XLIII CLEI | 46 JAIIO
Conferencia Latinoamericana de Informática | Jornadas Argentinas de Informática
4 al 8 de SEPTIEMBRE de 2017
UTN-FRC - Córdoba Argentina

INVOCACION POR PROTOCOLO COMO PLATAFORMA DE SERVICIOS DE FIRMA DIGITAL

(Modalidad INICIATIVAS DE ÉXITO)

Lic. Mauricio Decima[1]
Ing. Gaston Terdoslavich[2]

XLIII CLEI/46JAIIO - Jornadas Argentinas de Informática
SIE 2017 - 11º Simposio de Informática en el Estado

RESUMEN: Los distintos avances tecnológicos provocaron que la solución de firma digital que veníamos utilizando mediante el uso de Applets ya no sean viables en el presente, debido principalmente a que por distintos problemas de seguridad la mayoría de los browsers (Chrome, Edge) ya no soportan el uso de los mismos. Dicha situación nos llevó a tener que replantear la arquitectura para poder dar soporte a la firma digital en aplicaciones web, tarea que no resulta fácil ya que no es posible a priori acceder a los repositorios de certificados del sistema operativo o SmartCards/Tokens desde el browser. De todas las alternativas analizadas la invocación por protocolo es la que mejor aplica al problema planteado, ya que nos permitió tener una solución ágil, segura y simple de acceder. Dicha arquitectura no solo nos permite resolver el problema puntual de firma digital, sino que nos dejó las bases para seguir incorporando de funcionalidad criptográfica como ser encripcion/desencripcion de información con SmartCards, firma de distintos tipos de documentos: Pdf, Xml, binarios.

[1]Mauricio Décima - Gerente de Gestión de Proyectos y Servicios – Gerencia General de Tecnologías e Innovación – Subdirección de Administración y Tecnología - ARBA

mauricio.decima@arba.gov.ar

[2]Gastón Terdoslavich - Departamento Seguridad Lógica ARBA – Gerencia de Gestión de Proyectos y Servicios – Gerencia General de Tecnologías e Innovación – Subdirección de Administración y Tecnología - ARBA

gterdoslavich@arba.gov.ar

XLIII CLEI Conferencia Latinoamericana de Informática | 46 JAIIO Jornadas Argentinas de Informática
4 al 8 de SEPTIEMBRE de 2017
UTN-FRC - Córdoba Argentina

1. Introducción

La evolución de procesos en la organización que requieren firma digital y la complejidad de los mismos a la hora de la manipulación y generación de estos documentos firmados digitalmente determinan la necesidad de contar con una herramienta que provee una solución para resolver estas problemáticas. Es fundamental que dichos procesos se realicen de una forma homogénea, manteniendo un estándar a la hora de decidir las distintas alternativas y formatos a utilizar en el momento de firmar o encriptar información. Es deseable contar con un registro de todos los documentos firmados que permitan en cualquier momento poder recuperar o validar cualquier documento emitido. Dentro de los objetivos esperados podemos destacar:

- Contar con una solución homogénea que de soluciones y facilite el uso de certificados digitales con fines de firma digital, encripción y desencripción de datos.
- Proveer una herramienta que permita realizar firma digital de documentos en aplicaciones web, pensando siempre en la facilidad del usuario, sin que el mismo tenga que realizar operaciones complejas.
- Contar con un registro de todos los documentos firmados digitalmente con fines de auditoria. Además con dicho registro poder proveer de una copia de los mismos en caso de ser necesario y contar con la posibilidad que un documento impreso pueda ser recuperado y verificado digitalmente mediante el uso de un código QR en el mismo.
- Llevar un registro de todos los certificados emitidos, el cual permitirá en todo momento conocer que certificado es válido para cada persona. También llevar un registro/control de los certificados de servidores, alertando sobre el vencimiento de los mismos.
- Proveer servicios Web referentes a firma digital, encripción/desencripción, validación de firma. Dichos servicios harán uso de certificados de aplicaciones.
- Suministrar un manual de referencia para las áreas técnicas detallando el uso de los distintos módulos desarrollados como así también el uso de buenas prácticas.

2. Situación-Problema u Oportunidad

Uno de los desafíos más grandes del proyecto es poder firmar del lado del cliente desde una aplicación Web con el uso de un certificado personal almacenado en un SmartCard/Token. La complejidad de dicha tarea se basa principalmente en la imposibilidad de acceder a los recursos de la PC desde el browser, en este caso particular no poder acceder directamente al Token.

Anteriormente se estaba utilizando un Applet de Java que permitía cumplir esta tarea, pero el mismo nos presentaba cada vez más problemas, como ser:

- Continuos problemas de incompatibilidad ante cambios de versiones de Java
- Muchas advertencias de seguridad al usuario al momento de descargarlo/usarlo.
- Incompatibilidad con ciertos navegadores (Chrome, Edge).
- En próxima versión de Java 9 ya no estará disponible el plugin de java, el cual provocara que los Applets no funcionen en ningún browser.

