Documentación técnica de Firma.gub.uy

Autor

Área de Seguridad de la información de Agesic

Fecha de creación

11/05/2023

Tipo de publicación

Guía técnica

Resumen

Documentación técnica de Firma.gub.uy

Firma Digital: descripción técnica y legal

De acuerdo a la Ley Nº 18.600 del 21 de setiembre de 2009, la Firma Digital se define como los datos en forma electrónica anexos a un documento electrónico o asociados de manera lógica con el mismo, utilizados por el firmante como medio de identificación

Dicho de otra manera, la Firma Digital es un conjunto de datos que se agregan a un documento para identificar al firmante, comprometiéndolo con el contenido del documento en cuestión. Como se puede observar, esta definición carece de requerimientos específicos sobre cómo debe ser implementada tecnológicamente la Firma Digital; y, más aún, en su artículo quinto la ley establece que las partes son libres de acordar las condiciones en las cuáles aceptan las firmas, incluyendo el mecanismo tecnológico.

De lo anterior se desprende que las partes que intercambian documentos tienen flexibilidad para elegir el mecanismo de Firma Digital a utilizar, siempre que estén en acuerdo previo respecto al elegido, y que este permita identificar correctamente al firmante y su intención de firma sobre el documento. Esto permite que en el plano práctico se puedan implementar firmas digitales con elementos como usuarios y contraseñas, OTP, biometría, etc, simplemente haciendo un acuerdo entre las partes con lo que considerarán firma. Y, más importante aún, definiendo una forma en la que se asociará la información que identifica al usuario al documento digital en sí mismo. Es conveniente aclarar que **la firma hológrafa escaneada no constituye una Firma Digital**, dado que no es posible realizar un eventual peritaje de que efectivamente es la firma. Esto dota al esquema de firma de una altísima flexibilidad para acceder a garantías jurídicas de base en el intercambio de documentación y transacciones electrónicas.

No obstante, las garantías de riesgos reales de integridad y no repudoio de la Firma Digital, que son las características más deseables, estarán siempre dadas por los mecanismos utilizados. Por eso se define la Firma Electrónica Avanzada con requerimientos específicos que hacen que su validez no dependa de acuerdos previos, se otorguen garantías de integridad por defecto y pueda ser utilizada en documentos públicos.

Para la implementación de firmas digitales, se recomienda siempre el uso de Firma Electrónica Avanzada. La firma digital "común" debería ser utilizada solo cuando el uso de Firma Electrónica Avanzada no sea posible por fundamentadas restricciones tecnológicas o de negocio; y, en caso de usarse, se debe hacer especial hincapié en cómo se garantiza la voluntad del sujeto de firmar un documento en particular, ya que la libertad que otorga la ley se vuelve una debilidad si no se es cuidadoso en este punto.

Firma Digital

Si bien el término jurídico reconocido en la Ley Nº 18.600 del 21 de setiembre de 2009 es *Firma Electrónica Avanzada*, desde sus orígenes la comunidad tecnológica habla de *Firma Electrónica* y *Firma Digital* como términos equivalentes, por lo que en el presente sitio *Firma Digital* o *Firma Digital Avanzada* corresponden, simplemente, a otras formas de denominación técnica para la *Firma Electrónica* y la *Firma Electrónica Avanzada.* Por lo tanto, son utilizados en forma intercambiada.

Firma Electrónica Avanzada

Según la Ley Nº 18.600, del 21 de setiembre de 2009, la Firma Electrónica Avanzada es una Firma Digital con las siguientes características:

1. Requiere información de exclusivo conocimiento del firmante, permitiendo su identificación inequívoca.
2. Debe ser creada por medios que el firmante pueda mantener bajo su exclusivo control.
3. Debe ser susceptible de verificación por terceros.
4. Debe estar vinculada a un documento digital de tal modo que cualquier alteración subsiguiente en este sea detectable.
5. Debe haber sido creada utilizando un dispositivo de creación de firma técnicamente seguro y confiable y estar basada en un certificado reconocido como válido al momento de la firma.

La Firma Electrónica Avanzada tiene idéntica validez y eficacia que la firma autógrafa consignada en documento público o en documento privado con firmas certificadas, por lo que no requiere acuerdo previo para su uso. Además, provee garantías de integridad, autenticidad y no repudio sobre los documentos intercambiados. Esto hace que su uso permita la eliminación completa del papel en cualquier acto entre partes públicas o privadas, constituyendo un elemento fundamental para el desarrollo de la sociedad digital.

En la práctica, una Firma Electrónica Avanzada es una firma hecha con pares de llaves asimétricas y certificados x509 tradicionales, con las particularidades de que el certificado debe ser emitido por una autoridad acreditada ante la Unidad de Certificación Electrónica (UCE) y que, además, la llave privada debe residir en un dispositivo seguro de creación de firmas.

En el caso de firmas de persona física, se exige que el dispositivo seguro de creación de firmas sea un token criptográfico, tarjeta inteligente, HSM u otro dispositivo que otorgue garantías de seguridad equivalentes.

Para la firma de persona jurídica se admite el uso de dispositivos de software, dado que el uso es previsto para sistemas automáticos y por lo tanto el riesgo asociado a la mala manipulación de la llave privada es considerablemente menor. Los

prestadores acreditados son Abitab, Antel, el Correo y el Ministerio del Interior para los certificados de la Cédula Digital. Cada uno tiene en sus respectivos sitios web el procedimiento para la solicitud de sus certificados, además de la información adicional requerida como declaraciones de políticas, certificado propio y listas de revocación, entre otros.

En el esquema de Firma Electrónica Avanzada más común, el firmante realiza un hash del documento electrónico, lo cifra con su llave privada y anexa el resultado al documento como prueba de su firma. La contraparte, usando el certificado público del firmante (en particular, su llave pública), puede descifrar el hash originalmente enviado y compararlo contra el hash calculado nuevamente del documento que recibió. Si coinciden, la contraparte puede tener la certeza de que el documento no fue modificado por una tercera parte (Integridad) y que, además, fue firmado por la persona, física o jurídica, que figura identificada en el certificado (Autenticidad). Debido a que el certificado público del firmante fue emitido por un prestador acreditado y confiable, sus procedimientos de registro garantizan efectivamente la identidad del firmante (no repudio).

Por todo esto, la Firma Electrónica Avanzada es el mecanismo de firma recomendado para cualquier transacción o intercambio de documentación que requiera firma de la contraparte, pudiendo también ser utilizada para autenticación fuerte.

