Especificación Técnica del WebService de Autenticación y Autorización

Índice de contenido

Introducción

Propósito

El siguiente documento describe los aspectos técnicos del servicio de Autenticación y Autorización de WebServices (WSAA) perteneciente a la AFIP. Dicho servicio es necesario para que Entes Externos a la AFIP (EE) accedan a los WebServices de Negocio (WSN) ofrecidos por la AFIP.

Descripción General del Servicio

El WS de Autenticación y Autorización es un servicio B2B ("Business to Business") que permite que los computadores pertenecientes a la AFIP y Entes Externos a la AFIP intercambien información en forma directa sin intervención de operadores. En dicha tarea intervienen los siguientes componentes:

- Un cliente de WS desarrollado por un EE siguiendo las especificaciones de este documento.
- El WSAA, WS publicado por la AFIP que implementa la autenticación de los computadores del EE (CEE) y la autorización del mismo como consumidor de un determinado WebService de Negocio (WSN).

Al usar especificaciones y protocolos estándares (PKI, XML, CMS, WSDL y SOAP) el cliente puede ser desarrollado con cualquier lenguaje de programación moderno.

Para que un Ente Externo a la AFIP (EE) esté autorizado a usar un WSN de AFIP, deberá realizar un tramite administrativo previo, cuya descripción esta fuera del alcance de este documento. Una vez finalizado exitosamente dicho tramite, el que incluye el alta de los CEE, el EE quedara registrado en el servicio de autorización de AFIP como entidad autorizada para usar el WSN.

Para que un CEE pueda utilizar efectivamente un WSN, debera solicitar un "Ticket de Acceso" (TA) por medio del WS de Autenticación y Autorización (WSAA). Dicho requerimiento se realiza mediante el envió de un "Ticket de Requerimiento de Acceso" (TRA) del CEE al WSAA, mediante mensajeria SOAP.

El WSAA realiza la verificación del "TRA" y si el requerimiento es correcto, devuelve un mensaje que contiene el TA que habilita al CEE a utilizar el WSN solicitado. Una vez que CEE obtiene el TA, el mismo debe utilizarlo para acceder al WSN en cada requerimiento.

En la actualidad, los Web Services de la AFIP, no están incluidos en un UDDI (Universal Description Discovery Integration) de acceso externo, por lo tanto para acceder a los servicios que ofrece la AFIP, es necesario utilizar WSDL (Web Services Definition Language) según la URL definida por AFIP. A partir del WSDL el EE puede construir un Cliente, para poder consumir el de WSN correspondiente.

En términos generales, el presente documento detalla las operaciones a realizar para:

- Generar un "Ticket de Requerimiento de Acceso" (TRA)
- Invocar el "Web Service de Autenticación y Autorización" (WSAA)
- Interpretar el mensaje de respuesta del WSAA y obtener el "Ticket de Acceso" (TA)

Referencias

Para mejor entendimiento de la presente especificación, se recomienda estar familiarizado con los siguientes estándares:

- PKI, http://www.pki.org
- XML, http://www.w3.org/TR/XML/
- SOAP, http://www.w3.org/TR/soap/
- WSDL, http://www.w3.org/TR/wsdl/
- WS-I, http://www.ws-i.org/
- CMS, http://www.ietf.org/rfc/rfc3852.txt
- NTP, http://www.ntp.org

