Manual de instalación, configuración e integración

DEMO SP para PHP

Resumen: Este es un manual para la instalación, configuración e integración de una DEMO para proveedores de servicio.

Historial del Documento

Version	*Date*	*Modification reason*	*Modified by*
1.0	1/02/2012	Manual de Instalación	Joaquín Alcalde-Moraño Jensen
2.0	01/06/2014	Modificación del Software	José I. Cerón Bergantiños
2.1	15/12/2015	Nuevas configuraciones	Alfredo de Cea

Índice

Lista de figuras

Lista de abreviaturas

<Abreviatura>	<Explicación>
STORK	Secure idenTity acrOss boRders linKed
PEPS	Pan European Proxy Server
SP	Service Provider (Proveedor de Servicio)
IDP	Identity Provider (Proveedor de Identidad)

Resumen ejecutivo

Este documento ofrece información detallada sobre como configurar, crear y desplegar en PHP una aplicación para un Proveedor de Servicios (SP) para su uso en un intermediador de IDPs.

Como es necesaria la existencia de un Apache para desplegar la aplicación SP, el documento comienza dando una información básica sobre el servidor.

Después de eso, se describe qué necesita saber el usuario sobre las posibles configuraciones para su proyecto.

Tras leer este manual, el administrador / integrador debería ser capaz de configurar, crear y desplegar una aplicación que sea capaz de conectarse al Intermediador de IDPs.

1 Introducción

Este documento está dividido en varios capítulos con el fin de permitir al lector acceder fácilmente a las secciones más relevantes para el escenario específico en el que esté trabajando.

En el próximo capítulo se enseña cómo configurar un servidor Apache para usar el Framework simplesamlphp y donde se desplegará el paquete distribuido.

En el tercer capítulo se da una guía rápida de inicio de la DemoSP-PHP.

En el cuarto capítulo se describe cada configuración necesaria para la DemoSP-PHP.

En el quinto se muestra una típica sesión de ejecución.

En el sexto capítulo se demuestra cómo instalar DemoSP-PHP desde cero.

En el capítulo final se enseña cómo usar la SAML PHP API para generar y validar mensajes SAML.

2 Antes de empezar

Asegúrese de disponer de un servidor Apache operativo.

2.1 Keystore

La aplicación SP usa tres ficheros PEM (clave privada y dos claves públicas) para configurar el certificado para firmar las peticiones SAML, el certificado para incluir en la petición SAML y el certificado del Intermediador de IDPs en el que se confía.

2.1.1 Instalando los Certificados

La clave pública y privada del certificado del DemoSP-PHP deberán ser colocadas en el directorio cert, junto con la clave pública del Intermediador al que mandaremos la petición y en el cual tengamos que confiar. Esta acción se realizará más adelante, ya que para llevarla a cabo es necesario añadir previamente el proyecto al servidor (ver sección 3: Inicio Rápido sobre configurar la aplicación).

2.2 Configuración de Apache

1. Edite el fichero apache que contiene la información referente a virtual hosts. En la mayoría de distribuciones de Apache, este fichero se encuentra en la ruta */conf/httpd.conf*. En algunas distribuciones de Apache más modernas, el propio fichero *httpd.conf* importa otro fichero llamado *httpd-vhosts.conf* que se aloja en la ruta */conf/extra*.
2. Añada el siguiente host virtual a la configuración:

```
Alias /SP /var/simplesamlphp/www/SP/
```

3. Guarde y salga.
4. Reinicie su servidor apache.

Puede cambiar el alias de `/SP` si lo desea. Formará parte de la URL cuando accede al SP PHP. En esta guía consideraremos que el alias es "`/SP`".

3 Inicio rápido

El siguiente procedimiento le ayudará a poner en marcha en unos pocos minutos la aplicación usando una distribución de simplesamlphp. Para más detalles mire en los capítulos 4 y **¡Error! No se encuentra el origen de la referencia.**.

