Manual de Programación de Web Services

Integra 2.2.4\_000

|  |  |
| --- | --- |
| Documento nº: | @Firma-Integra-WS-MAN |
| Revisión: | 010 |
| Fecha: | 29-01-2024 |
| Período de retención: | Permanente durante su período de vigencia + 3 años después de su anulación |

Control de Modificaciones

|  |  |
| --- | --- |
| Documento nº: | @Firma-Integra-WS-MAN |
| Revisión: | 010 |
| Fecha: | 29-01-2024 |

|  |  |
| --- | --- |
| Rev. | 001 |
| Fecha | 23/05/2016 |
| Descripción | Documentación inicial |

|  |  |
| --- | --- |
| Rev. | 002 |
| Fecha | 20/04/2017 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.0\_000. |

|  |  |
| --- | --- |
| Rev. | 003 |
| Fecha | 16/10/2017 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.1\_000. |

|  |  |
| --- | --- |
| Rev. | 004 |
| Fecha | 13/04/2020 |
| Descripción | Se actualizan los tipos de respuesta de los métodos de validación y actualización de firma de los servicios de Integr@ para incluir la nueva funcionalidad que permite calcular la fecha de expiración de las firmas. |
| Rev. | 005 |
| Fecha | 23/12/2020 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.2\_000. |
| Rev. | 006 |
| Fecha | 22/02/2022 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.2\_001. |

|  |  |
| --- | --- |
| Rev. | 007 |
| Fecha | 21/04/2022 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.2\_002. |

|  |  |
| --- | --- |
| Rev. | 008 |
| Fecha | 26/09/2022 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.3\_000. |

|  |  |
| --- | --- |
| Rev. | 009 |
| Fecha | 07/10/2022 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.3\_001. |

|  |  |
| --- | --- |
| Rev. | 010 |
| Fecha | 29/01/2024 |
| Descripción | Se adecua el documento para hacer referencia a la versión 2.2.4\_000. |

Control de Distribución

|  |  |
| --- | --- |
| Documento: | @Firma-Integra-WS-MAN |
| Revisión: | 010 |
| Fecha: | 29-01-2024 |

Propiedad del documento:

Este documento pertenece al Gobierno de España y posee un carácter de público para uso y distribución en ámbitos autorizados por este mismo, según se recoge en la declaración de privacidad.

Declaración de privacidad:

El contenido de este documento está sujeto al protocolo de libre distribución dentro del entorno definido para el contexto.

Copias Electrónicas:

La distribución de este documento ha sido controlada a través del sistema de información.

Copias en Papel:

La vigencia de las copias impresas en papel está condicionada a la coincidencia de su estado de revisión con el que aparece en el sistema electrónico de distribución de documentos.

El control de distribución de copias en papel para su uso en proyectos u otras aplicaciones es responsabilidad de los usuarios del sistema electrónico de información.

Fecha de impresión 29 de enero de 2024

**Índice**

[Control de Modificaciones 2](#_Toc157413810)

[Control de Distribución 4](#_Toc157413811)

[1 Objeto 6](#_Toc157413812)

[2 Alcance 7](#_Toc157413813)

[3 Siglas 8](#_Toc157413814)

[4 Descripción de Interfaces Web Services 9](#_Toc157413815)

[4.1 Consideraciones previas 11](#_Toc157413816)

[4.2 Catálogo de Servicios 12](#_Toc157413817)

[4.2.1 AfirmaServices 12](#_Toc157413818)

[4.2.2 IntegraServices 32](#_Toc157413819)

[4.2.3 TSAServices 46](#_Toc157413820)

[4.2.4 EvisorServices 51](#_Toc157413821)

[4.2.5 Objetos comunes 61](#_Toc157413822)

# Objeto

El objeto de este documento es describir las interfaces WebServices publicadas por Integra 2.2.4\_000 en su versión Web Service. Mediante las citadas interfaces WebServices podrán realizarse operaciones de firma servidor sobre @firma, solicitudes de sello de tiempo sobre TSA, operaciones de firma desde integra, solicitudes de generación y validación de reportes sobre eVisor, así como operaciones de cifrado.

# 

# Alcance

Este documento cubre los siguientes aspectos:

* Descripción de la interfaz de Servicios web de Integra.
* Descripción de los elementos manejados por los servicios.

# 

# Siglas

|  |  |
| --- | --- |
| TGS | Telvent Global Services |
| XML | eXtensible Markup Language |
| XSL | eXtensible Stylesheet Language |
| XSL-FO | eXtensible Stylesheet Language Formatting Objects |
| XSLT | XSL Transformations |
| W3C | World Wide Web Consortium |

# Descripción de Interfaces Web Services

Integra publíca sus servicios divididos en 5 interfaces “AfirmaServices”, “TSAServices”, “IntegraServices”, “EvisorServices” y “CypherServices”.

* **AfirmaServices**: En esta interfaz se publicarán los métodos relativos a operaciones de firma y certificados ofrecidos por @firma y que podrán ser accedidos mediante Integra. También provee métodos para la validación mediante OCSP.

La interfaz contiene los siguientes métodos:

* **serverValidateCertificateOcsp**: Método que permite la validación de certificados mediante OCSP.
* **serverSign**: Método que realiza una petición de firma servidor a @firma.
* **serverCoSign**: Método que realiza una petición de co-firma servidor a @firma.
* **serverCounterSign**: Método que realiza una petición de contra-firma servidor a @firma.
* **serverUpgradeSignature**: Método que realiza una petición de actualización de firma servidor a @firma.
* **serverAsynchronousRequest**: Método que realiza la consulta asíncrona de una petición pendiente de resolver.
* **serverBatchVerifyCertificate**: Método que realiza una petición de validación de certificados en lote sobre @firma.
* **serverBatchVerifySignature**: Método que realiza una petición de validación de firmas en lote sobre @firma.
* **serverGetArchiveRetrieval**: Método que realiza una petición de obtención de firma almacenada en @firma.
* **serverVerifySignature**: Método que realiza una petición de validación de firma sobre @firma.
* **serverVerifyCertificate**: Metodo que realiza una petición de validación de certificado sobre @firma.