Invocación del WSAA

WSDL del WSAA

A continuación, se expone el WSDL perteneciente al WSAA. El mismo estará disponible en una URL perteneciente a la AFIP. El que se expone pertenece a un equipo de homologación.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://10.30.72.24:8080/axis/services/LoginCMS"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://10.30.72.24:8080/axis/services/LoginCMS"
xmlns:intf="http://10.30.72.24:8080/axis/services/LoginCMS"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <wsdl:message name="loginCmsResponse">
      <wsdl:part name="loginCmsReturn" type="xsd:string"/>
   </wsdl:message>
   <wsdl:message name="loginCmsRequest">
      <wsdl:part name="request" type="xsd:string"/>
   </wsdl:message>
   <wsdl:portType name="LoginCmsWs">
      <wsdl:operation name="loginCms" parameterOrder="request">
         <wsdl:input message="impl:loginCmsRequest" name="loginCmsRequest"/>
         <wsdl:output message="impl:loginCmsResponse" name="loginCmsResponse"/>
      </wsdl:operation>
   </wsdl:portType>
   <wsdl:binding name="LoginCMSSoapBinding" type="impl:LoginCmsWs">
      <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="loginCms">
         <wsdlsoap:operation soapAction=""/>
         <wsdl:input name="loginCmsRequest">
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://services.authws.sua.dvadac.desein.afip.gov" use="encoded"/>
         </wsdl:input>
         <wsdl:output name="loginCmsResponse">
            <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://10.30.72.24:8080/axis/services/LoginCMS" use="encoded"/>
         </wsdl:output>
      </wsdl:operation>
   </wsdl:binding>
   <wsdl:service name="LoginCmsWsService">
      <wsdl:port binding="impl:LoginCMSSoapBinding" name="LoginCMS">
         <wsdlsoap:address location="http://10.30.72.24:8080/axis/services/LoginCMS"/>
      </wsdl:port>
   </wsdl:service>
</wsdl:definitions>
```

Sincronización de Clocks:

El clock del computador que genera el TRA y recive el TA deberá estar sincronizado a través del protocolo NTP con el equipo “time.afip.gov.ar”.

Flujo Principal

A continuación se describen los pasos que se deberán seguir para solicitar un TA al WSAA. Cada uno de los puntos es explicado detalladamente en los apartados siguientes.

1. Generar el mensaje del TRA (LoginTicketRequest.xml)
2. Generar el TRA con el mensaje anterior y su firma electrónica. (LoginTicketRequest.xml.cms)
3. Codificar en Base64 el TRA (LoginTicketRequest.xml.cms.bse64)
4. Invocar WSAA con el TRA y recibir LoginTicketResponse.xml
5. Extraer y validar la información de autorización (TA).

Generación del documento del TRA (LoginTicketRequest.xml)

El primer paso para solicitar un TA es preparar el documento del TRA (denominado LoginTicketRequest.xml). Se puede utilizar una estructura XML ya definida que puede ser obtenida de un archivo externo o declarada como constante en el propio código. El esquema (schema, XSD) que describe dicho XML es el siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:annotation>
    <xsd:documentation xml:lang="es">
     Esquema de Ticket de pedido de acceso a un Web Service
     por parte de un computador de un Organismo Externo
     Version repositorio SVN: $Rev: 477 $
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="loginTicketRequest" type="loginTicketRequestType" />
  <xsd:complexType name="loginTicketRequestType">
    <xsd:sequence>
          <xsd:element name="header" type="headerType" minOccurs="1" maxOccurs="1"/>
          <xsd:element name="service" type="serviceType" minOccurs="1" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="xsd:decimal" use="optional" default="1.0" />
  </xsd:complexType>
  <xsd:complexType name="headerType">
    <xsd:sequence>
          <xsd:element name="source" type="xsd:string" minOccurs="1" maxOccurs="1"/>
          <xsd:element name="destination" type="xsd:string" minOccurs="1" maxOccurs="1"/>
          <xsd:element name="uniqueId" type="xsd:unsignedInt" minOccurs="1" maxOccurs="1"/>
          <xsd:element name="generationTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1"/>
          <xsd:element name="expirationTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType name="serviceType">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[a-z][a-z,-,_,0-9]*"/>
      <xsd:minLength value='3'/>
      <xsd:maxLength value='32'/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

A continuación se detalla la descripción de los atributos. Los mismos deben respetar el formato definido en el XSD:

- ***source***: DN del computador que realiza el requerimiento (CEE). El mismo se debera corresponder con el certificado a emplear en la validación de la firma electrónica del TRA.
- ***destination***: DN del WSAA, el mismo deberá ser "cn=wsaa,o=afip,c=ar,serialNumber=CUIT 33693450239"
- ***uniqueId***: Entero de 32 bits que identifica el requerimiento.
- ***generationTime***: Momento en que fue generado el requerimiento.
- ***expirationTime***: Momento en el que expira la solicitud.
- ***service***: Identificación el WSN para el cual se solicita el TA.

El siguiente es un ejemplo del documento LoginTicketRequest.xml generado por la EE *Empresa SA* cuya CUIT es *30123456789* y el DN del CEE es *cn=srv1,ou=facturacion,o=empresa s.a.,c=ar,serialNumber=CUIT 30123456789* solicitando acceso al WSN *wsfe*:

