente de Java 8 y se deben modificar los siguientes archivos para garantizar su funcionamiento:

Carpeta **config** / archivo **gatewayapi.properties**:

URL_GATEWAY: Dirección URL de la pasarela en pre-producción o producción, ejemplo:
URL_GATEWAY = https://gatewayws.pre.pki.gob.pa/rss-gateway/HESS/OperationGateWayRSS
URL_GATEWAY = https://gatewayws.pki.gob.pa/rss-gateway/HESS/OperationGateWayRSS

AUTH_STORE: Ruta del certificado P12 de acceso a la pasarela en pre-producción o producción, el cual es suministrado por la Dirección Nacional de Firma Electrónica a nombre de la Institución o Empresa que estará realizando los desarrollos con la Firma Electrónica en la Nube. En el siguiente ejemplo se está inicializando la variable AUTH_STORE con la ruta completa del certificado p12 copiado en la carpeta resources del proyecto en Eclipse:
AUTH_STORE=/ruta/completa/eclipse-workspace/Ejemplos/resources/certificado_acceso_PRE.p12

AUTH_STORE_PASS: contraseña para el certificado P12 mencionado en el punto anterior y que brindará acceso a la pasarela. Ejemplo: AUTH_STORE_PASS=Pa$$w0rd01

Paquetes:

es.sia.example.gateway.auth
GateWayAuthentication.java: Para ejecutar el ejemplo de autenticación de la clase GateWayAuthentication, se debe ubicar el método **doAuth()** e inicializar el objeto **String userID** con el ID del certificado de la nube que se desea verificar, el cual sería el correo electrónico del usuario en mayúscula, por ejemplo: userID = "EMAIL@CORREO.PA"; y al ejecutar el código, se generará un enlace único en la consola del Eclipse, el cual se podrá seleccionar, copiar y pegar en el navegador web para así autorizar el proceso de autenticación donde si es correcto, se dirigirá al enlace de OK y regresando a la consola, se presionará la tecla Enter para que se visualicen los datos del usuario, indicando así que la autenticación se realizó correctamente. Para los ejemplos se están utilizando unas direcciones URL de OK y Error demo que apuntan al sitio web de Firma Electrónica y que podrán ser cambiadas por las direcciones que establezcan los programadores (para mayor referencia, ver los objetos de tipo **String** llamados **successURL** y **errorURL** en la clase **GateWayOptimizedSignatureUtils** en el paquete **es.sia.example.utils.optimized**).

es.sia.example.signature.optimized
GatewayOptimizedSignature.java: Este ejemplo realiza el proceso de firmar un documento PDF (que está ubicado en la carpeta **resources**), donde se agrega el sellado de tiempo y LTV (Long-Term Validation) para que el documento se muestre siempre como válido (teniendo en

cuenta que los certificados raíces de la Dirección Nacional de Firma Electrónica deben estar importados en el programa que visualiza el PDF, tal como Adobe Reader). Para ejecutar el ejemplo, se debe ubicar el método **doSign** e inicializar el objeto **String userID** con el ID del certificado de la nube que se desea utilizar, el cual sería el correo electrónico del usuario en mayúscula, por ejemplo: userID = "EMAIL@CORREO.PA"; y al ejecutar el código, se generará un enlace único en la consola del Eclipse, el cual se podrá seleccionar, copiar y pegar en el navegador web para así autorizar el proceso de firma, donde si es correcto, se dirigirá al enlace de OK y regresando a la consola, se presionará la tecla Enter para que se pueda firmar el documento PDF y al finalizar, el archivo quedará guardado en la carpeta **output** con el nombre original y la fecha de la firma concatenada al inicio (para propósitos de prueba) y así tener un documento único con cada firma ejecutada para realizar las verificaciones correspondientes y no sobrescribir el último documento firmado en caso de ejecutar el código de firma en más de una ocasión.

Visual C# .NET:

En caso de que utilicen programación con .NET podrán descargar el siguiente archivo .zip con ejemplos:

https://www.firmaelectronica.gob.pa/api/NET.zip

El archivo contiene:

Librerías de integración de Pasarela de Firma en .NET.
Documentación del API de Integración.
Ejemplo en C# de integración.

*.nupkg
Son los paquetes NuGet que deben instalar para disponer de las API de integración de la Pasarela de Firma Centralizada.

*.PDF
Documentación de integración de las APIs.

Contiene:
- Información de instalación de paquetes NuGet.
- Descripción de las opciones de configuración.
- Descripción del API.
- Ejemplos de operaciones.

o Autenticación con Pasarela de Firma.

o Firma con Pasarela de Firma.

o Firma con Sello Electrónico/Certificado Local.

Gateway-api-net-TesterConsole.zip: Contiene un Proyecto C# que es básicamente un ejemplo de integración 100% funcional, con lo que pueden usarlo como base para las pruebas iniciales. El ejemplo contiene operaciones de autenticación/firma con Pasarela de Firma, sello electrónico/firma con certificado local y ambas operaciones combinadas. Cada ejemplo está acompañado de comentarios para facilitar la comprensión del código.

Comentarios adicionales con respecto al uso de los OTPs en varias operaciones.