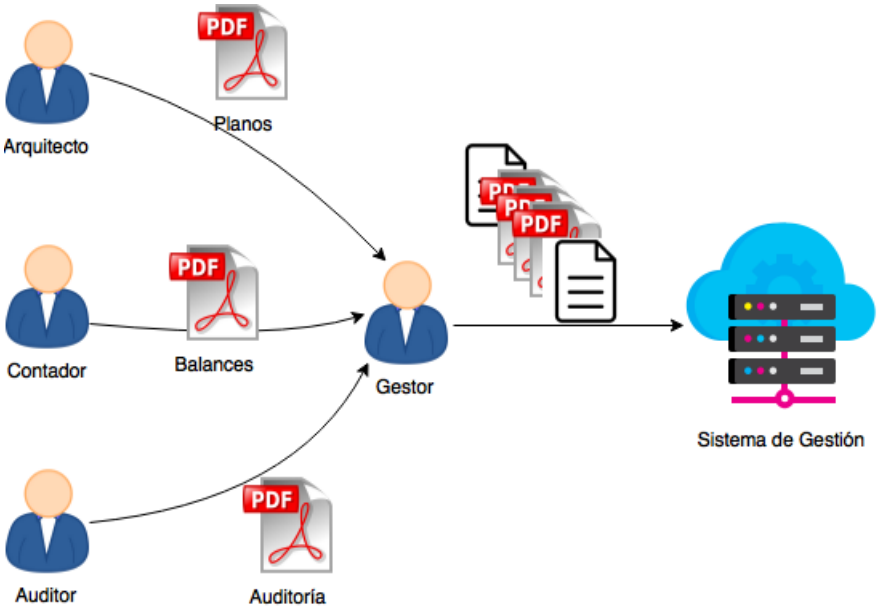
Implementación de la Firma

Siguiendo el esquema anteriormente mencionado, cualquier sistema que desee implementar la Firma Electrónica Avanzada debe acceder a la llave privada del firmante y cifrar un hash del documento que desea hacerlo firmar, siendo la implementación concreta dependiente de la plataforma obviamente. Siempre el acceso a la llave privada estará protegido por un secreto que el firmante debe cuidar, que es el PIN o contraseña. Este secreto puede que sea usado para autenticarse ante un dispositivo y habilitar acceso a los comandos de uso de la llave como en la Cédula de Identidad Digital o para descifrar el contenedor de software donde se almacena la llave, dependiendo del escenario de uso; a nivel conceptual, es lo que garantiza al firmante que, aún siendo comprometido el soporte en el que tiene la llave, sigue manteniendo control exclusivo sobre esta. Es por esto que el almacenamiento de pines o contraseñas, especialmente en el caso de personas físicas, es algo totalmente riesgoso y desaconsejable.

Firma de Persona Física

Es el caso en que una persona física desea comprometerse con el contenido de un documento, por ejemplo, la firma de un contrato, de un formulario o de una transacción bancaria, entre otros. La particularidad que presenta es que, al ser emitida exclusivamente en dispositivos criptográficos, el software que desee implementar Firma Electrónica Avanzada de persona física deberá necesariamente acceder a estos dispositivos. Para el caso de la Cédula de Identidad Digital, se recomienda ver la sección especialmente dedicada a esta, ya que si bien es similar a los demás dispositivos, presenta algunas particularidades y opciones de implementación adicionales.

El caso más básico se da cuando el usuario tiene que firmar documentos de elaboración manual, como cartas, contratos, acuerdos, formularios descargables o similares. En estos casos, lo más sencillo es que el usuario elabore o complete el documento en su editor de texto de preferencia, lo exporte como PDF, lo firme localmente con su Cédula de Identidad Digital o su dispositivo de seguridad de preferencia y lo remita a la otra parte por el medio que considere adecuado, que puede ser directamente un correo electrónico, upload de archivo o cualquier otra vía digital. En el caso de firma de contratos, términos y condiciones y similares, puede que ni siquiera sea necesario editar nada; simplemente se abre el PDF que se recibió, se lo firma y se lo devuelve a la contraparte. Para firmar un PDF localmente en tu PC con Adobe Reader, podés consultar el tutorial: Firmar PDF con Firma Avanzada en Adobe.



Cuando se quiere implementar un sistema que reciba documentación firmada, las opciones son muchas. Una primera

aproximación sencilla es la contraparte de la anterior: solicitar al usuario que firme su documento fuera del sistema y que lo cargue a este ya firmado. Si bien a priori puede ser percibido como incómodo desde el punto de vista funcional, existen casos de uso donde es la opción más directa a implementar, como el que se ve en la siguiente figura.

Se trata de una empresa u organismo del Estado (Sistema de Gestión) que requiere que un usuario realice cierta operativa, posiblemente, además en nombre de una empresa u organismo, y tiene como parte de ella la inclusión de informes técnicos firmados por terceros (planos, balances, informes de auditorías, revisiones, etc). Que el gestor reciba los documentos firmados en papel y los escanee no es una opción sólida, porque se pierde el carácter de original; por eso, para un uso 100% digital se deberían pedir los informes digitales. Si bien el sistema podría implementar la funcionalidad de que cada uno de los terceros suba su propio informe y lo firme digitalmente, esto sería incómodo y difícil de coordinar. Resulta natural, entonces, que el gestor pida a cada tercero que remita su informe ya firmado por mail u otra vía y que sea este quien se encargue de la gestión ante la empresa u organismo, remitiendo en el proceso los informes originales firmados digitalmente por cada uno de los profesionales. Si se utiliza formato de documentos PDF, cada tercero puede realizar la Firma Digital en su PC de forma sencilla con su Cédula de Identidad Digital u otro dispositivo criptográfico usando Adobe Reader de la misma forma que ya fue mencionada para el caso de que una persona enviaba una carta o solicitud firmada digitalmente.

Más allá de los ya mencionados casos en donde lo más conveniente es que el usuario firme en su PC y envíe por mail, existen muchos casos en donde esto no es cómodo; por ejemplo, cualquier sistema que modele un proceso de negocio donde la firma de un documento es un paso más del proceso. En esos casos, lo más conveniente es que el propio sistema implemente la firma como parte de él; y, para ello, hay varias alternativas.

PKCS#11

PKCS#11 es una interfaz estándar para el acceso a dispositivos criptográficos y, por lo tanto, es una alternativa natural para implementar Firma Electrónica Avanzada con cualquier token o con la Cédula de Identidad Digital. Por más información para su implementación se recomienda ver *Autenticación y Firma Avanzada a través de PKCS#11*

Servicio de Firma con Cédula de Identidad Digital

Agesic cuenta con una plataforma que ofrece firma con Cédula de Identidad digital como servicio. Ofrece facilidad de integración con aplicaciones por utilizar protocolos estándar y maduros para la integración y, al basarse en plugins y componentes de browser y ser específicamente diseñada para usar la CI uruguaya, provee una experiencia de usuario de gran calidad. Su funcionamiento esencial es sencillo: cuando una aplicación web quiere recoger la Firma Electrónica Avanzada de una persona, en lugar de realizarla de manera local, la redirige a esta plataforma, delegando la firma en este servicio. La aplicación recibe el documento ya firmado y esta puede efectuar su validación para asegurarse que todo está correcto.

