# https://encrypted-tbn2.google.com/images?q=tbn:ANd9GcSjE6inN_DbrwHuZE-lPoP4gVcN0gmXmQlLeFA50Y1NQAADxQwW

# El Cliente @firma en plataformas móviles

## Introducción

El actual Cliente @firma cubre con éxito la gran mayoría de plataformas cliente (Windows, Linux, Mac OS X, Solaris,…) pero desde hace unos pocos años los usuarios han ido migrando paulatinamente desde sus dispositivos tipo PC hacia otros más portátiles, pequeños y fáciles de usar para muchas de sus operaciones en la red.

Siendo este cambio de tendencia cada vez más acusado, se planteó desde el Proyecto Cliente @firma la necesidad de soporte de estas nuevas plataformas (incompatibles con las tradicionales tipo PC) para poder ofertar servicios de administración electrónica y seguridad basada en firma acorde con las nuevas preferencias y hábitos de ciudadanos y usuarios en general.

## El modelo de aplicación de firma electrónica en plataformas móviles

En el Proyecto Cliente @firma se ha trabajado con anterioridad en dos modelos de aplicaciones, por una parte aplicaciones Web en las que la firma electrónica se integra directamente en las páginas mediante un Applet de Java y cierta lógica en JavaScript, y por otra aplicaciones “de escritorio” que trabajan sin necesidad de conexiones de red con los ficheros locales del ordenador.

Para los dispositivos móviles de nueva generación, estos modelos han sido adaptados para permitir una migración sencilla por parte de los proveedores de servicios (AAPP, empresas, etc.) y una mayor cercanía a lo que pueden esperar los usuarios.

### Aplicación Web

El Proyecto Cliente @firma está desarrollando un modo de aplicación Web que permite una provisión de servicios o aplicaciones con firma electrónica de forma homogénea para entornos tipo PC y entornos móviles de nueva generación.

Así, una aplicación Web puede desplegarse normalmente y, al ser accedida desde un entorno PC (Windows, Linux, Mac OS X, etc.), se cargará en forma de Applet de Java, mientras que en un entorno móvil (Google Android, Apple iOS, etc.), se cargará como “App” móvil. Esta lógica de decisión es trasparente para los usuarios.

Este modo presenta grandes ventajas para los integradores, ya que permite mantener sus aplicaciones Web “clásicas”, con sus flujos de negocio asociados, pero dando servicio a una nueva colección de dispositivos (y por lo tanto de usuarios) sin costes asociados de desarrollo, mantenimiento, distribución o despliegue, etc.

### “App” móvil a medida

Aunque la aplicación Web puede adecuarse a prácticamente cualquier necesidad, las nuevas plataformas móviles han instaurado con éxito un nuevo modelo en el cual cada servicio (o agrupación de servicios cohesionados) se provee mediante una aplicación independiente, que se denomina “App” (denominación acuñada por Apple). En este nuevo modelo se descarta el Navegador Web como portal hacia las aplicaciones y se opta por multitud de aplicaciones nativas con interfaces de usuario avanzados, atractivos diseños gráficos y potentes capacidades multimedia.

Dado que de esta forma cada servicio necesita la codificación de una “App” independiente, no es ya responsabilidad del Cliente @firma el proporcionar una aplicación completa y “empaquetada” (esto debe construirlo el propio proveedor de servicios), y el Proyecto Cliente @firma opta por poner en manos de los desarrolladores una serie de módulos funcionales independientes con los que pueden desarrollar aplicaciones basadas en firma electrónica:

* Funcionalidades del núcleo del Cliente @firma
* Gestión de almacenes de claves y certificados
* Firmas CAdES
* Firmas PAdES
* Firmas XAdES
* Etc.

### Aplicación de firma independiente genérica

La última opción es la más simple y está orientada a solventar las necesidades más sencillas. Consiste simplemente en una aplicación independiente (instalada como una “App” más) que permite realizar firmas electrónicas sobre ficheros almacenados en el dispositivo.

Actualmente se trabaja también en la posibilidad de firmar ficheros remotos, como por ejemplo en la nube (Apple iCloud, Microsoft Skydrive, etc.), ficheros sincronizados (Apple iTunes, etc.), etc.