```
<?xml version="1.0" encoding="UTF-8"?>
<loginTicketRequest version="1.0">
  <header>
    <source>cn=srv1,ou=facturacion,o=empresa s.a.,c=ar,serialNumber=CUIT 30123456789</source>
    <destination>cn=wsaa,o=afip,c=ar,serialNumber=CUIT 33693450239</destination>
    <uniqueId>4325399</uniqueId>
    <generationTime>2001-12-31T12:00:00-03:00</generationTime>
    <expirationTime>2001-12-31T12:10:00-03:00</expirationTime>
  </header>
  <service>wsfe</service>
</loginTicketRequest>
```

Generación del Ticket de Requerimiento de Acceso (TRA)

Se debará empaquetar en un mensaje CMS, el mensaje anteriormente generado (LoginTicketRequest.xml) junto con su firma electrónica utilizando SHA1+RSA. De esta forma, se obtiene el TRA (LoginTicketRequest.xml.cms).

Codificación en Base64 el TRA

Para poder enviar el TRA al WSAA, el mismo debará ser codificado en Base64 (LoginTicketRequest.xml.cms.bse64)

Envió del TRA al WSAA

Se debe invocar el método LoginCMS del WSAA. El mismo recibe como parámetro una cadena correspondiente a la codificación en Base64 del TRA (LoginTicketRequest.xml.cms.base64) y devuelve una cadena denominada LoginTicketResponse.xml. De esta ultima se deberá extraer el Ticket de Acceso (TA).

En caso de encontrarse algún error, el mensaje SOAP devolverá un “SoapFault” conteniendo código y descripción del error producido. La descripción podrá contener adicionalmente detalles mas específicos del error (ej: el XML expiro hace 10 minutos). La siguiente tabla lista los códigos de errores y su correspondiente descripción. En caso de que la AFIP considere necesario, nuevos códigos de errores y su descripción serán agregados.

Código	Descripción
coe.notAuthorized	CEE no autorizado a acceder al servicio solicitado
cms.bad	No se ha podido interpretar el CMS.
cms.sign.notFound	No se ha encontrado una firma en el CMS
cms.sign.invalid	No se ha podido verificar correctamente la firma en el CMS
cms.cert.expired	El certificado del CEE ha expirado
cms.cert.invalid	El certificado del CEE no es valido (ej. fecha de generación posterior al presente o fue emitido con otros propósitos)
cms.cert.untrusted	El certificado del CEE no se pudo verificar con el certificado de la AC emisora del certificado
xml.bad	El documento no es un xml o no se puede verificar contra el xsd (schema)
xml.source.invalid	El atributo 'source' no se corresponde con el DN del CEE
xml.destination.invalid	El atributo 'destination' no se corresponde con el DN del WSAA
xml.version.notSupported	La version del documento no es soportada
xml.generationTime.invalid	El tiempo de generación es invalido (ej: posterior a la hora actual)
xml.generationTime.old	El tiempo transcurrido desde el momento de generación es mayor al esperado
xml.expirationTime.expired	El documento ha expirado
wsn.unavailable	El servicio al que se desea acceder se encuentra momentáneamente fuera de servicio
wsn.notFound	El servicio requerido por el CEE no es valido
wsaa.unavailable	El servicio de autenticación/autorización se encuentra momentáneamente fuera de servicio
wsaa.internalError	El WSAA no ha podido procesar correctamente el requerimiento

Extracción y validación del TA

LoginTicketResponse.xml es descripto en el siguiente esquema (schema, XSD):

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <xsd:annotation>
                <xsd:documentation xml:lang="es">
                        Esquema de Ticket de respuesta al pedido de acceso a un Web
                        Service por parte de un computador de un Organismo Externo
                        Version repositorio SVN: $Rev: 477 $
                </xsd:documentation>
        </xsd:annotation>
        <xsd:element name="loginTicketResponse" type="loginTicketResponseType" />
        <xsd:complexType name="loginTicketResponseType">
             <xsd:sequence>
               <xsd:element name="header" type="headerType" minOccurs="1" maxOccurs="1" />
               <xsd:element name="credentials" type="credentialsType" minOccurs="1" maxOccurs="1" />
             </xsd:sequence>
             <xsd:attribute name="version" type="xsd:decimal" use="optional" default="1.0" />
        </xsd:complexType>
        <xsd:complexType name="headerType">
             <xsd:sequence>
               <xsd:element name="source" type="xsd:string" minOccurs="1" maxOccurs="1" />
               <xsd:element name="destination" type="xsd:string" minOccurs="1" maxOccurs="1" />
               <xsd:element name="uniqueId" type="xsd:unsignedInt" minOccurs="1" maxOccurs="1" />
               <xsd:element name="generationTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1" />
               <xsd:element name="expirationTime" type="xsd:dateTime" minOccurs="1" maxOccurs="1" />
             </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="credentialsType">
             <xsd:sequence>
               <xsd:element name="token" type="xsd:string" minOccurs="1" maxOccurs="1" />
               <xsd:element name="sign" type="xsd:string" minOccurs="1" maxOccurs="1" />
             </xsd:sequence>
        </xsd:complexType>
</xsd:schema>
```