Las restricciones para su uso son que aplica solamente a aplicaciones web y que los plugins son sólo para versiones desktop de los navegadores. Por lo tanto, si bien se tiene un alto nivel de compatibilidad en PC y navegadores tradicionales, de momento no soporta los entornos Mobile. Las ventajas que ofrece usar esta plataforma son la simplicidad en la implementación y la experiencia del usuario final, por lo que es la opción recomendada siempre salvo cuando el entorno de los usuarios no sea compatible y/o cuando no se quiera incurrir en dependencia de un servicio externo. En esos casos se recomienda implementar una solución propia, para lo cual se puede ir por alguna de las soluciones alternativas ya mencionadas.

Su documentación puede encontrarse en la página dedicada al*Servicio de Firma con la Cédula de Identidad Digital.*

Servicio de Autenticación con Cédula de Identidad Digital

Dado que el sistema de llaves asimétricas y certificados en que se basa la Firma Electrónica Avanzada también puede ser utilizado para la autenticación, muchas de las técnicas mencionadas pueden ser utilizadas en forma prácticamente análogas para realizar la autenticación de usuarios. En particular, para la autenticación utilizando la Cédula de Identidad Digital, Agesic cuenta con un servicio análogo al servicio de firma de la sección anterior.

Su documentación puede encontrarse en la página dedicada al*Servicio de Autenticación con la Cédula de Identidad Digital.*

Firma de Persona Jurídica

Por ser una Firma Electrónica Avanzada, la firma de persona jurídica también se basa en el uso de una llave privada para firmar documentos y una llave pública extraída del certificado para validar. La diferencia es que el sujeto registrado en el certificado es una persona jurídica, es decir, una organización. En el mundo físico, pensar en firma de organización carece de sentido, puesto que la organización en sí misma no tiene voluntad, y siempre son personas físicas las que actúan en su nombre. Si bien en el mundo digital esto se mantiene y las actuaciones fundamentales de las organizaciones siguen siendo realizadas por personas físicas en su nombre, existe la posibilidad de que los sistemas de la organización realicen actuaciones en forma autónoma, sin intervención humana directa. Y para dotar de seguridad y garantías jurídicas esos actos, se crea y reconoce la Firma Electrónica Avanzada de persona jurídica como tal. Casos de uso de firma de persona jurídica incluyen la facturación electrónica, la emisión de constancias, la firma de documentos para certificar que se controló alguna regla de negocio explícitamente y la autenticación de transacciones que se realizan automáticamente, entre otros.

En este escenario, las claves y certificados de la persona jurídica se instalarán en el equipo en que se encuentre la aplicación que los acceda, que será típicamente en un servidor físico o virtual, y la o las aplicaciones accederán al certificado y las llaves en forma automática autenticándose debidamente. En este sentido, el acceso a las llaves por parte de las aplicaciones se vuelve mucho más directo que en el caso de persona física, ya que en este caso siempre se estará ejecutando en el mismo equipo donde se encuentra el dispositivo de creación de firmas, sea este de software o de hardware.

Tanto en caso de software como de hardware, se recomienda usar las llaves utilizando bibliotecas estables y maduras de la propia plataforma en la que se esté trabajando. Si se trata de un dispositivo de software, es decir, un archivo PKCS#12, se debe utilizar una biblioteca para acceso a PKCS12 y proveer la contraseña de acceso, mientras que si se trata de un dispositivo de hardware, se debe usar un pkcs11 Provider instanciado con el driver del dispositivo (que debe estar previamente instalado) y el correspondiente PIN de acceso. Si se tratase de un HSM, es posible que además se requiera su activación fuera de banda, que es específica de cada marca y modelo, y se determina al momento de su instalación.

Estándares de firma en documentos

La Firma Electrónica Avanzada es un mecanismo que agrega información al documento para dotarlo de, mínimamente, capacidad de verificación de integridad e identificación del firmante. Es importante este detalle, debido a que los datos agregados no previenen que el documento sea modificado, sino que hacen que las modificaciones sean detectables; es decir, lo hacen "susceptible de verificación". Para que esa verificación sea posible, la firma debe ser expresada de una forma que pueda ser interpretada por quien la verifica y es por eso que existen los estándares de Firma Digital. Cada tipo de documento tiene un conjunto de extensiones estándar que especifican cómo deben ser expresados los resultados de las firmas y cada validador de firmas para esos documentos usa esas mismas extensiones para efectuar las validaciones.

Es así que existen los siguientes estándares de firma que vale la pena destacar:

- PDF Signature.
- PAdES.
- XML Signature.
- XAdES.
- CMS.
- CAdES.

Validación de Firma Electrónica Avanzada

La validación de la firma es el proceso mediante el cual un receptor de un documento se asegura de su integridad y no repudio por parte del firmante y sigue una serie de pasos bien definida, aunque su implementación concreta pueda diferir un poco según el entorno de implementación. En cualquier caso, una condición necesaria para poder validar la firma es contar con el certificado del firmante. Los estándares de firma anteriormente mencionados establecen campos específicos donde se coloca el certificado al momento de realizar la firma. Análogamente, el validador puede obtener el certificado de allí. Si esta opción no fuese posible, el validador debe obtener el certificado por alguna otra vía, como ser un directorio público o tenerlo previamente porque fue entregado fuera de banda. De todas formas, lo más común y recomendable es que se firme el documento dejando en el mismo una copia del certificado y que el validador lo obtenga de allí, las garantías que da el esquema PKI a través de las Autoridades Certificadoras de confianza permite esta flexibilidad.

Validación de la integridad del documento

El primer paso consiste en verificar que el documento o la firma no fueron alterados en el intercambio, ya que si esta propiedad no se cumple, el resultado todos los controles subsiguientes es dudoso. Para hacerlo, se debe:

- Obtener la llave pública del firmante de su certificado y utilizarla para descifrar la firma, lo que da como resultado el hash del documento que calculó el firmante.
- Volver a calcular aparte el hash del documento recibido nuevamente. Si coincide con el hash descifrado previamente, entonces se puede estar seguro de que el documento no fue modificado por una tercera parte entre su firma y la validación, obteniéndose así la garantía de identidad buscada.

Tanto las firmas como los hashes pueden ser realizados con múltiples algoritmos, aplicando transformaciones intermedias y demás. Si las dos partes no aplican las mismas transformaciones y los mismos algoritmos, los resultados de las operaciones no serán los mismos y, por lo tanto, no será posible verificar correctamente una firma aún cuando no se haya modificado nada efectivamente. Para esto, los estándares de firma especifican también qué mecanismos se utilizaron en cada firma en particular, y es otro motivo por el cual siempre se recomienda utilizarlos en lugar de hacer una solución propia.

Validación de la identidad del firmante

Una vez validada la integridad del documento, lo que se sabe es que este no fue modificado en tránsito y que fue firmado con el certificado que se utilizó para validar. Lo que no se sabe aún es si los datos del firmante que figuran en el certificado son ciertos, es decir, *si el certificado es válido*.