## Las plataformas móviles

En este nuevo ecosistema de dispositivos móviles, nuevas plataformas se están rápidamente posicionando y copando distintas cuotas de mercado:

### Google Android



El sistema operativo de Google para teléfonos, tabletas y portátiles ligeros es líder en cuota de mercado en nuestro país, aunque no en conexiones a Internet. Aun así, es una de las prioridades en cuanto a soporte por parte del Proyecto Cliente @firma.

Una de las peculiaridades de Google Android es que las aplicaciones se codifican en lenguaje Java y se ejecutan en una máquina virtual similar a la JVM (Apache Harmony / Dalvik). Dado este entorno de ejecución, el Proyecto @firma trabajó durante 2011 para tener una base Java que proporcionase una compatibilidad cruzada entre JSE (Linux, Mac OS X, Windows y Solaris) y Dalvik. Esta compatibilidad global ha permitido contar con una infraestructura de firma electrónica muy robusta y depurada con una funcionalidad excepcional.

En la actualidad, el Proyecto @firma cuenta con los siguientes productos para Google Android:

* Módulos para desarrollo de aplicaciones
  + Gestión de almacenes de claves y certificados.
    - Actualmente en progreso el soporte de los nuevos almacenes centralizados de Android 4.
  + Módulo de firmas CAdES
    - Soporte de firmas en una, dos y tres fases[[1]](#footnote-1)
  + Módulo de firmas PAdES
    - Soporte de firmas en una (en progreso), dos y tres fases1
* Aplicación para firma desde Web (JavaScript + “App”)
* Aplicación de firma independiente

### Apple iOS (iPad, iPhone, iPod)



El sistema operativo de Apple es líder indiscutible en tabletas y en general en dispositivos móviles conectados a Internet, y, gracias a su altísimo nivel de innovación y actualización tecnológica, se prevé que mantendrá en el medio plazo esta posición privilegiada, siendo objetivo obligado para el Proyecto Cliente @firma.

Al contrario que Android, las aplicaciones iOS no se codifican en Java, sino en Objective C. Esta peculiaridad no permite la reutilización de código desde la base JSE actual del Cliente @firma, y está obligando a una recodificación completa de ciertos módulos. No obstante, la experiencia, conceptos e implementaciones de los estándares aprendidos en el desarrollo Java están permitiendo conseguir una implementación de gran calidad.

En la actualidad, el Proyecto @firma cuenta con los siguientes productos para Apple iOS:

* Módulos para desarrollo de aplicaciones
  + Gestión de almacenes de claves y certificados (en progreso).
  + Módulo de firmas CAdES (en progreso)
    - Soporte de firmas en una y tres fases
* Aplicación para firma desde Web (JavaScript + “App”) (en progreso)
* Aplicación de firma independiente (en progreso)

### RIM BlackBerry



La plataforma RIM BlackBerry, aunque en nuestro país ha mantenido una presencia destacable, sufre un severo retroceso frente a Apple iOS y Google Android, y es esta menor relevancia la que convierte a RIM en una prioridad secundaria para el Proyecto Cliente @firma.

A nivel tecnológico, RIM BlackBerry cuenta con un entorno de ejecución de aplicaciones basado en Java y en concreto en la plataforma JME. Esta compatibilidad con Java permite la reutilización directa de buena parte de los componentes del Cliente @firma, heredando sus funcionalidades y su excelente nivel de documentación y pruebas.

En la actualidad, no se prevé la adaptación del Cliente @firma a dispositivos RIM BlackBerry, aunque se ha cuidado el diseño del Proyecto para dejar la puerta abierta a los siguientes productos:

* Módulos para desarrollo de aplicaciones
  + Gestión de almacenes de claves y certificados.
  + Módulo de firmas CAdES
    - Soporte de firmas en tres fases
  + Módulo de firmas PAdES
    - Soporte de firmas en tres fases

### Microsoft Windows Phone



Aunque Microsoft Windows Phone es un “recién llegado” al ecosistema y apenas cuenta con base de usuarios, sus optimistas previsiones de crecimiento hacen que el Proyecto Cliente @firma se lo plantee como plataforma compatible en un futuro.