XLIII CLEI Conferencia Latinoamericana de Informática | 46 JAIIO Jornadas Argentinas de Informática
4 al 8 de SEPTIEMBRE de 2017
UTN-FRC - Córdoba Argentina

3. Solución

La sustitución de los Applets de Java por otra tecnología no es tarea fácil, ya que, por una parte, se necesita una comunicación bidireccional con el JavaScript de la página Web (la firma es solo un paso dentro de un complejo flujo de trabajo) y, por otra, un acceso a los almacenes de claves y certificados.

Dentro de las alternativas evaluadas podemos mencionar:

1. Desarrollo de extensiones para navegadores Web
2. Implementación de un API 100% JavaScript
3. Uso de aplicaciones nativas con invocación por protocolo

Desarrollo de extensiones para navegadores Web: La mayoría de los navegadores Web soportan ampliaciones de su funcionamiento vía los llamados plugins o addons. Si bien esta alternativa permitiría dar una experiencia de usuario muy buena, ya que se tendría una vía directa desde el JavaScript del browser, el costo de implementación es muy alto, ya que se necesita un desarrollo particular para cada browser a soportar.

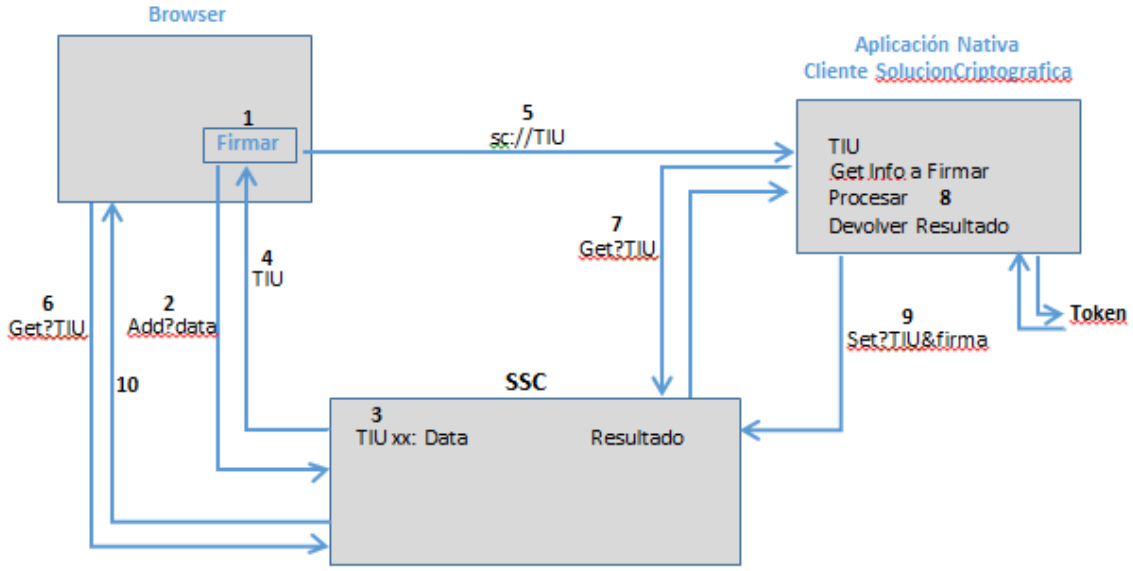
Una solución 100% JavaScript seria la ideal, eliminando la necesidad de aplicaciones adicionales. Sin embargo al día de hoy no existe soporte para todas las operaciones criptográficas necesarias y menos para el acceso directo al repositorio de los certificados. Si bien se están escribiendo algunos estándares el futuro es incierto respecto a si los distintos browsers optaran por implementarlo.

La invocación por protocolo consiste en registrar una aplicación para atender el llamado de un determinado protocolo, de forma que cuando se produzca una llamada para apertura de una URI con ese protocolo, se invocará dicha aplicación. Para esto sería necesario entonces el desarrollo de una aplicación nativa que resuelva la firma y acceso a certificados y la registración de un protocolo asociada a la misma.

De esta forma para el caso de la firma web se invocaría por JavaScript a dicho protocolo, pasando por parámetros todo lo necesario para realizar la firma. El problema reside que esta comunicación es en un solo sentido, no pudiendo devolver la aplicación nativa la respuesta al JavaScript del browser que la invoco. Para solucionar esta vuelta, lo que haremos será que la comunicación entre el JavaScript y la aplicación nativa se produzca mediante un servidor intermediador (el cual además nos dará la posibilidad de tener un registro de auditoria de todo lo firmado).

XLIII CLEI | 46 JAIIO
Conferencia Latinoamericana de Informática | Jornadas Argentinas de Informática
4 al 8 de SEPTIEMBRE de 2017
UTN-FRC - Córdoba Argentina

De las tres alternativas planteadas, la de invocación por protocolo es la que nos parece que posee más beneficios y es la elegimos y expondremos en detalle a continuación.