Para garantizar la validez de un certificado, se deben verificar cuatro propiedades:*la vigencia, la autenticidad, el estado de revocación y la cadena de confianza.*

Vigencia

El primer control que se debe hacer al verificar cualquier certificado es su validez, ya que se emiten por un período acotado. Si la fecha de validación está fuera del rango de fechas de validez que indica el certificado, esta debe fallar y no continúa.

Autenticidad

El certificado es en sí mismo un documento electrónico que contiene la llave pública del firmante y sus datos. Es firmado por una autoridad certificadora (CA) que, mediante dicha firma, certifica que registró correctamente al sujeto identificado en el certificado, dando fe de que esos son efectivamente sus datos y su llave pública. Se debe entonces validar la firma del certificado del firmante utilizando el certificado del emisor, es decir, de la autoridad certificadora, siguiendo el mismo procedimiento que se siguió para el documento y teniendo en cuenta que el estándar x.509 en el cual se elaboran los certificados estandariza el modo de firmarlos. Si se valida correctamente, se puede estar seguro de que el firmante pasó por el registro de esa autoridad certificadora, resultando en un certificado auténtico. Típicamente, los certificados a validar cuentan con una extensión llamada *Authority Information Access*, que contiene la URL de Internet donde se publica el certificado de la CA que lo emitió, y de esa forma cualquier validador puede descargarlo. Esta extensión es obligatoria para los certificados emitidos por prestadores acreditados de Uruguay, por lo que se puede asumir que estará presente cuando se validan firmas electrónicas avanzadas.

Estado de Revocación

Las CA emiten los certificados por un período de validez dado, como ya fue visto, y además una vez que son emitidos la llave privada y el certificado quedan en control exclusivo del usuario final. Si este usuario incurriera en un mal uso del certificado, faltará a las condiciones de negocio acordadas o, peor aún, si su clave privada fuera comprometida por un tercero que pueda comenzar a usar el certificado en su nombre, la CA no tiene forma de eliminar esa llave privada o certificado. Es por eso que lo que sí hace es publicar uno o varios servicios de tipo "lista negra" donde se publican los certificados cuya validez fue terminada prematuramente o, dicho de otra manera, los certificados revocados.

El mecanismo más básico para consultar el estado de revocación es la CRL (*Certificate Revocation List*), que consiste simplemente en un archivo firmado por la CA (para garantizar su autenticidad) que contiene los números de serie de los certificados emitidos por ella que se encuentran revocados. Para verificar si un certificado dado está revocado, basta con obtener la CRL de la CA correspondiente, validar su firma con el certificado de la CA y verificar si el número de serie del certificado se encuentra en la lista. Si está, se encuentra revocado; y si no lo está, no. Los certificados de usuario final, además, cuentan con una extensión llamada *CRL Distribution Points*, que contiene el o las URL de Internet donde se publica la CRL de la CA, que es la que se debe consultar para ver el estado de revocación. Esta extensión es obligatoria para los certificados emitidos por prestadores acreditados de Uruguay. La implementación de la CRL también lo es, por lo que si bien es un método precario, en la mayoría de los escenarios es suficiente y resulta en una validación muy sencilla de implementar.

Otro método de verificación de estado de revocación es el OCSP (*Online Certificate Status Protocol*), que consiste en un servicio publicado por la CA, que recibe consultas por un certificado dado a través de su número de serie. Responde un mensaje especial del protocolo indicando si se encuentra revocado, si no lo está o si no puede contestar en este momento. Cabe destacar que estos mensajes de respuesta son firmados por la CA o por una *Autoridad de Validación* de esta para garantizar su autenticidad. OCSP es un protocolo más flexible que la CRL y que escala mejor en escenarios donde las CA manejan muchas revocaciones, dado que evita descargar e interpretar listas largas; como contrapartida, no todas las CA lo implementan (no es obligatorio) y su implementación en validadores es más compleja que mediante CRL. Si implementa OCSP, la CA incluirá en los certificados que emita la URL de consulta de dicho servicio dentro de la extensión *Authority Information Access.*

Cadena de Confianza

Luego de ejecutadas las validaciones anteriores se sabe que el certificado está dentro de su período de validez, que no está revocado, y que fue emitido por una CA bien identificada. Lo que potencialmente se desconoce aún es si la CA misma es auténtica, es decir, si su certificado es válido. Para esto, se debe realizar el mismo proceso de validación sobre el certificado de la CA con el certificado de su CA emisora, en un proceso iterativo que repite los mismos pasos de validación de vigencia, autenticidad y estado de revocación. Se va así construyendo una cadena de certificados digitales, donde un certificado "hijo" es validado con su certificado "padre" de otra CA, hasta llegar al caso base en que se encuentra una CA que emitió su propio certificado, es decir, una CA Autofirmada, una raíz de confianza. Esta cadena verificable de certificados se denomina *Cadena de Confianza.*

Un certificado raíz es de confianza solo si se lo tiene previamente instalado en el software de validación como tal, sea porque viene de fábrica o porque los sistemas operativos, navegadores web, editores de PDF y sistemas similares que realizan procesamiento de certificados cuentan con sus propios *stores* de confianza, con las raíces que son internacionalmente reconocidas. La raíz nacional de Uruguay (ACRN) se encuentra en algunos de estos *stores* y está en proceso de inclusión en

varias más. Si se va a instalar una raíz de confianza, se debe tener mucho cuidado de que sea un certificado auténtico y de confianza, porque de lo contrario se está abriendo la puerta a que se validen cadenas de confianza fraudulentas.

En el contexto de la Firma Electrónica Avanzada, las autoridades certificadoras acreditadas son Abitab (ID Digital), El Correo y el Ministerio del Interior (Cédula Digital). Estas autoridades son emitidas por la Autoridad Certificadora Raíz Nacional (ACRN), operada por Agesic. El certificado de la ACRN puede ser descargado aquí.

A modo de resumen, se debe realizar la siguiente cadena de validaciones:

1. Validar firma de documento con certificado del firmante.
2. Validar certificado del firmante (vigencia, autenticidad y estado de revocación) usando certificado de AC.
3. Validar certificado de AC (vigencia, autenticidad y estado de revocación) usando certificado de ACRN.
4. Validar certificado de ACRN verificando que sea el mismo que se tiene instalado previamente.

Los puntos 2 al 4 conforman lo que se denomina "validación de la Cadena de Confianza".

Integración con Servicio de Firma Digital

El Servicio de Firma Digital (DSS) está basado en el estándar OASIS DSS, con binding HTTP Post. Esto permite interactuar con el usuario a través de redirecciones del navegador, no requeriendo resoluciones de artefactos punto a punto. El Servicio de Firma Digital es propiedad de Agesic y se encuentra disponible en la Plataforma de Gobierno Digital. Para la integración con el DSS de Agesic, se puede utilizar cualquier tecnología o conjunto de tecnologías que sean capaces de manejar los elementos XML *SignRequest* y *SignResponse* del protocolo DSS.