1. Copiar el contenido de la distribución DemoSP-PHP a `/var/simplesamlphp/`
2. Abrir el fichero `simplesamlphp/lib/SAML2/Constants.php`
    a. Editar la propiedad `ASSERTION_URL` con el valor de la URL del servidor:
    https://**inserta.tu.ip.aqui** /SP/return.php
    Por ejemplo: https://localhost/SP/return.php
    b. Editar el primer *require_once* que aparece en el fichero y establecer la ruta acorde con la ubicación del servidor.
    c. Editar la propiedad `LOGOUT_RETURN_URL` con el valor de la URL del servidor:
    https://**inserta.tu.ip.aqui** /SP/returnLogoutRequest.php
    Por ejemplo: https://localhost/SP/returnLogoutRequest.php
    d. Editar la propiedad `LOGOUT_SEND_URL` con el valor de la URL del servidor:
    https:// **inserta.aqui.tu.url.de.acceso** /Proxy/LogoutAction
    Por ejemplo: https://se-pasarela.clave.gob.es/Proxy/LogoutAction
3. Abrir el fichero `simplesamlphp/metadata/saml20-idp-remote.php`
    a. Editar la propiedad `SingleSignOnService` con el valor de la URL del Intermediador de IDPs:
    https://**inserta.aqui.tu.url.de.acceso**
    Por ejemplo:
    https://se-pasarela.clave.gob.es/Proxy/ServiceProvider

Ahora puede abrir su navegador y utilizar la aplicación (vea el capítulo 5).

4 Configurar DemoSP-PHP

4.1 SP

El proyecto DemoSP-PHP ofrece ficheros de configuración que pueden modificarse. En esta sección se explica cada propiedad.

4.1.1 config/authsources.php

El fichero `config/authsources.php` provee las principales configuraciones para el SP.

Creando múltiples entradas 'identifier' se hace posible crear más de un Proveedor de Servicios (SP) por máquina.

Key	Description
`'identifier'`	Identificador utilizado para identificar el SP. Ej: DEMO-SP
`Name`	Nombre del SP
`Certificate`	Nombre del fichero del Certificado del SP
`validate.certificate`	Nombre del fichero del Certificado del Intermediador de IDPs
`Privatekey`	Nombre del fichero con la clave privada del SP (en formato PEM)
`privatekey_pass`	Password de la clave privada del certificado del SP
`attributes.NameFormat`	Nombre del formato usado en los atributos
`sign.authnrequest`	Firma del AuthnRequest
`sign.logout`	Firma del logoutRequest

Los certificados y claves deben ser situadas en el directorio `cert/`.

Para más información, consulte: http://simplesamlphp.org/docs/1.8/saml:sp

4.1.2 lib/SAML2/Constants.php

Hay dos propiedades localizadas en este fichero que tienen que ver con el SP.

Key	Description
`ASSERTION_URL`	`La URL del SP que gestionará las respuestas del Intermediador de IDPs.`
`SPID`	`El ID del SP en uso, necesario para recibir credenciales alemanas.`
`SP_VC_FILE`	`El path al fichero de control de versiones generado por la herramienta generadora del control de versiones.`

4.2 Intermediador (IDP)

Existe un Intermediador entre el SP y los IDPs.

4.2.1 metadata/saml20-idp-remote.php

Contiene información acerca del Intermediador de IdPs de destino.

Key	Description
`$metadata['identifier']`	Identificador usaro para identificar al Intermediador
`Name`	Nombre del Intermediador.
`SingleSignOnService`	URL del Intermediador que gestionará la petición
`certFingerprint`	Fingerprint del certificado del IdP
`sign.authnrequest`	Firma del AuthnRequest
`Redirect.validate`	Firma válida en redirecciones

Para más información consulte: http://simplesamlphp.org/docs/1.8/simplesamlphp-reference-idp-remote.

4.3 SAML

El fichero `lib/SAML2/Constants.php tiene varias configuraciones sobre el mensaje SAML`.

4.3.1 Atributos

Los arrays `$idsPersonal`, `$idsBusiness`, `&idsLegal` y `$attrs` contienen los distintos atributos SAML soportados. Si desea añadir un nuevo atributo, tan solo debe añadir una nueva entrada en alguno de los tres array `$ids` y añadir la consiguiente entrada en `$attrs`:

Key	Description
`'attr'.name`	Nombre del nuevo atributo
`'attr'.uri`	URI del nuevo atributo
`'attr'.nameFormat`	Formato del nombre del nuevo atributo
`'attr'.value`	Valor por defecto del nuevo atributo

4.3.2 Países de los ciudadanos

Si desea añadir un nuevo país de ciudadanos debe modificar la propiedad `$countries`. Simplemente modifique la entrada entrada a esa propiedad.