1-Desde la aplicación Web se invoca a una API JavaScript para firmar. El mismo dispara los siguientes eventos:

2,3,4-Realiza llamada a servidor intermediador (SSC) el cual guarda toda la información necesaria para realizar la firma y devuelve un token de identificación único (TIU)

5-Dicha API además invoca por protocolo a la URI: crypto://TIU el cual dispara a la aplicación nativa y pasa el parámetro TIU.

6-Se queda esperando de forma asincrónica la respuesta del servidor.

7-La aplicación toma el TIU que recibió y con el mismo recupera desde el servidor toda información necesaria para firmar.

8-Se realiza la firma

9-La aplicación nativa devuelve el dato firmado al servidor

10-Se le responde al cliente JavaScript que se había quedado esperando por la respuesta (punto 6) con los datos firmados.

Como se ve en el gráfico, la solución consta de tres partes:

- Módulo JavaScript: El cual exponga una API que provea el método de firma y devuelva los datos firmados. Este módulo tendrá desarrollada toda la lógica para hablar con el servidor SSC, invocar por protocolo al firmador y quedarse de forma asincrónica esperando la respuesta de la firma.
- Aplicación nativa: Esta aplicación será la que se disparara por protocolo. En este caso al ser una aplicación nativa que corre en la máquina del usuario nos encontramos con distintos desafíos entre los que podemos enumerar:
  - Como distribuir esa aplicación a todos los clientes de forma fácil

- Como mantener a la misma actualizada una vez que se encuentre distribuida
- Como lidiar con temas de privilegios de administrador para la instalación de la misma.

  Para lidiar con estos temas nos encontramos con dos posibles soluciones: ClickOnce o Java Web Start. La opción adoptada es la de ClickOnce ya que resuelve todos los problemas expuestos anteriormente de manera transparente, publicando la aplicación desde una página web.

- Servidor intercambiador (SSC): Es una aplicación web que expone los servicios necesarios para que la aplicación cliente web que requiere firmar logre comunicarse de manera bidireccional con la aplicación nativa. Como ventaja adicional nos permite mantener un registro y control de todo lo que es firmado por los clientes.

La plataforma fue desarrollada íntegramente por personal de la Gerencia de Gestión de Proyectos y Servicios de la Agencia, sobre el lenguaje de programación C# exponiendo API Rest y aplicación ClickOnce que posee las operaciones criptográficas y acceso a certificados digitales, módulo Javascript para acceso desde las aplicaciones web al firmador, motor Oracle como repositorio de información.

4. Aportes

La arquitectura planteada no solo logra el reemplazo de Applets para firma digital sino que nos deja dotados de una plataforma sobre la que podemos ir montando distintos tipos de servicios, como ser: encripcion/desencripcion de datos, integración con APIs para agregar información adicional en la firma como por ejemplo el cargo de la persona firmante, el agregado de firma de aplicación del lado del servidor en forma transparente, comprobación de revocación/vencimientos de los certificados, etc.

Los beneficiarios directos son los agentes ARBA que utilizan la firma digital dentro de la organización ya que están alcanzados por procesos que requieren documentación respaldada con firma digital entre los que podemos destacar: workflows de aprobación de permisos, sistema de expedientes, tramites en general. Por otro lado podemos señalar como beneficiarios indirectos a el resto de la organización ya que permite sustituir procesos en papel por digitales logrando que los mismos sean más simples/agiles.

La problemática resuelta es algo que hoy está presente en todos los organismos debido a la necesidad de contar con una plataforma ágil para firma digital que reemplace el uso de los Applets actuales.

La arquitectura con Invocación por protocolo logró que la instalación del firmador lo pueda resolver directamente el usuario final, ya que el mismo es descargado en forma automática y no requiere ningún tipo de configuración para su funcionamiento (con el uso de Applet era necesario que un personal de TI realice seteos de configuración/seguridad en la PC para su correcto funcionamiento).

El grado de facilidad de reproducción es muy alto para cualquier organismo público o privado, ya que uno de los principales objetivos a alcanzar fue la facilidad de implementación/uso. Como la solución de firma es independiente de las aplicaciones que poseen la lógica de negocio, el uso del firmador es posible por cualquier aplicativo mediante la incorporación de una referencia a una librería Javascript y uso de una API que provee la misma.

5. Referencias

Firma Digital

http://www.iprofesional.com/notas/15246-La-firma-digital-una-herramienta-para-agilizar-los-trmites

Gobierno de España

http://svn-ctt.administracionelectronica.gob.es/svn/clienteafirma/docs/
http://forja-ctt.administracionelectronica.gob.es/file/frs/download.php/1859/Estudio_Alternativas_Applets_Java_cliente_Afirma_1.2.pdf?session_hash=

Invocación por protocolo

https://msdn.microsoft.com/en-us/library/aa767914(v=vs.85).aspx

ClickOnce

https://en.wikipedia.org/wiki/ClickOnce
http://stackoverflow.com/questions/7334460/how-can-i-associate-a-custom-url-protocol-with-a-clickonce-app

Certificados

http://paulstovell.com/blog/x509certificate2