Arquitectura

Descripción de los actores involucrados:

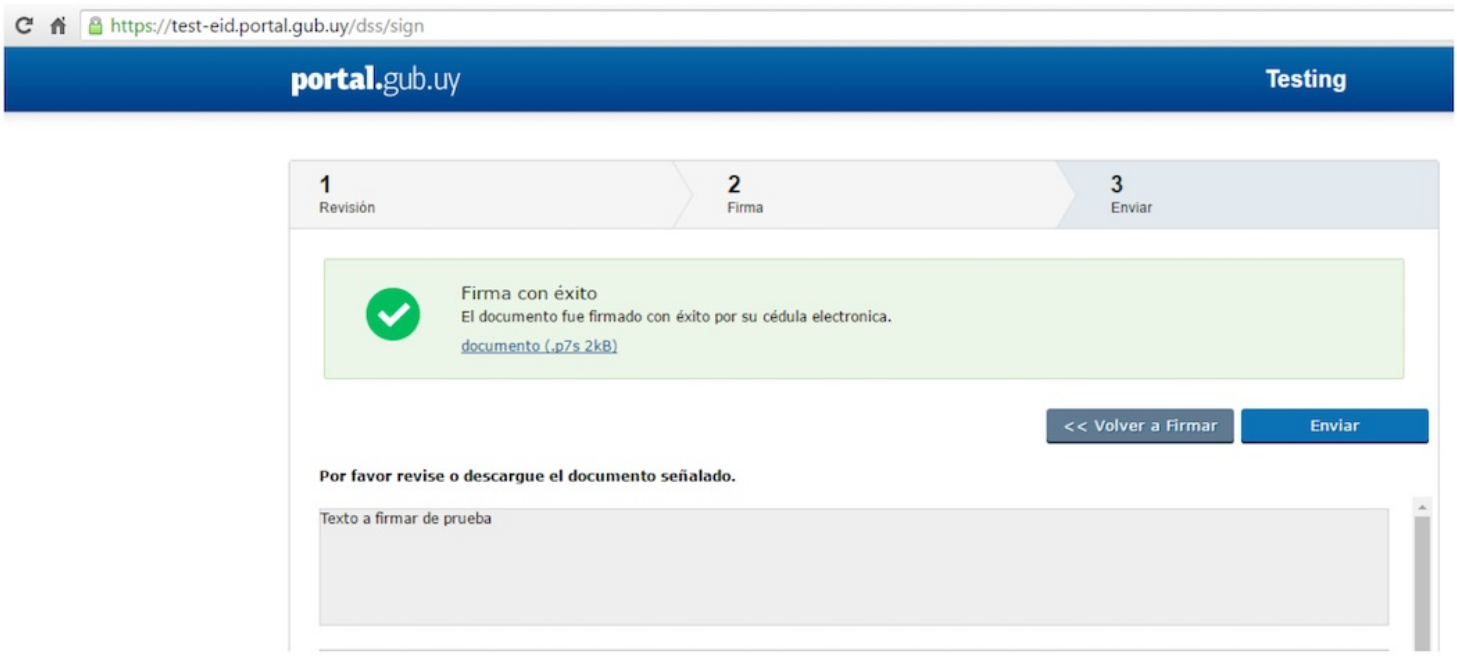
- Servidor DSS dentro de la Plataforma de Gobierno Digital.
- Aplicaciones de los organismos que deberán integrarse con el Servidor DSS a través de un componente a desarrollar denominado Cliente DSS.
- Usuarios con Cédula de Identidad Digital con acceso a las aplicaciones de los organismos integradas con el Servidor DSS

Descripción del servidor DSS

El servidor DSS disponible en la Plataforma de Gobierno Digital guía al usuario durante el proceso de firma con la Cédula de Identidad Digital mediante una aplicación web y plugin que resuelven la comunicación con dicho documento.

La participación del Servidor DSS en el proceso de firma comienza cuando el usuario de la Cédula de Identidad Digital es redireccionado por la aplicación integrada por el organismo.

Una vez que el usuario hace clic en “Enviar”, finaliza el proceso de firma y la aplicación web del Servidor DSS lo redirige nuevamente hacia la aplicación del organismo proveniente.



En esta etapa final la aplicación del organismo contendrá la información necesaria para validar que el proceso de firma se haya realizado correctamente.

Descripción del cliente DSS

Se denomina "cliente DSS" a la aplicación del organismo o componente de la aplicación que deberá manejar los elementos necesarios para utilizar el Servicio de Firma Digital en la Plataforma de Gobierno Digital.

El cliente DSS deberá generar el elemento XML *SignRequest* que el usuario de la Cédula de Identidad Digital utilizará para iniciar el proceso de Firma Digital en el Servidor DSS.

Luego de finalizado el proceso de firma en el Servidor DSS, la aplicación del organismo recibe el elemento XML *SignResponse,* que contiene información sobre este proceso.

La validación del elemento XML *SignResponse* obtenido como resultado del proceso de firma deberá ser realizada por el cliente DSS.

A modo de resumen, el elemento *SignRequest* contiene el documento a firmar por el usuario de la Cédula de Identidad Digital y el elemento SignResponse contiene el resultado de la firma.

Ambos elementos XML deberán ser firmados y validados mediante claves RSA públicas y privadas. El elemento *SignRequest* deberá ser firmado por una clave privada en custodia del organismo y validado por el Servidor DSS mediante la clave pública y certificado del organismo. El elemento XML *SignResponse* que envía el Servidor DSS es firmado con la clave privada en custodia de este y validado por el organismo utilizando la clave pública y el certificado correspondientes.