A nivel tecnológico, Windows Phone incluye un entorno de ejecución .NET, lo que obliga a recodificar los componentes del Cliente @firma necesarios, aunque la similitud entre .NET y Java hacen que estar tarea se pueda realizar con un riesgo limitado.

En la actualidad, el Proyecto @firma no se plantea el desarrollo a corto-medio plazo de productos para Windows Phone. Sin embargo, en vista de la evolución del mercado, se puede evaluar el desarrollo futuro de:

* Módulos para desarrollo de aplicaciones
  + Gestión de almacenes de claves y certificados.
  + Módulo de firmas CAdES
    - Soporte de firmas en una y tres fases
  + Módulo de firmas PAdES
    - Soporte de firmas en una y tres fases

Diferentes alternativas en la firma multi-fase avanzada cliente-servidor

# Firma electrónica en una fase

## Descripción

La firma electrónica en una única fase es aquella en la que la totalidad del proceso re realiza en un mismo sistema, no existiendo comunicación alguna con servicios o servidores externos.

En este modelo, el sistema recibe los datos a firmas, obtiene el certificado y la clave privada localmente y firma los datos igualmente de forma local.

Es el modo habitual de operación del Cliente @firma.

# Firma electrónica en tres fases

## Descripción

La firma electrónica en tres fases está pensada para entornos donde la clave privada reside en un sistema con al menos alguna de las siguientes restricciones:

* El sistema no es compatible con el Cliente @firma. En este caso, dado que el 95% del código se ejecuta en un sistema externo, solo es necesario portar el 5% restante.
* El sistema tiene unas capacidades muy limitadas en cuanto a proceso computacional, memoria o comunicaciones por red. En este caso, el sistema solo realiza una operación criptográfica, una firma PKCS#1, mucho menos demandante de potencia de proceso que una firma completa CAdES, y, adicionalmente, no trata el documento a firmar completo, sino únicamente una pequeña cantidad de datos resultante de un pre-proceso (la pre-firma) realizado por el sistema externo, lo que resulta en un enorme decremento en las necesidades de memoria y transmisión de datos (esto último si decide omitirse la transferencia del fichero a firmar).
* Por motivos de seguridad, el documento a firmar no puede salir de un sistema externo. Como se ha descrito en el punto anterior, en este caso es posible omitir por completo la salida del documento del sistema externo, y puede transferirse únicamente el resultado de la pre-firma, desde la cual es imposible reconstruir el documento original.

Estos condicionantes convierten la firma trifásica en una opción perfectamente adaptada a los dispositivos móviles, donde se dan tanto la heterogeneidad de sistemas operativos (Apple iOS, Google Android, RIM BlackBerry, Microsoft Windows Phone, etc.) y las limitaciones en potencia de proceso, memoria y comunicaciones; en estas últimas hay que tener en cuenta el coste, especialmente si estamos haciendo uso de una red de otro operador en itinerancia (*roaming*).

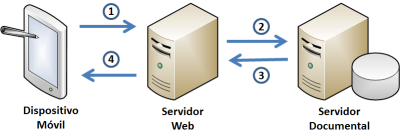
En una firma trifásica, los datos que se transfieren entre servidor y cliente consisten en (previamente el cliente ha debido iniciar una petición de firma trifásica indicando referencia de documento y enviando la cadena de certificados del firmante):

* Atributos firmados en el caso de CAdES.
* Atributos firmados más identificador de fichero PDF y fecha de inicio del proceso (para reutilizarla en todas sus fases) en el caso de PAdES.
* Nodo XML a firmar (que contiene las huellas digitales de las referencias a firmar) en el caso de XAdES.

El cliente devuelve al servidor en todos los casos la firma PKCS#1, acompañada en el caso de PAdES de el identificador de fichero PDF y la fecha de inicio del proceso.