4.3.3 Espacio de nombres (namespaces)

Para ajustar los namespaces y adaptarlos a sus propios requerimientos modifique las siguientes propiedades:

Key	Description
`STORKP_NS`	namespace del protocolo STORK
`SAMLP_NS`	namespace del protocolo SAML
`STORK_NS`	namespace de la assertion STORK

5 Iniciando DemoSP-PHP

Existe un demoSP funcional desplegado en la siguiente URL para realizar pruebas: https://pre-pasarela.clave.gob.es/SPProxy

Para probar el suyo, abra su navegador y entre en la siguiente página: "http(s):// **inserta.tu.ip.aqui** /SP/". Ahora debe de estar navegando en la DemoSP- PHP. Para modificar su propio SP refiérase a la siguiente sección.

En el SP debería ver una página similar a la Figura 1.

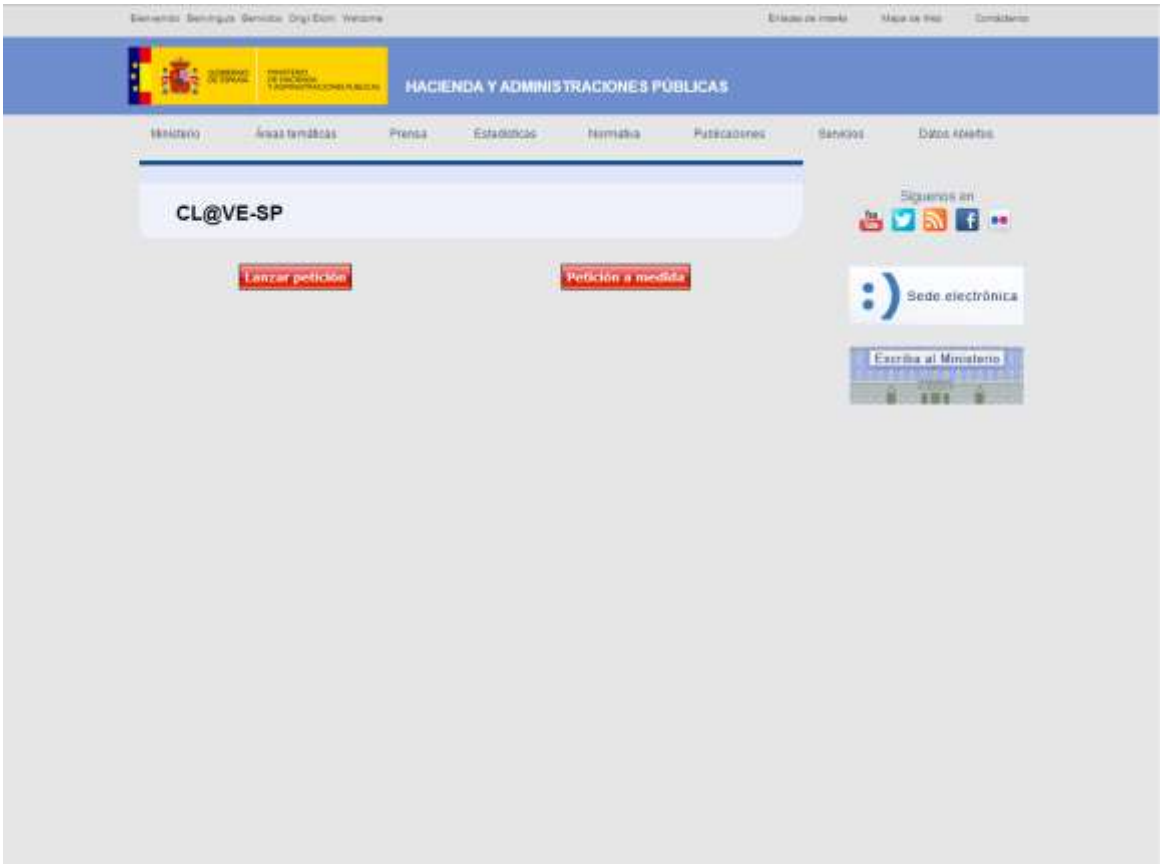

Figura 1 – Página de Inicio - DemoSP:php

Aquí tiene dos grandes opciones para proseguir.

El botón "Lanzar petición" servirá para hacer una prueba rápida y simple contra el Intermediador de IdPs que lo único que pedirá será el DNI, el nombre y los apellidos con un nivel QAA 3.