Puede consultarse el WSDL descriptor del servicio en la siguiente URL:

|  |
| --- |
| http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices?wsdl |

* **IntegraServices**: En esta interfaz se publicarán los métodos relativos a operaciones de firma ofrecidos por Integra. Integra genera firmas, co-firmas y contra-firmas en los formatos CAdES\_BES, CAdES\_EPES, CAdES\_T (solicitando sello de tiempo a una TSA), CAdES\_B\_LEVEL, CAdES\_T\_LEVEL (solicitando sello de tiempo a una TSA), XAdES\_BES, XAdES\_EPES, XAdES\_T (solicitando sello de tiempo a una TSA), XAdES\_B\_LEVEL, XAdES\_T\_LEVEL (solicitando sello de tiempo a una TSA), PAdES\_BASIC, PAdES\_BES, PAdES\_EPES, PAdES\_LTV, PAdES\_B\_LEVEL y PAdES\_T\_LEVEL (solicitando sello de tiempo a una TSA). Para formatos mas avanzados, integra se comunica con @firma para actualizar las firmas generadas al formato superior.

La interfaz contiene los siguientes métodos:

* **generaSignature**: Método que genera una firma desde Integra.
* **generateCoSignature**: Método que genera una co-firma desde Integra.
* **generateCounterSignature**: Método que genera una contra-firma desde Integra.
* **generateSignaturePAdESRubric**: Método que genera una firma PAdES con rúbrica desde Integra.
* **generateMultiSignaturePAdESRubric**: Método que genera una multi-firma PAdES con rúbrica desde Integra.
* **upgradeSignature**: Método que realiza una actualización de firma desde.
* **verifySignature**: Método que valida una firma desde Integra.
* **getSignedData**: Método que obtiene la información de una firma.

Puede consultarse el WSDL descriptor del servicio en la siguiente URL:

|  |
| --- |
| http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices?wsdl |

* **TSAServices**: En esta interfaz se publicarán los métodos relativos a operaciones de obtención de sellos de tiempo desde una TSA.

La interfaz contiene los siguientes métodos:

* **generateTimestamp**: Método que realiza una petición de sello de tiempo a la TSA.
* **verifyTimestamp**: Método que realiza una petición de validación de sello de tiempo a la TSA.
* **renewTimestamp**: Método que realiza una petición de renovación de sello de tiempo a la TSA.

Puede consultarse el WSDL descriptor del servicio en la siguiente URL:

|  |
| --- |
| http://<SERVIDOR>:<PUERTO>/Integra-services/services/TSAServices?wsdl |

* **EvisorServices**: En esta interfaz se publicarán los métodos relativos a operaciones de obtención reportes de firma desde eVisor.

La interfaz contiene los siguientes métodos:

* **generateReport**: Método que realiza una petición generación de reporte de firma a eVisor.
* **validateReport**: Método que realiza una petición de validación de un reporte de firma a eVisor.

Puede consultarse el WSDL descriptor del servicio en la siguiente URL:

|  |
| --- |
| http://<SERVIDOR>:<PUERTO>/Integra-services/services/EvisorServices?wsdl |

## Consideraciones previas

El protocolo de acceso a dichos servicios se define mediante un mensaje de petición y otro de respuesta al mismo. Ambos mensajes se intercambian haciendo uso del protocolo SOAP 1.2.

Para la implementación de los servicios se ha utilizado la tecnología AXIS2

Los servicios web publicados reciben como parámetro de entrada un objeto que representa en cada caso la petición. Los objetos se especificarán mas adelante.

Así mismo, los servicios web publicados retornarán también un objeto que representa la respuesta concreta para cada uno de los métodos existentes.

## Catálogo de Servicios

En este apartado se describe las interfaces WS publicadas por Integra.

### AfirmaServices

La interfaz engloba los servicios de firma, co-firma, contra-firma, actualización, validación de firma y certificados enviando peticiones al servidor de @firma y OCSP.

#### Validación de certificados por OCSP

Este servicio permite la validación de certificados mediante OCSP enviando una petición de validación al servidor OCSP configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverValidateCertificateOcsp |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestValidateOCSP.

**RequestValidateOCSP:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| certificate | base64Binary | Certificado a validar. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseValidateOCSP.

**ResponseValidateOCSP:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| status | int | Estado asociado a la respuesta OCSP. Los valores que devuelven pueden ser:   * **0** 🡪 La respuesta de validación de certificado es correcta. * **1** 🡪 La petición de validación de certificado es incorrecta. * **2** 🡪 Se ha producido un error interno en el emisor. * **3** 🡪 Intentar la validación del certificado más tarde. * **5** 🡪 La petición de validación de certificado debe ir firmada. * **6** 🡪 La petición no está autorizada. |
| errorMsg | string | Mensaje de error en el caso de que la validación del certificado no haya sido correcta. |
| revocationDate | date | Fecha de revocación del certificado, en caso de que se encuentre revocado. |
| maxAge | date | Fecha en que la respuesta OCSP cacheada expira en el servidor OCSP al que se ha realizado la petición de validación de certificado. |

#### Firma servidor

Este servicio permite la generación de una firma servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverSign |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerSign.

**RequestServerSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| document | base64Binary | Datos a firmar. | SI (una de las opciones) |
| documentHash | DocumentHash | Hash del documento a firmar (representado por el objeto DocumentHash). |
| documentId | string | Identificador único del documento a firmar. |
| documentRepository | Repository | Localización del documento a firmar dentro de un repositorio o gestor de documentos (representado por el objeto Repository). |
| hashAlgorithm | HashAlgorithmEnum | Algoritmo de hash definido para calcular la firma. | NO |
| ignoreGracePeriod | booleano | Valor lógico que indica si se debe ignorar el periodo de gracia (verdadero) o no (falso). | NO |
| keySelector | string | Identificador de la clave usada para generar la firma delegada. | SI |
| signatureFormat | SignatureFormatEnum | Formato de la firma a generar. | SI |
| signaturePolicyIdentifier | string | Identificador de la política de firma asociada. | NO |
| xmlSignatureMode | XmlSignatureModeEnum | Modo de firma en formato XML. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerSign.

**ResponseServerSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signature | base64Binary | Firma generada, en el caso de que se haya realizado una petición de generación de firma. |
| signatureFormat | string | Formato de la firma generada o actualizada. |
| transactionId | string | Identificador único de la transacción generada. |
| updatedSignature | base64Binary | Firma actualizada, en el caso de que se haya realizado una petición de actualización de firma. |

#### Co-Firma servidor

Este servicio permite la generación de una co-firma servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverCoSign |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerCoSign.

**RequestServerCoSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| document | base64Binary | Datos a firmar. | NO |
| documentRepository | Repository | Localización del documento a firmar dentro de un repositorio o gestor de documentos (representado por el objeto Repository). | NO |
| signature | base64Binary | firma a utilizar en la generación de la co-firma. | SI (una de las opciones) |
| signatureRepository | Repository | Localización de la firma dentro de un repositorio o gestor de documentos (representado por el objeto Repository). |
| transactionId | string | Identificador de transacción asociado a la generación de la firma a co-firmar. |
| hashAlgorithm | HashAlgorithmEnum | Algoritmo de hash definido para calcular la firma. | NO |
| ignoreGracePeriod | booleano | Valor lógico que indica si se debe ignorar el periodo de gracia (verdadero) o no (falso). | NO |
| keySelector | string | Identificador de la clave usada para generar la firma delegada. | SI |
| signaturePolicyIdentifier | string | Identificador de la política de firma a utilizar en la generación de la co-firma. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerSign.

**ResponseServerSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signature | base64Binary | Firma generada, en el caso de que se haya realizado una petición de generación de firma. |
| signatureFormat | string | Formato de la firma generada o actualizada. |
| transactionId | string | Identificador único de la transacción generada. |
| updatedSignature | base64Binary | Firma actualizada, en el caso de que se haya realizado una petición de actualización de firma. |

#### Contra-Firma servidor

Este servicio permite la generación de una contra-firma servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverCounterSign |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerCounterSign.

**RequestServerCounterSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| transactionId | string | Identificador de transacción asociado a la generación de la firma que contra-firmar. | SI (una de las opciones) |
| signature | base64Binary | Firma a utilizar en la generación de la contra-firma. |
| signatureRepository | Repository | Localización de la firma dentro de un repositorio o gestor de documentos (representado por el objeto Repository). |
| hashAlgorithm | HashAlgorithmEnum | Algoritmo de hash definido para calcular la firma. | NO |
| ignoreGracePeriod | booleano | Valor lógico que indica si se debe ignorar el periodo de gracia (verdadero) o no (falso). | NO |
| keySelector | string | Identificador de la clave usada para generar la contra-firma delegada. | SI |
| targetSigner | base64Binary | El firmante objetivo de la contra-firma. | NO |
| signaturePolicyIdentifier | string | Identificador de la política de firma a utilizar en la generación de la contra-firma. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerSign.

**ResponseServerSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signature | base64Binary | Firma generada, en el caso de que se haya realizado una petición de generación de firma. |
| signatureFormat | string | Formato de la firma generada o actualizada. |
| transactionId | string | Identificador único de la transacción generada. |
| updatedSignature | base64Binary | Firma actualizada, en el caso de que se haya realizado una petición de actualización de firma. |

#### Upgrade Firma servidor

Este servicio permite la actualización de una firma en servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverUpgradeSignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerUpgradeSignature.

**RequestServerUpgradeSignature:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| transactionId | string | Identificador de transacción asociado a la generación de la firma que actualizar. | SI (una de las opciones) |
| signatureRepository | Repository | Localización de la firma dentro de un repositorio o gestor de documentos (representado por el objeto Repository). |
| signature | base64Binary | Firma a actualizar. |
| ignoreGracePeriod | booleano | Valor lógico que indica si se debe ignorar el periodo de gracia (verdadero) o no (falso). | NO |
| signatureFormat | SignatureFormatEnum | Formato al que actualizar la firma. | SI |
| targetSigner | base64Binary | El firmante objetivo de la actualización. | NO |
| processAsNotBaseline | booleano | Valor lógico que indica si se debe forzar el procesamiento de la petición como no baseline (verdadero) o no (falso). | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerSign.

**ResponseServerSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signature | base64Binary | Firma generada, en el caso de que se haya realizado una petición de generación de firma. |
| signatureFormat | string | Formato de la firma generada o actualizada. |
| transactionId | string | Identificador único de la transacción generada. |
| updatedSignature | base64Binary | Firma actualizada, en el caso de que se haya realizado una petición de actualización de firma. |

#### Validación de firmas en servidor

Este servicio permite la validación de una firma enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverVerifySignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerVerifySignature.

**RequestServerVerifySignature:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente. | SI |
| document | base64Binary | Documento original que fue firmado. | NO |
| documentHash | DocumentHash | Hash del documento original que fue firmado. | NO |
| documentRepository | Repository | Ubicación del documento original en un gestor de documentos o repositorio. | NO |
| optionalParameters | OptionalParameters | Conjunto de parámetros adicionales que incluir en la petición. | NO |
| signature | base64Binary | Firma a validar. | SI (una de las opciones) |
| signatureRepository | Repository | Ubicación de la firma que validar en un gestor de documentos o repositorio. |
| verificationReport | VerificationReport | Resultado e información adicional generada durante el proceso de verificación de la firma. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerVerifySignature.

**ResponseServerVerifySignature:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| signatureFormat | string | Formato de la firma validada. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signedDataInfo | DataInfo (lista) | Lista con los datos firmados de por cada uno de los firmantes contenidos en la firma (representado por el objeto DataInfo). |
| verificationReport | IndividualSignatureReport (lista) | Lista con la información detallada sobre el procesamiento de cada firma contenida en la firma original (representado por el objeto IndividualSignatureReport). |

#### Validación de certificado en servidor

Este servicio permite la validación de un certificado enviando una petición de validación al servidor @fima configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverVerifyCertificate |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerVerifyCertificate.

**RequestServerVerifyCertificate:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| certificate | base64Binary | Certificado a validar. | SI (una de las opciones) |
| certificateRepository | Repository | Ubicación del certificado a validar en un gestor de documentos o repositorio. |
| returnReadableCertificateInfo | boolean | Valor lógico que indica si se solicitará información acerca de campos del certificado en la petición (verdadero), o no (falso). | NO |
| returnVerificationReport | VerificationReport | Validaciones a realizar sobre el certificado, así como la información que debe ser devuelta en la respuesta. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerVerifyCertificate.

**ResponseServerVerifyCertificate:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| certificatePathValidity | CertificatePathValidity | Información asociada a la validación del certificado firmante. |
| readableCertificateInfo | map | Información detallada del certificado firmante. Esta información será un mapa de campos del tipo atributo/valor con el resultado de haber procesado el certificado según la configuración del sistema. |
| result | Result | Resultado del proceso. |

#### Validacion de certificados en lote

Este servicio permite la validación de certificados en lote en servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverBatchVerifyCertificate |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerBatchVerifyCertificate.

**RequestServerBatchVerifyCertificate:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| listVerifyCertificate | VerifyCertificateRequest (lista) | Lista de peticiones para validar certificados (elementos representados por el objeto VerifyCertificateRequest). | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerBatchVerifyCertificate.