El funcionamiento típico de una firma trifásica en la que intervienen un dispositivo móvil, un servidor Web (que hace la pre-firma y la post-firma) y un servidor documental podría ser el siguiente:

Pre-firma:



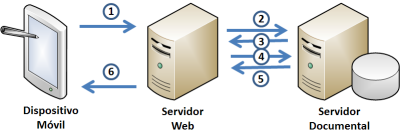
1. El dispositivo móvil solicita una pre-firma al servidor Web indicando un identificador de documento.
2. El servidor Web solicita el documento a servidor documental.
3. El servidor documental entrega el documento al servidor Web.
   1. Es importante recalcar que el servidor documental no necesita almacenar ningún dato de sesión y que este no está expuesto a Internet de forma directa en ningún momento.
4. El servidor Web calcula la pre-firma, entregando el resultado (muy pequeño en tamaño) al dispositivo.
   1. Es importante recalcar que el servidor Web no necesita almacenar ningún dato de sesión ni exponer los documentos directamente al dispositivo.

Firma:



1. El dispositivo móvil realiza, de forma completamente aislada una firma electrónica simple (computacionalmente ligera) de los datos de la pre-firma. La clave privada del usuario nunca sale del dispositivo y no se expone externamente en ningún momento.

Post-firma:



1. El dispositivo móvil solicita una post-firma al servidor Web indicando un identificador de documento y proporcionando el resultado de su pre-firma firmada.
2. El servidor Web solicita el documento a servidor documental.
3. El servidor documental entrega el documento al servidor Web.
4. El servidor Web calcula la post-firma y compone el documento final firmado, entregando el resultado al servidor documental para su almacén.
5. El servidor documental almacena el nuevo documento y devuelve un identificador al servidor Web.
6. El servidor Web comunica al dispositivo el éxito de la operación y el identificador del fichero ya firmado y almacenado.

El esquema podría ser igualmente implementado sin servidor documental, pudiendo obtener el Servidor Web el documento desde otro origen, incluyendo el propio dispositivo móvil. Igualmente, una vez firmado el documento, su destino puede ser cualquiera, incluyendo de nuevo al propio dispositivo.

Es conveniente tener en cuenta al usar firmas trifásicas que es necesario disponer de un mecanismo para que el usuario pueda ver en todo momento los documentos que está firmando (una copia que refleje con fidelidad el contenido firmado puede ser suficiente) para evitar situaciones de repudio.

Una ventaja adicional en las firmas trifásicas es que, puesto que la última fase la realiza el servidor y cuenta ya con el documento

## Implementación

La implementación de la firma trifásica es posible en cualquier caso, pero siempre teniendo en cuenta las siguientes consideraciones:

CAdES

La implementación de firma trifásica CAdES no presenta complicaciones extraordinarias:

* Dificultad: Baja
* No es necesaria la modificación de ningún API externo a @firma.

PAdES

La implementación de firma trifásica PAdES presenta las siguientes peculiaridades:

* Dificultad: Media-Alta
* Es necesario modificar el API iText.
  + Realmente, la modificación de iText no supone una traba en la evolución de @firma, ya que este usa una versión antigua concreta (2.1.7) por temas de licenciado.

La dificultad de la implementación de las firmas trifásicas PAdES radica en la adición de elementos aleatorios (por ejemplo, el identificador de fichero) y fechas de creación de secciones dentro de los documentos PDF que son necesario sincronizar entre cliente y servidor para asegurar que las huellas digitales no difieren.

XAdES

La implementación de firma trifásica XAdES presenta ciertas dificultades dado el encapsulamiento del API XMLDSig de Java, siendo necesario implementar el concepto de *Facets* de firma XML.

* Dificultad: Alta
* Es necesario modificar el API JXAdES.
  + Realmente, la modificación de iText no supone una traba en la evolución de @firma, ya que estas modificaciones se realizarían conjuntamente con el equipo de JXAdES atendiendo específicamente a las necesidades de @firma, y las modificaciones se incorporarían de forma definitiva a JXAdES.