Si desea una petición más específica, pulse el otro botón (el de "Petición a medida") y debería ver una página similar a la Figura 2.

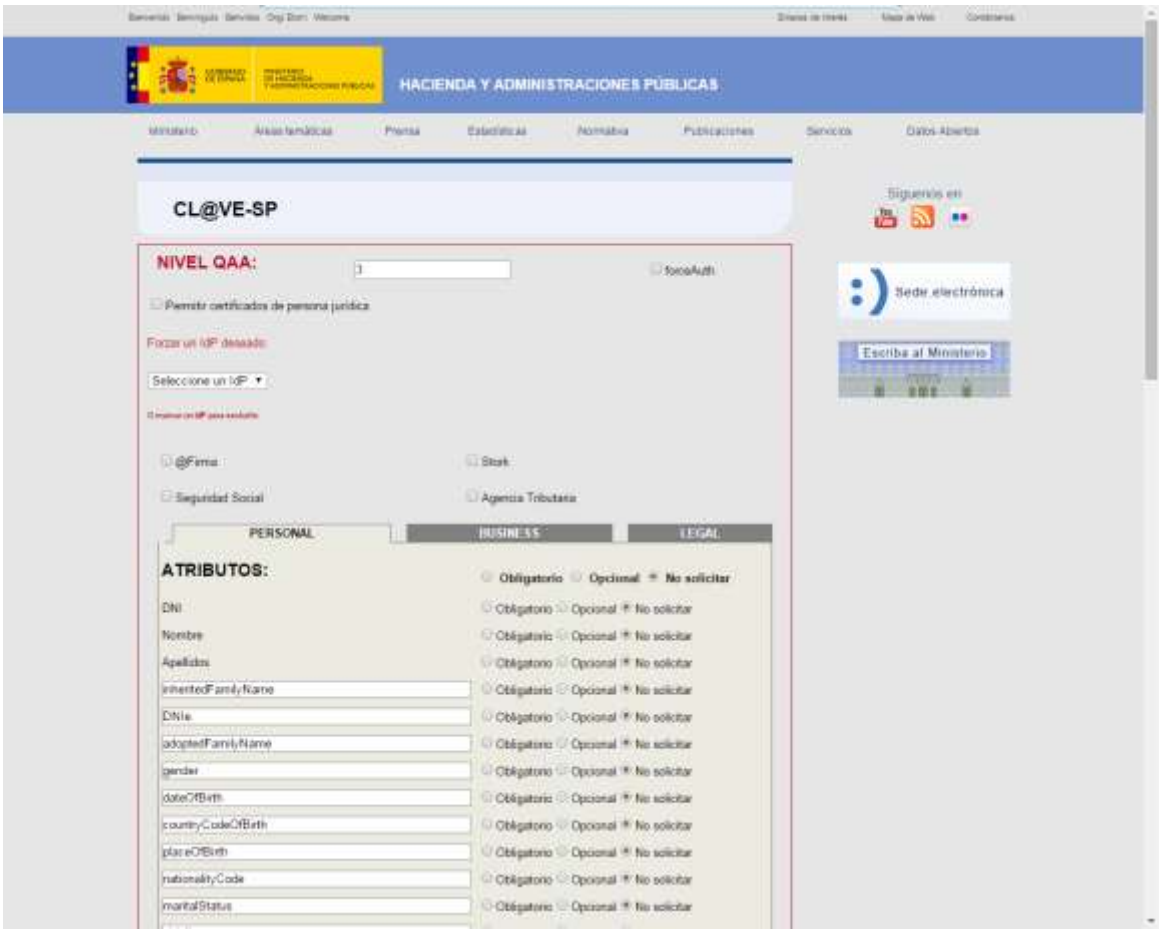

Figura 2 – Página de petición a medida - DemoSP:php

Aparecerá una lista con todos los IdP disponibles en el Intermediador. Si se selecciona uno de ellos, se eliminará la posibilidad de autenticarse con ese IdP cuando el usuario sea redirigido al Intermediador.

Existe una vía más rápida si lo único que se quiere es permitir un único IdP, en cuyo caso deberá elegirse en el menú desplegable que se encuentra encima de las opciones de IdPs.

Seleccione los atributos que desea solicitar en el proceso de autenticación y pulse "Lanzar petición".

Cuando el ciudadano sea redirigido al Intermediador de IdPs, tendrá que elegir entre una de las opciones para autenticarse.