**ResponseServerBatchVerifyCertificate:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. | |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. | |
| asyncResponse | string | | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | | Resultado del proceso (representado por el objeto Result). |
| listVerifyResponse | VerifyCertificateResponse (lista) | | Lista con las respuestas de la validación de cada uno de los certificados (representados por el objeto VerifyCertificateResponse). |

#### Validacion de firmas en lote

Este servicio permite la validación de firmas en lote en servidor enviando una petición al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverBatchVerifySignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerBatchVerifySignature.

**RequestServerBatchVerifySignature:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| listVerifySignature | VerifySignatureRequest (lista) | Lista de peticiones para validar firmas (elementos representados por el objeto VerifySignatureRequest). | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerBatchVerifySignature.

**ResponseServerBatchVerifySignature:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. | |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. | |
| asyncResponse | string | | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | | Resultado del proceso (representado por el objeto Result). |
| listVerifyResponse | VerifySignatureResponse (lista) | | Lista con las respuestas de la validación de cada uno de las firmas (representados por el objeto VerifySignatureResponse). |

#### Petición asíncrona servidor

Este servicio permite enviar una solicitud para recuperar información de forma asíncrona al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverAsynchronousRequest |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerPending.

**RequestServerPending:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| responseId | string | Identificador del proceso asíncrono. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerAsynchronous.

**ResponseServerAsynchronous:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| batVerCerRes | BatchVerifyCertificateResponse | Respuesta del servicio de validación de certificados en lote (representado por el objeto BatchVerifyCertificateResponse). |
| batVerSigRes | BatchVerifySignatureResponse | Respuesta del servicio de validación de firmas en lote (representado por el objeto BatchVerifySignatureResponse). |
| invAsyRes | InvalidAsyncResponse | Respuesta incorrecta del servicio (representado por el objeto InvalidAsyncResponse). |
| serSigRes | ServerSignerResponse | Respuesta del servicio de firma servidor (representado por el objeto ServerSignerResponse). |

**ServerSignerResponse:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signature | base64Binary | Firma generada, en el caso de que se haya realizado una petición de generación de firma. |
| signatureFormat | string | Formato de la firma generada o actualizada. |
| transactionId | string | Identificador único de la transacción generada. |
| updatedSignature | base64Binary | Firma actualizada, en el caso de que se haya realizado una petición de actualización de firma. |

**InvalidAsyncResponse:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| result | Result | Resultado del proceso (representado por el objeto Result). |

**BatchVerifyCertificateResponse:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| listVerifyResponse | VerifyCertificateResponse (lista) | Lista con las respuestas de la validación de cada uno de los certificados (representados por el objeto VerifyCertificateResponse). |

**BatchVerifySignatureResponse:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| asyncResponse | string | Identificador de proceso asíncrono que identifica la petición en el caso de que se haya realizado una petición de generación o actualización de firma con periodo de gracia. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| listVerifyResponse | VerifySignatureResponse (lista) | Lista con las respuestas de la validación de cada una de las firmas (representado por el objeto VerifySignatureResponse). |

#### Obtención firma almacenada

Este servicio permite la obtención de una firma almacenada en el servidor de @firma enviando una petición de obtención al servidor @firma configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/AfirmaServices |
| **Método** | serverGetArchiveRetrieval |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestServerArchive.

**RequestServerArchive:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de @firma. | SI |
| transactionId | string | Identificador de transacción de la firma a recuperar. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseServerArchive.

**ResponseServerArchive:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| result | Result | Resultado del proceso. |
| signature | base64Binary | Firma obtenida en la petición. |

### IntegraServices

La interfaz engloba los servicios de firma, co-firma, contra-firma, actualización, validación de firmas realizados por Integra.

#### Generación de firmas

Este servicio permite la generación de firmas desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | generateSignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestSign.

**RequestSign:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| alias | string | Alias del certificado firmante | SI |
| dataToSign | base64Binary | Datos a firmar. | SI |
| includeSignaturePolicy | boolean | Booleano que indica si la firma a generar debe incluir política de firma (verdadero) o no (falso). | NO |
| signatureFormat | SignatureFormatEnum | Formato de la firma a generar (Los formatos CMS, PDF y ODF no están soportados para esta operación). | SI |
| signature | base64Binary | Firma original que co-firmar o contra-firmar (solo para los métodos generateCoSignature y generateCounterSignature). | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |

#### Generación de co-firmas

Este servicio permite la generación de co-firmas desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | generateCoSignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestSign.

**RequestSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| alias | string | Alias del certificado firmante | SI |
| dataToSign | base64Binary | Datos a co-firmar. | SI (Solo para firmas CAdES explícitas) |
| includeSignaturePolicy | boolean | Booleano que indica si la co-firma a generar debe incluir política de firma (verdadero) o no (falso). | NO |
| signatureFormat | SignatureFormatEnum | Formato de la co-firma a generar (Los formatos CMS, PDF y ODF no están soportados para esta operación). | SI |
| signature | base64Binary | Firma original que co-firmar o contra-firmar (solo para los métodos generateCoSignature y generateCounterSignature). | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |

#### Generación de contra-firmas

Este servicio permite la generación de contra-firmas desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | generateCounterSignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestSign.

**RequestSign:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| alias | string | Alias del certificado firmante | SI |
| dataToSign | base64Binary | Datos a contra-firmar. | SI (Solo para firmas CAdES explícitas) |
| includeSignaturePolicy | boolean | Booleano que indica si la contra-firma a generar debe incluir política de firma (verdadero) o no (falso). | NO |
| signatureFormat | SignatureFormatEnum | Formato de la contra-firma a generar(Los formatos CMS, PDF y ODF no están soportados para esta operación). | SI |
| signature | base64Binary | Firma original que co-firmar o contra-firmar (solo para los métodos generateCoSignature y generateCounterSignature). | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |

#### Generación de firmas pades con rúbrica

Este servicio permite la generación de firmas pades con rúbrica desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | generateSignaturePAdESRubric |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestPAdESRubricSign.

**RequestPAdESRubricSign:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| alias | string | Alias del certificado firmante | SI |
| dataToSign | base64Binary | Datos a firmar. | SI |
| includeSignaturePolicy | boolean | Booleano que indica si la firma CAdES que se incluirá en el diccionario de firma contendrá política de firma o no. | NO |
| includeTimestamp | boolean | Booleano que indica si la firma CAdES que se incluirá en el diccionario de firma contendrá sello de tiempo o no. | NO |
| image | base64Binary | Imagen a insertar en el documento PDF como rúbrica. | SI |
| imagePage | string | Número de página en el que se quiere insertar la rúbrica. Si es “-1”, la rúbrica se insertará en la última página del documento. | SI |
| lowerLeftX | int | Coordenada horizontal inferior izquierda (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| lowerLeftY | int | Coordenada vertical inferior izquierda (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| upperRightX | int | Coordenada horizontal superior derecha (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| upperRightY | int | Coordenada vertical superior derecha (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |

#### Generación de milti-firmas pades con rúbrica

Este servicio permite la generación de multi-firmas pades con rúbrica desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | generateMultiSignaturePAdESRubric |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestPAdESRubricSign.

**RequestPAdESRubricSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| alias | string | Alias del certificado firmante | SI |
| dataToSign | base64Binary | Datos a firmar. | SI |
| includeSignaturePolicy | boolean | Booleano que indica si la firma CAdES que se incluirá en el diccionario de firma contendrá política de firma o no. | NO |
| includeTimestamp | boolean | Booleano que indica si la firma CAdES que se incluirá en el diccionario de firma contendrá sello de tiempo o no. | NO |
| image | base64Binary | Imagen a insertar en el documento PDF como rúbrica. | SI |
| imagePage | string | Número de página en el que se quiere insertar la rúbrica. Si es “-1”, la rúbrica se insertará en la última página del documento. | SI |
| lowerLeftX | int | Coordenada horizontal inferior izquierda (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| lowerLeftY | int | Coordenada vertical inferior izquierda (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| upperRightX | int | Coordenada horizontal superior derecha (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |
| upperRightY | int | Coordenada vertical superior derecha (en pixels) de la posición donde será insertada la imagen como rúbrica en la página correspondiente. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |

#### Actualización de firmas

Este servicio permite la actualización de firmas desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | upgradeSignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestUpgradeSign.

**RequestUpgradeSign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| listSigners | SignerToUpgrade (lista) | Lista de firmantes que actualizar. Si la lista es nula o vacía se actualizarán todos los firmantes. Este parámetro es sólo aplicable a firmas ASN.1 ,XML o ASiC-S Baseline. | NO |
| signature | base64Binary | Firma a actualizar. | SI |

SignerToUpgrade:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| signer | base64Binary | Certificado X509v3 firmante. |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseSign.

**ResponseUpgradeSign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| sign | base64Binary | Representa la firma generada. |
| expirationDate | Date | Fecha que representa la caducidad de la firma, esto es, la fecha a partir de la cual, la firma dejará de ser válida. |

#### Validación de firmas

Este servicio permite la validación de firmas desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | verifySignature |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestVerifySign.

**RequestVerifySign:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| signature | base64Binary | Firma a validar. | SI |
| signedData | base64Binary | Datos originales que han sido firmados por la firma a validar. Sólo será requerido en el caso de que la firma a validar sea ASN.1 y explícita. | SI (solo para firmas CAdES explícitas) |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseVerifySign.

**ResponseVerifySign:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| validationResult | ValidationResultWS | Resultado de la validación (representado por el objeto ValidationResultWS). |

**ValidationResultWS:**

| **Parámetro** | **Tipo** | **Descripción** |
| --- | --- | --- |
| correct | boolean | Indica si el resultado del proceso de validación de la firma ha sido correcto. Devuelve un valor lógico que indica si la firma es válida (verdadero) o no (falso). |
| errorDescription | string | Descripción del error producido en el proceso de validación de la firma, en caso de que ésta no sea válida. |
| signersList | ValidationInfoWS (lista) | Lista de elementos que contienen información asociada a la validación de los firmantes de la firma (elementos representados por el objeto ValidationInfoWS). |
| expirationDate | Date | Fecha que representa la caducidad de la firma, esto es, la fecha a partir de la cual, la firma dejará de ser válida. |

**ValidationInfoWS:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| certificate | base64Binary | Certificado X509v3 asociado al firmante. |
| correct | boolean | Indica si el resultado del proceso de validación del firmante ha sido correcto. Devuelve un valor lógico que indica si el firmante es válido (verdadero) o no (falso). |
| errorDescription | string | Descripción del error producido en el proceso de validación del firmante, en caso de que éste no sea válido. |
| signersList | ValidationInfoWS (lista) | Lista de objetos java que contienen información asociada a la validación de los contra-firmantes para este firmante. |

#### Obtención datos de firmas

Este servicio permite la obtención de datos de una firma desde el propio Integra.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/IntegraServices |
| **Método** | getSignedData |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestGetSignedData.

**RequestGetSignedData:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| signature | base64Binary | Firma de la que se quiere extraer la información. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseGetSignedData.

**ResponseGetSignedData:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| originalSignedData | OriginalSignedData | Objeto OriginalSignedData, que contendrá información acerca de los datos originalmente firmados, diferenciando dicha información en función del tipo de la firma procesada. Si la firma es de tipo PAdES (Baseline o no), el método devolverá la revisión que se corresponde con el documento PDF sin firmar. Si la firma es de tipo CAdES (Baseline o no), si la firma es implícita, devolverá los datos firmados y si es explícita, devolverá el resumen de los datos firmados y el algoritmo de resumen usado para calcularlo. Si la firma es ASiC-S, devolverá el fichero que se encuentra dentro del fichero ZIP y que se corresponde con los datos firmados. Si la firma es XAdES (Baseline o no) el método lanza directamente la excepción SigningException ya que no está soportado para las firmas XAdES. |

**OriginalSignedData:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| hashAlgorithm | string | Algoritmo de resumen utilizado para calcular el resumen de los datos firmados. |
| hashSignedData | base64Binary | Resumen de los datos firmados. |
| mimetype | string | Valor del mimetype del documento firmado. |
| signedData | base64Binary | Datos firmados. |

### TSAServices

La interfaz engloba los servicios relativos a sellos de tiempo obtenidos por petición a una TSA.

#### Generación de sellos de tiempo

Este servicio permite la generación de sellos de tiempo enviando una petición a una TSA configurada.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/TSAServices |
| **Método** | generateTimestamp |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestTimestamp.

**RequestTimestamp:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de TSA. | SI |
| dataToStamp | base64Binary | Datos del fichero al que vamos a generar el sello de tiempo. | SI (una de las opciones) |
| documentHash | DocumentHash | Hash del documento a generar el sello de tiempo. |
| documentType | DocumentTypeEnum | Tipo de documento al que vamos a generar el sello de tiempo. | SI |
| timestampPreviousTimestampToken | base64Binary | Datos del sello de tiempo al que vamos a hacer el resellado. Solo para el método renewTimestamp. | NO |
| timestampTimestampToken | base64Binary | Datos del sello de tiempo que vamos a validar. Solo para el método verifyTimestamp. | NO |
| timestampType | TimestampTypeEnum | Tipo de sello de tiempo que vamos a generar. | SI |
| transformData | TransformData | Valor del objeto TransformData para la canonicalización. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseTimestamp.