Flujos de la Firma Digital

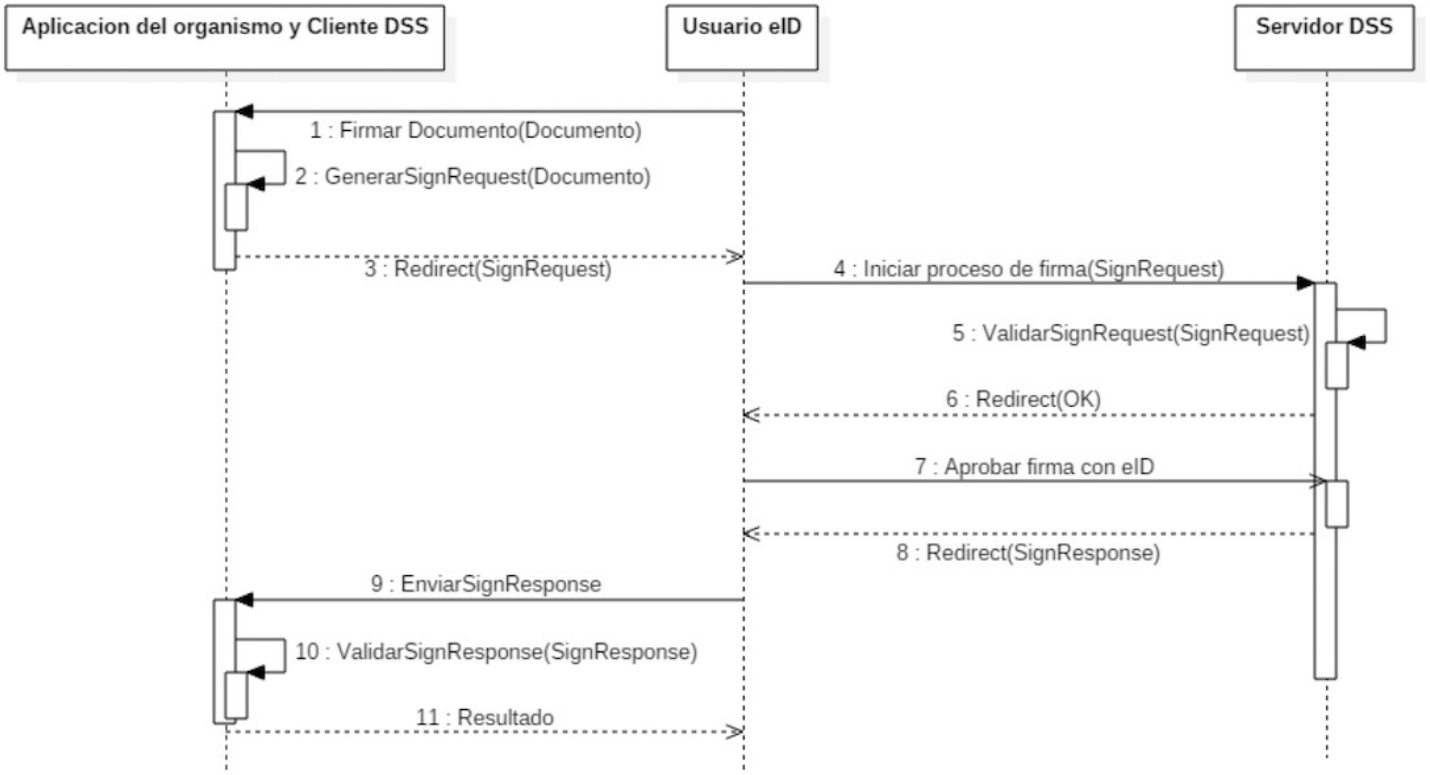
El flujo de una Firma Digital utilizando el Servicio de Firma Digital se compone de pasos independientes; es un proceso modular al que se le pueden agregar pasos adicionales.

Un caso de uso estándar es el siguiente:

1. El usuario ingresa a la aplicación del organismo integrada con el Servicio de Firma Digital e indica que desea realizar la Firma Digital de un documento.
2. La aplicación del organismo mediante el componente Cliente DSS construye firma el elemento *SignRequest,* que contiene el documento indicado por el usuario del eID.
3. La aplicación del organismo redirecciona al usuario hacia el portal del Servidor DSS con el elemento *SignRequest* generado.
4. El Servidor DSS recibe una petición HTTP POST del usuario que contiene el elemento *SignRequest*.
5. El Servidor DSS valida la firma y estructura del elemento *SignRequest* antes de continuar con el proceso de firma en la aplicación web.
6. En caso de que el *SignRequest* sea válido, el Servidor DSS redirecciona al usuario a la aplicación web, que lo guiará en el proceso de firma.
7. El usuario valida la información en pantalla, acepta la instalación del plugin sConnect en el navegador y firma el

documento utilizando la Cédula de Identidad Digital, siguiendo los pasos indicados en la aplicación web.

8. Luego de finalizada la firma del documento, el Servidor DSS redirige al usuario hacia la aplicación del organismo. La URL hacia la cual es redirigido es la contenida en el elemento *SignRequest* en el tag ML"*<SignResponseConsumerURL>*…"
9. La aplicación del organismo recibe una petición HTTP POST del usuario que contiene el elemento *SignResponse* devuelto por el Servidor DSS.
10. El Cliente DSS valida que la firma del elemento *SignResponse* sea correcta.
11. El resultado e información de la firma obtenidos del elemento *SignResponse* es desplegado al usuario en la aplicación del organismo.



Implementación del cliente DSS

El cliente DSS, como componente de software integrado en la aplicación web que desee utilizar los Servicios de Firma Digital, deberá **recibir los documentos a firmar y formatos para construir el '*SignRequest*'.** Deberá recibir las respuestas del Servidor DSS "*SignResponse*" y obtener información sobre el resultado de la firma realizada.**Tendrá acceso a la clave privada utilizada para firmar el "*SignRequest*" y ser, de esta forma, un cliente DSS de confianza para el Servidor DSS.**

Dependiendo de la infraestructura y las tecnologías del organismo, el cliente DSS puede considerarse de las siguientes formas:

1. Único componente utilizado por las aplicaciones del organismo.
2. Integrado en cada aplicación del organismo.

Siendo las características del cliente DSS las mismas en ambos casos, para el primer caso las aplicaciones del organismo que deseen hacer uso del Servicio de Firma Digital se comunican con la aplicación cliente DSS, por ejemplo, a través de servicios Web para el armado del *SignRequest* y validación del *SignResponse*.

En el segundo caso, las aplicaciones del organismo tienen integradas las características del cliente DSS; se puede considerar cada aplicación como un cliente DSS en sí misma.

Contenido *del SignRequest:*

Dentro de la estructura XML del *SignRequest* se pueden destacar los siguientes elementos:

- ***ClaimedIdentity:*** Deberá indicar la identidad del Cliente DSS y contener la URL hacia la cual el usuario es redirigido por el Servidor DSS al finalizar el proceso de firma. La URL es sensible a mayúsculas y minúsculas. En el ejemplo de la imagen son los valores "clienteDSSOrganismo" y la URL "http://clienteDSSOrganismo.uy/RespuestaDSS"

```
<dss:ClaimedIdentity>
  <dss:Name Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
clienteDSSOrganismo</dss:Name>
  <dss:SupportingInfo>
```

```
    <SignResponseConsumerURL xmlns="urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo">
http://clienteDSS-Organismo.uy/RespuestaDSS</SignResponseConsumerURL>
  </dss:SupportingInfo>
```

- ***InputDocuments:*** Este elemento deberá contener el documento sobre el cual el usuario del eID va realizar la firma. Dependiendo del formato (texto, PDF, XML), puede variar el contenido. En el ejemplo que se ve a continuación, el documento a firmar es un texto plano representado en el formato base64. El texto a firmar: *"Texto a firmar para el taller"*.

```
<dss:InputDocuments>
  <dss:Document>
    <dss:Base64Data MimeType="text/plain">VGV4dG8gYSBmgCFyYSBlbCB0YWxsZXIu
</dss:Base64Data>
  </dss:Document>
</dss:InputDocuments>
```

Contenido*del SignResponse:*

Dentro de la estructura XML del *SignResponse* se pueden destacar los siguientes elementos:

- ***Result:*** El elemento Result contiene información sobre el resultado de firma de los documentos en el servidor DSS. En el ejemplo siguiente se puede observar un caso de éxito de firma en todos los documentos.

```
<Result>
  <ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</ResultMajor>
  <ResultMinor>urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments
</ResultMinor>
</Result>
```

- ***SignatureObject:*** El elemento *SignatureObject* contiene la Firma Digital del documento enviado. El ejemplo siguiente contiene codificada en base64 la firma para el texto plano enviado en base64 en el *SignRequest*.

```
<SignatureObject>
  <Base64Signature>MIIHxgY...</Base64Signature>
</SignatureObject>
```

Consideraciones de seguridad

Firma del *SignRequest*:

La clave privada que utilizara el Cliente DSS para firmar los*SignRequest* debe ser almacenada teniendo en cuenta los recaudos de seguridad necesarios, ya que aquella permite autorizar y autenticar las solicitudes de firma provenientes del organismo hacia el Servidor DSS.