Los datos que incluidos son los siguientes:

- ***source***: DN del WSAA. Corresponde al DN ***destination*** presente en LoginTicketRequest.xml
- ***destination***: DN del CEE. Corresponde al DN ***source*** informado en LoginTicketRequest.xml
- ***uniqueId***: Entero de 32 bits sin signo que identifica al requerimiento.
- ***generationTime***: Momento en que fue generado el TA.
- ***expirationTime***: Momento en el que expira el TA.
- ***token*** y ***sign***: cadenas de caracteres que deben ser informadas al WSN (como variables TOKEN y SIGN). Las mismas componen el TA. El formato interno de estas cadenas puede diferir de un servicio a otro y su información contenida es interpretada por el WSN.

Se deberá verificar que el mensaje de respuesta, que incluye al TA, no se encuentre expirado mediante la variable “expirationTime” y su momento de generación sea valido mediante la variable “generationTime”.

Una ejemplo de respuesta al requerimiento expuesto anteriormente es el siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<loginTicketResponse version="1.0">
  <header>
    <source>cn=wsaa,o=afip,c=ar,serialNumber=CUIT 33693450239</source>
    <destination>cn=srv1,ou=facturacion,o=empresa s.a.,c=ar,serialNumber=CUIT
30123456789</destination>
    <uniqueId>383953094</uniqueId>
    <generationTime>2001-12-31T12:00:02-03:00</generationTime>
    <expirationTime>2002-01-01T00:00:02-03:00</expirationTime>
  </header>
  <credentials>
    <token>cES0SSuWIIPlfe5/dLtb0Qeg2jQuvYuuSEDOrz+w2EnAQiEeS86gzYf7ehiU3UaYit5FRb9z/3zq</token>
    <sign>a6QSSZBgLf0TTcktSNteeSg3qXsMVjo/F5py/Gtw7xucTrUWbsrVCdIoGE8Cm1bixpuVPlr58k6n</sign>
  </credentials>
</loginTicketResponse>
```

Notar que el tiempo de vida del TA presente en este ejemplo es de 12 horas. El CEE podría utilizar este TA sin necesidad de solicitar otro, indistintamente de la cantidad de veces que consuma el servicio al cual solicito acceso. Para el acceso a un WSN para el cual un CEE posea un TA valido, se recomienda utilizar dicho TA y no solicitar uno nuevo.

Requerimientos de los certificados pertenecientes a los COE

La autenticación de los CEE, se realizara mediante certificados “x.509v3”. Los mismos deberan cumplir con los siguientes requerimientos:

1. Ser emitido por una autoridad certificante reconocida por AFIP.
2. El DN deberá enmarcarse dentro de la RFC 2256 (http://www.ietf.org/rfc/rfc2256.txt)
3. El contenido del DN se debe cumplir los siguientes requisitos establecidos por la “Oficina Nacional de Tecnologías de Información” (ONTI), disponible en http://www.sgp.gov.ar/contenidos/onti/productos/docs/infraestructura/Anexo_III_Perfil_Minimo_de_Certificados_y_CRLs_v1.pdf
    - Campo 'commonName': en caso de existir DEBE corresponder al nombre del servicio o aplicación (ej. Sistema de Consulta) o al nombre de la unidad operativa responsable del servicio (ej. Gerencia de Compras).
    - Campo 'serialNumber' (OID 2.5.4.5: Nro de serie): DEBE estar presente y DEBE contener el número de identificación de la Persona Jurídica Pública o Privada, expresado como texto y respetando el siguiente formato y codificación: "CUIT numero_de_cuit"
    - Campo 'organizationalUnitName': en caso de existir contendrá las unidades operativas relacionadas con el suscriptor del certificado, pudiendo utilizarse varias ocurrencias de este atributo de ser necesario.
    - Campo 'organizationName': DEBE coincidir con el nombre de la Persona Jurídica Pública o Privada.
    - Campo 'countryName': DEBE estar presente y DEBE representar el país en el cual está constituida la Persona Jurídica, codificado según el estándar [ISO3166].