**ResponseTimestamp:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| result | Result | Resultado del proceso. |
| timestamp | base64Binary | Sello de tiempo generado, en el caso de que se haya realizado una petición de generación de sello de tiempo o de renovación. |

#### Validación de sellos de tiempo

Este servicio permite la validación de sellos de tiempo enviando una petición a una TSA configurada.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/TSAServices |
| **Método** | verifyTimestamp |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestTimestamp.

**RequestTimestamp:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. | SI |
| applicationId | string | Identificador de la aplicación cliente de TSA. | SI |
| dataToStamp | base64Binary | Datos del fichero al que vamos a generar el sello de tiempo. | SI (una de las opciones) |
| documentHash | DocumentHash | Hash del documento a generar el sello de tiempo. |
| documentType | DocumentTypeEnum | Tipo de documento al que vamos a generar el sello de tiempo. | SI |
| timestampPreviousTimestampToken | base64Binary | Datos del sello de tiempo al que vamos a hacer el resellado. Solo para el método renewTimestamp. | NO |
| timestampTimestampToken | base64Binary | Datos del sello de tiempo que vamos a validar. Solo para el método verifyTimestamp. | SI |
| timestampType | TimestampTypeEnum | Tipo de sello de tiempo que vamos a generar. | SI |
| transformData | TransformData | Valor del objeto TransformData para la canonicalización. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseTimestamp.

**ResponseTimestamp:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| result | Result | Resultado del proceso. |
| timestamp | base64Binary | Sello de tiempo generado, en el caso de que se haya realizado una petición de generación de sello de tiempo o de renovación. |

#### Renovación de sellos de tiempo

Este servicio permite la renovación de sellos de tiempo enviando una petición a una TSA configurada.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/TSAServices |
| **Método** | renewTimestamp |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestTimestamp.

**RequestTimestamp:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. | SI |
| applicationId | string | Identificador de la aplicación cliente de TSA. | SI |
| dataToStamp | base64Binary | Datos del fichero al que vamos a generar el sello de tiempo. | SI (una de las opciones) |
| documentHash | DocumentHash | Hash del documento a generar el sello de tiempo. |
| documentType | DocumentTypeEnum | Tipo de documento al que vamos a generar el sello de tiempo. | SI |
| timestampPreviousTimestampToken | base64Binary | Datos del sello de tiempo al que vamos a hacer el resellado. Solo para el método renewTimestamp. | SI |
| timestampTimestampToken | base64Binary | Datos del sello de tiempo que vamos a validar. Solo para el método verifyTimestamp. | NO |
| timestampType | TimestampTypeEnum | Tipo de sello de tiempo que vamos a generar. | SI |
| transformData | TransformData | Valor del objeto TransformData para la canonicalización. | NO |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseTimestamp.

**ResponseTimestamp:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| result | Result | Resultado del proceso. |
| timestamp | base64Binary | Sello de tiempo generado, en el caso de que se haya realizado una petición de generación de sello de tiempo o de renovación. |

### EvisorServices

La interfaz engloba los servicios relativos a la generación y validación de reportes de firma enviando peticiones al servidor eVisor.

#### Generación de reporte de firma

Este servicio permite la generación de reportes de firma enviando peticiones al servidor eVisor configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/EvisorServices |
| **Método** | generateReport |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestEvisorGenerateReport.

**RequestEvisorGenerateReport:**

| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| --- | --- | --- | --- |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de eVisor. | SI |
| barcodeList | BarcodeEvisorRequest (lista) | Lista de códigos de barra que se quieren incluir en el informe generado (elementos representados por el objeto BarcodeEvisorRequest). | NO |
| docRepositoryLocationObjectId | string | Identificador del documento a incluir en el reporte localizado en un repositorio externo. | NO |
| docRepositoryLocationRepositoryId | string | Identificador del repositorio donde se localiza el documento a incluir en el reporte generado. | NO |
| document | base64Binary | Documento a inluir en el reporte de firma que se generará. | NO |
| externalParameterList | ParameterEvisorRequest (lista) | Parámetros adicionales incluidos en la petición de generación del reporte (elementos representados por el objeto ParameterEvisorRequest). | NO |
| includeSignature | string | Indica si debe incluirse la firma en el reporte. Valores permitidos (true o false) | NO |
| signRepositoryObjectId | string | Identificador del objeto firma obtenido desde un repositorio externo. | SI (una de las opciones) |
| signRepositoryRepositoryId | string | Identificador del repositorio externo donde debe localizarse la firma. |
| validationResponse | base64Binary | Respuesta SOAP Firmada de @firma de validación de firma |
| signature | base64Binary | Firma a incluir en el reporte. |
| templateId | string | Identificador de la plantilla utilizada para generar el reporte. | SI |

**BarcodeEvisorRequest:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| barcodeMessage | string | Message para generar el código de barras. |
| barcodeType | BarcodeTypeEnum | Tipo del código de barras. |
| configurationParameterList | ParameterEvisorRequest (lista) | Lista con parámetros de configuración adicionales (elementos representados por el objeto ParameterEvisorRequest). |

**ParameterEvisorRequest:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| parameterId | string | Identificador el parámetro. |
| parameterValue | string | Valor del parámetro |

**BarcodeTypeEnum:**

Tipo enumerado con los posibles valores **CODABAR, CODE128, CODE39, DATAMATRIX, EAN128, PDF417, QRCODE**.

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseEvisorGenerateReport.

**ResponseEvisorGenerateReport:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| report | base64Binary | Reporte generado. |
| result | EvisorResult | Resultado del proceso (representado por el objeto EvisorResult). |

EvisorResult:

| **Parámetro** | **Tipo** | **Descripción** |
| --- | --- | --- |
| codeResult | string | Indica el código de resultado del proceso:   1. Proceso finalizado satisfactoriamente. 2. Los parámetros de entrada no son válidos. 3. La firma no es válida. 4. La firma de la respuesta SOAP no es válida. 5. El documento pasado no es válido. 6. El servidor @firma no está disponible. 7. La plantilla de generación no es válida. 8. El número de páginas del documento a extraer excede al permitido por el sistema.   -1. Error no categorizado. |
| cause | string | Causa del error si lo hubiera. |
| message | string | Mesnaje con el resultado del proceso. |

#### Validación de reporte de firma

Este servicio permite la validación de reportes de firma enviando peticiones al servidor eVisor configurado.