Se recomienda utilizar un dispositivo del tipo HSM o Token que cumpla con el estándar FIPS 140-2.

El certificado digital correspondiente a la clave privada generada en el dispositivo seguro será entregado a Agesic para ser dado de alta en el Servidor DSS.

Validación del *SignResponse*:

Luego de enviado el *SignRequest* y finalizado el proceso de firma, el usuario es redirigido a la aplicación del organismo con el elemento *SignResponse*.

El elemento *SignResponse* es firmado con la clave privada del Servidor DSS y debe ser validado por el Cliente DSS utilizando el certificado digital del Servidor DSS.

Puede descargar el certificado del servidor aquí.

Utilización del Servicio de Firma Digital

Para hacer uso del Servicio de Firma Digital, además de implementar los cambios necesarios en las aplicaciones, el organismo y Agesic deben intercambiar cierta información necesaria para generar la relación de confianza:

- El organismo debe proveer:
  - URL de retorno *SignResponseConsumerURL* para las redirección y envió del*SignResponse*.
  - Certificado digital con el cual el cliente DSS firmará los*SignRequest* DSS
- Agesic proveerá:
  - URL del Servicio de Firma Digital donde recibe las solicitudes de firma.
  - Certificado Digital utilizado por el DSS para firmar los*SignResponse*

Anexo Técnico

Perfiles DSS

El servidor DSS implementa los perfiles Gemalto DSS AdES y Demo Profile (DP) para la Firma Digital.

El perfil Gemalto DSS AdES como extensión de OASIS DSS AdES define la especificación de los elementos *SignRequest* y *SignResponse* para la firma de documentos bajo los formatos CAdES, PAdES y XAdES.

El perfil Demo Profile (DP) define la especificación de los elementos *SignRequest* y *SignResponse* para la firma de CMS y plantilla PDF con datos.

La aplicación del organismo entonces deberá construir las solicitudes de firma *SignRequest* acorde a sus necesidades de negocio. Por ejemplo, en el caso de Firma Electrónica Avanzada de documentos PDF, como pueden ser los contratos, la construcción del elemento *SignRequest* debe realizarse ajustada a la especificación definida por el perfil PAdES.

Descripción detallada de los elementos XML SignRequest y SignResponse

El DSS está basado en el estándar OASIS DSS con bindings HTTP POST, sin resolución directa de artefactos.

El proceso de firma comienza con el *SignRequest* enviado por el usuario del documento de identidad digital a través de un HTTP POST en formato base64 y finaliza con la verificación del SignResponse devuelto por el servidor DSS también en formato base64.

El *SignRequest* contiene el elemento a firmar, el perfil seleccionado para la firma y la dirección URL SignResponseConsumerURL, donde el usuario será redireccionado por el servidor DSS para enviar el *SignResponse* recibido.

Ambos elementos XML (*SignRequest*, *SignResponse*) son firmados con una clave privada y validados con su correspondiente Certificado Digital.

La construcción y firma del *SignRequest* la realiza el cliente DSS quien posee una clave privada y Certificado Digital de confianza para el servidor DSS. De esta forma, el servidor DSS puede autenticar y autorizar a sus diferentes clientes.

El *SignResponse* contiene información sobre el resultado de la firma realizada y el elemento firmado. La aplicación del organismo deberá realizar una validación del elemento *SignResponse* utilizando el certificado electrónico del servidor DSS.

Estructura del SignRequest y SignResponse

El Servicio de Firma Digital utiliza un binding basado en HTTP POST.

- El *SignRequest* se envía al servidor en una petición HTTP POST en formato application/x-www-form-urlencoded. El request debe contener un único parámetro *SignRequest*, con el valor del XML *SignRequest* codificado en Base64.
- El *SignResponse* se envía al servidor en un request HTTP POST en formato application/x-www-form-urlencoded. El request debe contener un único parámetro *SignResponse*, con el valor del XML *SignResponse* codificado en Base64.

Especificación de los elementos XML del *SignRequest*, *SignResponse* para el perfil definido como Demo Profile:

Signature

Namespace: http://www.w3.org/2000/09/xmldsig#

El *SignRequest* debe estar firmado digitalmente. Esta firma es representada por el elemento *Signature*, que debe estar presente en el elemento OptionalInputs.

InputDocuments

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento InputDocuments es utilizado para enviar documentos a firmar al servidor DSS. Puede contener los siguientes elementos (exactamente una vez):

- Document.
- DocumentHash.
- El elemento TransformedData no está soportado.

Document

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento Document es utilizado para transportar el documento completo a firmar.

Puede contener los siguientes elementos:

- Base64Data.
- Other.
- Los elementos ID, RefURI, RefType y SchemaRefs no deben estar presentes.

Base64Data

El elemento Base64Data representa datos codificados en Base64. Soporta el mimetype text/plain.

Other

El elemento Other debe contener un elemento TemplateWithData.

TemplateWithData

El elemento TemplateWithData contiene dos subelementos:

- Data.
- Template.
- Cada uno de estos elementos debe contener el atributo mimeType.

En el elemento Data, el atributo mimeType debe contener el valor application/json. El elemento debe contener un objeto codificado en Base64 en formato JSON.

En el elemento Template, el atributo mimeType debe contener el valor application/pdf.

El elemento debe contener un documento PDF codificado en Base64.

DocumentHash

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento DocumentHash contiene los siguientes elementos:

- Transforms debe estar vacío.
- DigestMethod identifica el algoritmo de digest utilizado para calcular el hash del documento del lado del cliente. Los siguientes valores son soportados:
  - http://www.w3.org/2000/09/xmldsig#sha1
  - http://www.w3.org/2001/04/xmldsig-more#sha224
  - http://www.w3.org/2001/04/xmlenc#sha256
  - http://www.w3.org/2001/04/xmldsig-more#sha384
  - http://www.w3.org/2001/04/xmlenc#sha512
- DigestValue contiene el valor del hash del documento.