# Firma electrónica en dos fases

La firma electrónica en dos fases comparte algunos escenarios de uso preferente con la firma en tres fases, pero presenta diferencias significativas:

* El 90% del código se ejecuta en servidor, lo que facilita migrar el 10% restante a plataforma actualmente no soportadas por el Cliente @firma.
* El documento inicia el proceso desde el dispositivo y lo finaliza también en el dispositivo, por lo que es adecuado para procesos donde no interviene un servidor de documentos.
* Se reducen las conexiones de red respecto a la firma trifásica (solo se necesita una conexión), pero el tráfico de estas aumenta, lo cual simplifica la operación cuando el servidor Web requiere autenticación.
* Se mantiene, tal y como ocurre en la firma trifásica, una demanda baja en cuanto a potencia computacional en el dispositivo, pero no así la demanda de memoria. Este traslado de necesidades de memoria del servidor al dispositivo permite a este primero tratar un altísimo volumen de peticiones con un hardware de gama media.

En una firma bifásica, los datos que se transfieren entre cliente y servidor constan de:

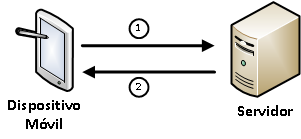
* Documento a firmar.
* Cadena de certificados del firmante.

Y la respuesta del servidor al cliente:

* Documento pre-firmado.
* Datos a firmar mediante PKCS#1.
* Información necesaria para insertar esta firma PKCS#1 en el documento pre-firmado.
  + Desplazamiento (*offset*) dentro del binario donde debe colocarse la firma PKCS#1, cadena de texto a sustituir por la firma PKCS#1 (en Base64 o en su representación ASCII del hexadecimal, etc.).

El funcionamiento típico de una firma bifásica en la que intervienen un dispositivo móvil y un servidor Web (que hace la pre-firma) podría ser el siguiente:

Pre-firma



1. El dispositivo móvil solicita una pre-firma al servidor Web enviando la cadena de certificados del firmante (puede enviar igualmente el documento o el servidor Web puede obtenerlo de una fuente externa, como un servidor de documentos).
2. El servidor Web devuelve la pre-firma al dispositivo (que contiene el documento preparado para la firma final y los datos binarios a firmar mediante PKCS#1) y da por finalizado el proceso en su extremo.

Firma



1. El dispositivo móvil realiza, de forma completamente aislada una firma electrónica simple (computacionalmente ligera) PKCS#1 de los datos de la pre-firma y realiza él mismo el proceso de inserción en el documento pre-firmado. Este proceso es relativamente ligero en cuanto a potencia computacional, pero puede requerir mucha memoria en el dispositivo.

# Implementación de tecnologías multi-fase dentro del Proyecto @firma

## Comunicaciones entre cliente y servidor y desarrollos en la parte servidora

Para la comunicación entre cliente y servidor se hace uso de tecnologías REST (Transferencia de Estado Representacional). REST presenta numerosas ventajas respecto a otros sistemas en el ámbito de las firmas multi-fase:

* Es un protocolo sin estado.
  + Combinado con una implementación en la que no es necesario almacenar ningún dato de sesión en el servidor incrementa la seguridad del sistema, ya que en caso de compromiso de este no hay documentos del usuario almacenados susceptibles de apropiación indebida.
* Es un protocolo simple.
  + La ausencia de SOAP y las limitaciones en el uso de XML lo hacen apto para dispositivos con capacidades limitadas, a la vez que facilitan una implementación rápida y fácil de mantener en el lado cliente.

Se realiza una implementación utilizando exclusivamente tecnologías presentes en JEE 6 (sin usar API de productos externos), lo cual permite una completa independencia tecnológica en cuanto a servidores de aplicaciones y blinda en cierto modo la futura obsolescencia.

El resultado es un servicio por completo independiente del resto de servicios de servidor de la plataforma @firma (que están ligados a versiones obsoletas de Axis, no aptas para implementar modernos servicios basados en REST). No obstante, la aplicación servidora, en forma de EAR o WAR, podrá desplegarse en el mismo servidor de aplicaciones que la plataforma @firma, siempre que este sea compatible JEE6.

Para facilitar las labores de pruebas e implantaciones de referencia se ofrece adicionalmente un servidor GassFish Embedded configurado para el arranque automático del servicio.

Uso del Cliente @firma en un entorno servidor

El servicio servidor hace uso del Cliente @firma a modo de bibliotecas, beneficiándose de la organización en módulos actual.

1. Ver anexo sobre firma electrónica en varias fases [↑](#footnote-ref-1)