La petición a este servicio se puede realizar invocando al método:

|  |  |
| --- | --- |
| **URL del Servicio** | http://<SERVIDOR>:<PUERTO>/Integra-services/services/EvisorServices |
| **Método** | validateReport |

* Datos de entrada:

Como parámetro de entrada recibe un objeto de tipo RequestEvisorValidateReport.

**RequestEvisorValidateReport:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** | **Requerido** |
| idClient | string | Id del cliente con el que se invoca el WS. Asignado por el administrador. Las configuraciones concretas del WS van asociadas a este id. | SI |
| applicationId | string | Identificador de la aplicación cliente de eVisor. | SI |
| report | base64Binary | Reporte a validar. | SI |

* Datos de salida:

Como parámetro de salida retorna un objeto de tipo ResponseEvisorValidateReport.

**ResponseEvisorValidateReport:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| integraSuccess | boolean | Booleano que indica si la petición al WS a integra ha sido exitosa y no han ocurrido errores internos. Retorna **true** si la invocación ha sido correcta y **false** en caso contrario. |
| integraErrorMsg | string | Mensaje de error descriptivo indicando el motivo del error en la invocación del WS de integra. |
| result | EvisorResult | Resultado del proceso (representado por el objeto EvisorResult). |

**EvisorResult:**

| **Parámetro** | **Tipo** | **Descripción** |
| --- | --- | --- |
| codeResult | string | Indica el código de resultado del proceso:  100 La firma del informe es válida.  101 Los parámetros de entrada no son válidos.  102 La firma del informe no es válida.  103 El proceso de validación de la firma es válido con observaciones.  104 No se ha podido determinar el estado de la firma electrónica.  105 El servidor @firma no está disponible.  -101 Error no categorizado. |
| cause | string | Causa del error si lo hubiera. |
| message | string | Mesnaje con el resultado del proceso. |

#### Parámetros para los códigos de barras.

Se enumeran los distintos tipos de códigos de barras soportados.

##### Codabar

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | Codabar | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| HumanReadablePlacement | Localización del mensaje. | BOTTON (Valor por defecto) |
| TOP |
| NONE |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### Code 39

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | Code39 | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| HumanReadablePlacement | Localización del mensaje. | BOTTON (Valor por defecto) |
| TOP |
| NONE |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### Code 128

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | Code128 | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| HumanReadablePlacement | Localización del mensaje. | BOTTON (Valor por defecto) |
| TOP |
| NONE |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### EAN-128

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | EAN128 | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| HumanReadablePlacement | Localización del mensaje. | BOTTON (Valor por defecto) |
| TOP |
| NONE |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### PDF 417

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | PDF417 | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### DataMatrix

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | DataMatrix | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |

##### QRCode

|  |  |  |
| --- | --- | --- |
|  | | |
| **Identificador** | QRCode | |
| **Configuración** | | |
| Id. Parámetro | Descripción | Valores definidos |
| Rotation | Ángulo de rotación (en sentido horario) de la imagen del código de barras | 90 |
| 180 |
| 270 |
| QRCodeWidth | Anchura en pixels. | Valor por defecto: 200 pixels |
| QRCodeHeight | Altura en pixels. | Valor por defecto: 200 pixels |

### Objetos comunes

Se describen a continuación los objetos comunes mencionados en los apartados anteriores.

#### Objetos de entrada:

##### DocumentHash:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| digestMethod | HashAlgorithmEnum | Algoritmo de resumen. |
| digestValue | base64Binary | Valor del resumen. |
| transform | TransformData | Valor del objeto TransformData para la canonicalización. |

##### Repository:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| id | string | Identificador del repositorio o gestor documental. |
| object | string | UUID del objeto almacenado en el repositorio o gestor documental. |

##### HashAlgorithmEnum:

Tipo enumerado con los posibles valores **SHA1, SHA256, SHA384, SHA512**.

##### SignatureFormatEnum:

Tipo enumerado con los posibles valores **CMS (solo para AfirmaServices), CAdES, CAdES\_BES, CAdES\_EPES, CAdES\_T, CAdES\_X, CAdES\_X1, CAdES\_X2, CAdES\_XL, CAdES\_XL1, CAdES\_XL2, CAdES\_A, CAdES\_BASELINE, CAdES\_B\_LEVEL, CAdES\_T\_LEVEL, CAdES\_LT\_LEVEL, CAdES\_LTA\_LEVEL, XAdES, XAdES\_BES, XAdES\_EPES, XAdES\_T, XAdES\_C, XAdES\_X, XAdES\_X1, XAdES\_X2, XAdES\_XL, XAdES\_XL1, XAdES\_XL2, XAdES\_A, XAdES\_BASELINE, XAdES\_B\_LEVEL, XAdES\_T\_LEVEL, XAdES\_LT\_LEVEL, XAdES\_LTA\_LEVEL, ODF (solo para AfirmaServices), PDF (solo para AfirmaServices), PAdES, PAdES\_BES, PAdES\_EPES, PAdES\_LTV, PAdES\_BASELINE, PAdES\_B\_LEVEL, PAdES\_T\_LEVEL, PAdES\_LT\_LEVEL, PAdES\_LTA\_LEVEL**.

##### XmlSignatureModeEnum:

Tipo enumerado con los posibles valores **ENVELOPING, ENVELOPED, DETACHED**.

##### DetailLevelEnum:

Tipo enumerado con los posibles valores **NO\_DETAILS, NO\_PATH\_DETAILS, ALL\_DETAILS**.

##### TransformData:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| XPath | String (lista entre 0 e infinitas ocurrencias) | Lista de algoritmos de canonicalización que se aplican al fichero indicado. |
| algorithm | string | Algoritmo de codificación usado para los métodos de la canonicalización. |

##### VerifyCertificateRequest:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| applicationId | string | Identificador de la aplicación cliente de @firma. |
| certificate | base64Binary | Certificado a validar. |
| certificateRepository | Repository | Ubicación del certificado a validar en un gestor de documentos o repositorio. |
| returnReadableCertificateInfo | boolean | Valor lógico que indica si se solicitará información acerca de campos del certificado en la petición (verdadero), o no (falso). |
| returnVerificationReport | VerificationReport | Validaciones a realizar sobre el certificado, así como la información que debe ser devuelta en la respuesta. |

##### VerificationReport:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| checkCertificateStatus | boolean | Valor lógico que indica si se desea verificar el estado de revocación del certificado (verdadero), o no (falso). |
| includeCertificateValues | boolean | Valor lógico que indica si se desea que la respuesta incluya los certificados validados (verdadero), o no (falso). |
| includeRevocationValues | boolean | Valor lógico que indica si se desea que la respuesta incluya los elementos de consulta de estado de revocación CRL u OCSP utilizados en la validación de los certificados que forman la cadena validada (verdadero), o no (falso). |
| reportDetailLevel | DetailLevelEnum | Nivel de detalle que se desea obtener en la respuesta del servicio. |