SignatureType

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento *SignatureType* indica el tipo de firma o timestamp a producir. Los valores soportados por el Demo Profile son:

- urn:ietf:rfc:3369 – Refiere a la firma CMS (RFC 3852)
- urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo:signaturetype:pdf – Refiere a la firma PDF

El elemento *SignatureType* debe estar presente en el elemento OptionalInputs.

ClaimedIdentity

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento ClaimedIdentity indica la identidad del cliente realizando el request. Debe estar presente en el elemento OptionalInputs.

SignResponseConsumerURL

Namespace: urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo

El elemento *SignResponseConsumerURL* determina la URL a donde deberá ser enviado el *SignResponse*. Este elemento debe estar presente en el elemento OptionalInputs/ClaimedIdentity/SupportingInfo.

IncludeEContent

El elemento IncludeEContent es opcional. En caso de estar presente, el servidor debe agregar el documento enviado en el *SignResponse*. El elemento no debe estar presente si el *SignRequest* contiene un elemento DocumentHash.

SignatureMethod

El elemento *SignatureMethod* representa el método de firma solicitado. El único valor posible es SmartCard, especificando que la firma digital debe ser creada utilizando una SmartCard.

El perfil Gemalto DSS AdES es una extensión del perfil OASIS DSS AdES, con definición de un nuevo binding y un conjunto de limitaciones.

El perfil OASIS AdES es definido por la organización OASIS y puede ser descargadoaquí

El perfil define requests para CAdES, PAdES y XAdES, la extensión de Gemalto implementa el perfil con las siguientes limitaciones:

AdES-BES

El nivel CAdES-BES está soportado. El valor del atributo Content type está preconfigurado en el servidor, existiendo un único valor disponible.

El nivel PAdES-BES está soportado. La apariencia de la firma está preconfigurada en el servidor, habiendo un único valor disponible.

El nivel XAdES-BES está soportado. Los elementos AllDataObjectsTimeStamp y IndividualDataObjectsTimeStamp no están soportados.

AdES-EPES

El nivel EPES está soportado, un valor de policy puede ser preconfigurado en el servidor.

AdES-T

Los niveles CAdES-T y PAdES-T están soportados. El nivel XAdES-T no está soportado.

Elementos opcionales

El elemento ClaimedIdentityes usado para la identificación del Service Provider.

El elemento KeySelector no está soportado.

Especificación de los elementos XML del *SignRequest*, *SignResponse* para el perfil definido como Gemalto DSS AdES profile:

Signature

Namespace: http://www.w3.org/2000/09/xmldsig#

El *SignRequest* debe estar firmado digitalmente. Esta firma es representada por el elemento*Signature*, que debe estar presente en el elemento OptionalInputs.

ClaimedIdentity

Namespace: urn:oasis:names:tc:dss:1.0:core:schema

El elemento ClaimedIdentity indica la identidad del cliente realizando el request. Debe está presente en el elemento OptionalInputs.

SignResponseConsumerURL

Namespace: urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo

El elemento *SignResponseConsumerURL* determina la URL a donde deberá ser enviado el*SignResponse*. Este elemento debe estar presente en el elemento OptionalInputs/ClaimedIdentity/SupportingInfo.

SignatureMethods

Namespace: urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo

El elemento *SignatureMethods* especifica los métodos de firma permitidos para procesar el*SignRequest*. El elemento debe

contener al menos un método de firma representado por el elemento *SignatureMethod*. Si hay más de un método presente, la aplicación debe permitir al usuario seleccionar uno.

El elemento *SignatureMethods* debe estar presente en el elemento OptionalInputs.

SignatureMethod

El elemento *SignatureMethod* representa el método de firma solicitado. El único valor posible es SmartCard, especificando que la Firma Digital debe ser creada utilizando una smart card.

Ejemplo de SignRequest y SignResponse

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<dss:SignRequest xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
Profile="urn:gemalto:dss:profiles:demo" RequestID="0.25888494712651455">
  <dss:OptionalInputs>
    <dss:ClaimedIdentity>
      <dss:Name Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
clienteDSSOrganismo</dss:Name>
      <dss:SupportingInfo>
        <SignResponseConsumerURL xmlns="urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo">
        http://clienteDSSOrganismo.uy/RespuestaDSS</SignResponseConsumerURL>
      </dss:SupportingInfo>
    </dss:ClaimedIdentity>
    <dss:IncludeEContent/>
    <dss:SignatureType>urn:ietf:rfc:3369</dss:SignatureType>
    <demo:SignatureMethods xmlns:demo="urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo">
      <demo:SignatureMethod>SmartCard</demo:SignatureMethod>
    </demo:SignatureMethods>
    <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
      <SignedInfo>
        <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments"/>
        <SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
        <Reference URI="">
          <Transforms>
            <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
          </Transforms>
          <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
          <DigestValue>lxlSM+0yzDWG6CPJ8gO9DU33KePyC5fE3g7fl2OZFTQ=</DigestValue>
        </Reference>
      </SignedInfo>
      <SignatureValue>rFi5KnOnRT3KpVbirp...</SignatureValue>
      <KeyInfo>
        <X509Data>
          <X509Certificate>MIIDezCCAmO...</X509Certificate>
        </X509Data>
      </KeyInfo>
    </Signature>
  </dss:OptionalInputs>
  <dss:InputDocuments>
    <dss:Document>
      <dss:Base64Data MimeType="text/plain">VGV4dG8gYSBmgcGFyYSBlbCB0YWxsZXIu
</dss:Base64-Data>
    </dss:Document>
  </dss:InputDocuments>
</dss:SignRequest>
```

Ejemplo de *SignRequest*:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SignResponse xmlns="urn:oasis:names:tc:dss:1.0:core:schema" xmlns:ns2="http://www.w3.org/2000/09/xmldsig#"
xmlns:ns3="urn:oasis:names:tc:SAML:1.0:assertion" xmlns:ns4="urn:oasis:names:tc:dssx:1.0:profiles:gemalto:demo"
Profile="urn:gemalto:dss:profiles:demo" equestID="0.2108839114169051">
  <Result>
    <ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</ResultMajor>
    <ResultMinor>urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments</ResultMinor>
  </Result>
  <OptionalOutputs>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
        <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
        <ds:Reference URI="">
          <ds:Transforms>
            <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
            <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments"/>
          </ds:Transforms>
```

```
            <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
            <ds:DigestValue>qg53vYht8ghiFHJWn4LxcAiAAHhlEPoCIDTokbJpwTg=</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>xZxe6N...</ds:SignatureValue>
        <ds:KeyInfo>
          <ds:X509Data>
            <ds:X509Certificate> MIIGHDCC...</ds:X509Certificate>
          </ds:X509Data>
        </ds:KeyInfo>
      </ds:Signature>
    </OptionalOutputs>
    <SignatureObject>
      <Base64Signature>MIIHxgY...</Base64Signature>
    </SignatureObject>
  </SignResponse>
```

Recursos de interés

Recursos y materiales técnicos Identidad Electrónica

Recursos y materiales técnicos de Firma Digital

Integración de aplicación DSS a ASPNET