Cada vez que se firma un documento con un certificado electrónico de la Nube, el usuario aparte de escribir la contraseña personal que definió el día de la emisión, tendrá que utilizar un código OTP que recibirá por SMS en su número de celular personal, por lo tanto, el comportamiento por defecto por cada firma es emplear un OTP distinto; sin embargo, existe la posibilidad de mantener en caché las contraseñas y OTPs. Las contraseñas se pueden cachear durante ciertos periodos de tiempo (por ejemplo: 2 horas) y de igual forma, también hay un mecanismo similar para OTPs. En el caso de la contraseña / PIN del certificado, es posible cachear su valor y el control de esta caché de contraseña se establece mediante una cookie de sesión del navegador. En el caso de la OTP/2FA/SFDA, es posible cachear su valor y el control NO se basa en esa cookie de sesión (lo gestiona directamente Safecert).

Es posible establecer un periodo de tiempo (2 horas) o un número máximo de firmas (solicito 1 código OTP, y me permite lanzar entre 0 y N firmas, si tras varias transacciones de firma se superan las N firmas, en la siguiente transacción se solicitará de nuevo un código OTP que me permitirá realizar otras N firmas más sin requerirme introducir un código OTP). De esta forma, se podría lograr utilizar la firma en la nube para lote de documentos empleando un sólo código OTP que, por el momento, sólo se reciben por medio de SMS.

Acceso al entorno de Pre-Producción

Para probar los ejemplos de la Firma en la Nube, se requiere un acceso a nuestro ambiente de pre-producción por medio de un certificado P12 que emitimos a nombre de la Institución del Gobierno que realizará las pruebas, por lo tanto, si al personal técnico de una Entidad del Estado le parece más viable la utilización de los certificados en la Nube, en vez del uso de las tarjetas, le deben informar por correo electrónico a la Dirección Nacional de Firma Electrónica para solicitar un certificado de prueba de acceso para desarrolladores (en estos momentos sólo se le está brindando dicho acceso a Entidades del Estado que estén realizando implementaciones de firma para el Gobierno).

IV. Uso de otros programas para firmar documentos

Aparte de Adobe, si el software que manipula documentos PDF permite el uso de firma electrónica (PAdES), se puede emplear con nuestros certificados sin problemas, por ejemplo: Nitro PDF, Master PDF Editor, XolidoSign, entre otros (se deben tener en cuenta sus limitantes, ya que, si un programa no permite agregar características como el sellado de tiempo ni LTV, no se recomendaría su uso en caso de requerir dichas opciones). Incluso se pueden firmar otros formatos de documentos, por ejemplo: en el caso de Microsoft Office para Windows, se pueden emplear las tarjetas para firmar documentos en Word, Excel, PowerPoint y hasta correos electrónicos en Outlook. De igual forma, una plataforma que utilice el conjunto de extensiones CAdES y XAdES permitiría aplicar la firma electrónica en distintos formatos de archivos, donde se tendría el archivo original, el hash cifrado con la ayuda de la clave privada del firmante y el certificado digital con la clave pública para validar dicha firma.

En el caso de XolidoSign para Windows se pueden firmar múltiples documentos colocando el número pin de la tarjeta de firma una sola vez. Para configurar este programa se pueden seguir estas instrucciones:

https://www.firmaelectronica.gob.pa/manuales/Xolido-Sign.zip

El archivo zip tiene algunas imágenes que pueden servir de guía para configurar XolidoSign con el sellado de tiempo, LTV y la imagen de la firma.

Enlace de descarga:

https://www.xolido.com/lang/xolidosign/modulo/xolidosign-desktop/descargar/

Manual de configuración oficial de XolidoSign que puede servir como referencia (si aplican las configuraciones de las imágenes en Xolido-Sign.zip no es necesario estudiar en detalle el manual):

https://www.xolido.com/extras/xolidosign/2020/Manual_XolidoSign_V_2_2_1_es_firmado_por_XOLIDO_SYSTEMS.pdf

Manual para validar documentos PDF firmados con nuestras tarjetas.

https://www.firmaelectronica.gob.pa/manuales/Manual-Validacion-Adobe-Reader.pdf

Dirección URL de la Autoridad de Sellado de Tiempo de Panamá que se debe configurar en XolidoSign:

http://tsp.pki.gob.pa/

V. Conclusión

Con las recomendaciones brindadas en el presente documento (el cual es sólo una breve guía), los programadores y el personal técnico de las diversas Instituciones del Estado e incluso empresas privadas tendrán una idea general de cómo podrían realizar desarrollos en sus sistemas o plataformas y llevar a cabo las implementaciones necesarias para así emplear el uso de la Firma Electrónica Calificada de Panamá en sus organizaciones, optimizando así los procesos que actualmente se realizan de forma manual.

Es importante tener en cuenta que los desarrollos realizados en las plataformas de terceros son responsabilidad de los programadores contratados por las Instituciones o Empresas privadas, los cuales deberían tener los conocimientos adecuados sobre las normas internacionales que rigen todos los temas relacionados a la firma electrónica, las buenas prácticas y uso de los certificados de manera segura, entre otras disposiciones. La Dirección Nacional de Firma Electrónica es sólo una Autoridad Certificadora que se encarga de emitir de manera correcta y segura, los certificados electrónicos que son utilizados por los usuarios para firmar sus documentos electrónicos y, por ende, no tiene responsabilidad sobre los sistemas desarrollados por terceros que empleen los certificados de autenticación y firma de los usuarios.

Firmado por: JAVIER EDUARDO BATISTA MARTINEZ
Fecha: 2023.10.20 10:16:55 -05:00