##### VerifySignatureRequest:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| applicationId | string | Identificador de la aplicación cliente. |
| document | base64Binary | Documento original que fue firmado. |
| documentHash | DocumentHash | Hash del documento original que fue firmado. |
| documentRepository | Repository | Ubicación del documento original en un gestor de documentos o repositorio. |
| optionalParameters | OptionalParameters | Conjunto de parámetros adicionales que incluir en la petición. |
| signature | base64Binary | Firma a validar. |
| signatureRepository | Repository | Ubicación de la firma que validar en un gestor de documentos o repositorio. |
| verificationReport | VerificationReport | Resultado e información adicional generada durante el proceso de verificación de la firma. |

##### OptionalParameters:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| additionalReportOption | boolean | Valor lógico que indica si se debe incluir en la respuesta del servicio cierta información adicional sobre el proceso de validación, referente a atributos firmados y no firmados incluidos en la firma (verdadero), o no (falso). |
| returnProcessingDetails | boolean | Valor lógico que indica si se debe incluir en la respuesta del servicio el resultado de cada tarea de verificación que forma el proceso de validación de la firma (verdadero), o no (falso). |
| returnReadableCertificateInfo | boolean | Valor lógico que indica si se debe devolver información detallada de los certificados validados (verdadero) o no (falso). |
| returnSignPolicyDocument | boolean | Valor lógico que indica si se debe incluir en la respuesta del servicio el documento de la política de firma asociada (verdadero), o no (falso). |
| returnSignedDataInfo | boolean | Valor lógico que indica si se debe incluir en la respuesta del servicio información acerca de los datos firmados (verdadero), o no (falso). |
| returnNextUpdate | Boolean | Valor lógico que indica si se debe incluir en la respuesta del servicio la fecha de expiración de la firma (verdadero), o no (falso). |
| processAsNotBaseline | Boolean | Valor lógico que indica si la petición debe ser procesada como no baseline (verdadero), o no (falso). |

##### DocumentTypeEnum:

Tipo enumerado con los posibles valores **BASE64\_DATA, BASE64\_XML, INLINE\_XML, ESCAPED\_XML, DOCUMENT\_HASH, TRANSFORMED\_DATA, DOCUMENT\_HASH\_TRANSFORMED\_DATA.**

##### TimestampTypeEnum:

Tipo enumerado con los posibles valores **XML, RFC\_3161**.

#### Objetos de salida:

##### Result:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| resultMajor | string | URI del resultado global del proceso. |
| resultMessage | string | Mensaje descriptivo del resultado del proceso. |
| resultMinor | string | URI del resultado concreto. |

##### VerifyCertificateResponse:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| certificatePathValidity | CertificatePathValidity | Información asociada a la validación del certificado firmante. |
| readableCertificateInfo | map | Información detallada del certificado firmante. Esta información será un mapa de campos del tipo atributo/valor con el resultado de haber procesado el certificado según la configuración del sistema. |
| result | Result | Resultado del proceso. |

##### CertificatePathValidity:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| detail | CertificateValidity (lista) | Lista con la información de verificación de cada uno de los certificados que componen la cadena de certificación (elementos representados por el objeto CertificateValidity). |
| identifier | string | Identificador asociado al certificado (emisor y número de serie). |
| summary | Detail | Resultado global de verificar el certificado firmante (representado por el objeto detail). |

##### CertificateValidity:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| infoMap | map | Mapa con todos los aspectos asociados a la validación del certificado. |

##### VerifySignatureResponse:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| signatureFormat | string | Formato de la firma validada. |
| result | Result | Resultado del proceso (representado por el objeto Result). |
| signedDataInfo | DataInfo (lista) | Lista con los datos firmados de por cada uno de los firmantes contenidos en la firma (representado por el objeto DataInfo). |
| verificationReport | IndividualSignatureReport (lista) | Lista con la información detallada sobre el procesamiento de cada firma contenida en la firma original (representado por el objeto IndividualSignatureReport). |

##### DataInfo:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| contentData | base64Binary | Datos originalmente firmados. |
| documentHash | DocumentHash | Resumen de los datos originalmente firmados (representado por el objeto DocumentHash). |
| signedDataRefs | string (lista) | Lista con la información de las referencias firmadas por el firmante. |

##### DocumentHash:

| **Parámetro** | **Tipo** | **Descripción** |
| --- | --- | --- |
| digestMethod | HashAlgorithmEnum | Algoritmo de resumen. |
| digestValue | base64Binary | Valor del resumen. |
| transform | TransformData | Valor del objeto TransformData para la canonicalización. |

##### TransformData:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| XPath | String (lista entre 0 e infinitas ocurrencias) | Lista de algoritmos de canonicalización que se aplican al fichero indicado. |
| algorithm | string | Algoritmo de codificación usado para los métodos de la canonicalización. |

##### HashAlgorithmEnum:

Tipo enumerado con los posibles valores **SHA1, SHA256, SHA384, SHA512**.

##### IndividualSignatureReport:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| detailedReport | string | Información adicional solicitada en la petición. |
| processingDetails | ProcessingDetail | Resultado de los diferentes pasos involucrados en el proceso de verificación de la firma. |
| readableCertificateInfo | map | Mapa (clave, valor) con la información asociada al certificado firmante. |
| result | Result | Resultado del procesamiento de una firma en particular. |
| sigPolicyDocument | base64Binary | Documento que especifica la política de firma asociada a la firma. |
| signaturePolicyIdentifier | string | Identificador de la política de firma asociada a la firma. |

##### ProcessingDetail:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| listIndeterminateDetail | Detail (lista) | Lista con las tareas de validación que han arrojado un resultado indeterminado (Elementos representados por el objeto Detail). |
| listInvalidDetail | Detail (lista) | Lista con las tareas de validación que han arrojado un resultado no satisfactorio (Elementos representados por el objeto Detail). |
| listValidDetail | Detail (lista) | Lista con las tareas de validación que han arrojado un resultado satisfactorio (Elementos representados por el objeto Detail). |

##### Detail:

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Tipo** | **Descripción** |
| type | string | URI que identifica la tarea de validación ejecutada. |
| code | string | URI que identifica el código asociado al resultado. |
| message | string | Mensaje descriptivo asociado al resultado del proceso. |