Si la autenticación termina en éxito, deberá ver una tabla con los atributos pedidos junto con el valor obtenido, como se ve en la Figura 3.

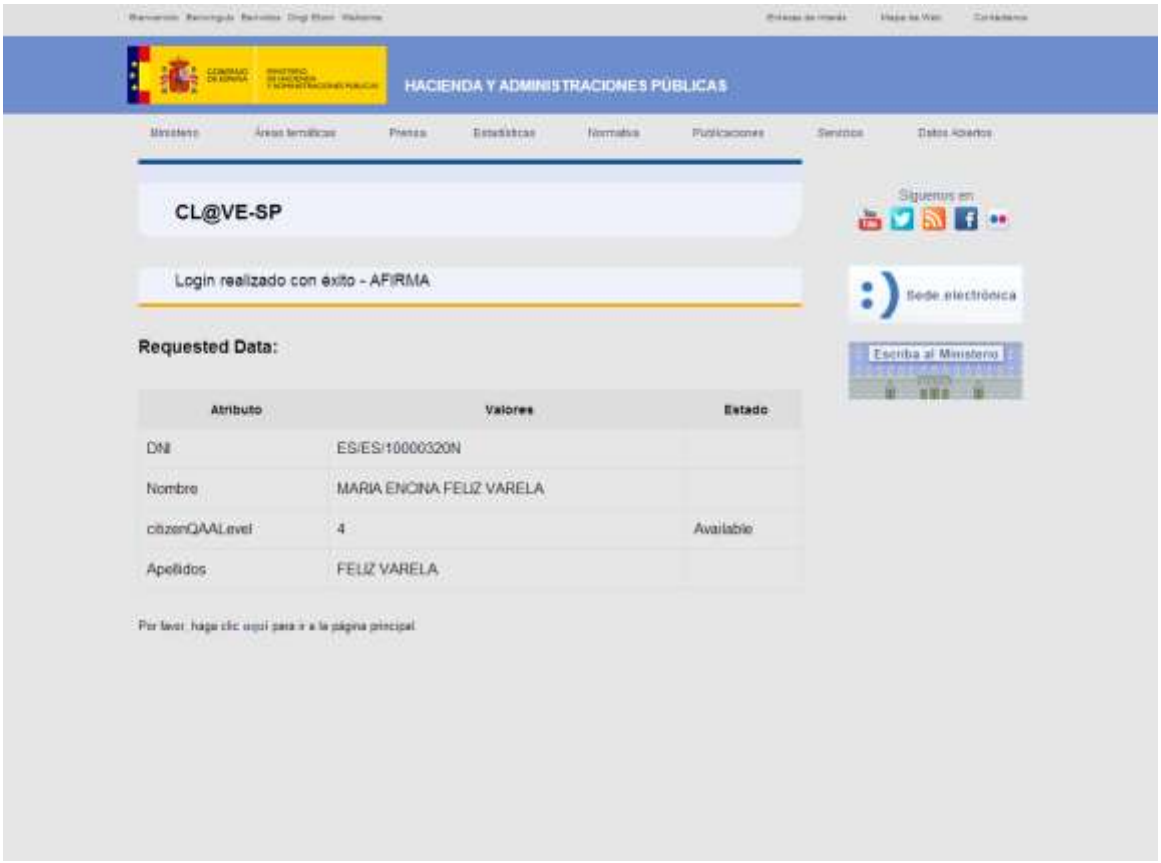

Figura 3 – Página de Retorno - DemoSP php

El fichero de control de versiones para SPs está disponible aquí: "http(s)://inserta.tu.ip.aqui/SP/spInfo.php".

6 SAML Engine API

6.1 Generando una petición de autenticación

Existe tres propiedades fundamentales que deben ser creadas antes de la petición SAML: extensions, SP metadata e IdP metadata. Sólo entonces se puede generar el SAML.

```
//el nombre del metadata del SP
$authSource = StorkConstants::SPID;

//cargar el metadata del SP
$as = SimpleSAML_Auth_Source::getById($authSource);
$metadata = $as->getMetadata();

//cargar el metadata del IdP. Para el parámetro 'country' se espera que sea
puesto por el HTML form
$idp = $_POST['spcountry'];
$idpMetadata = $as->getIdPMetadata($idp);

//cargar extensiones. Se comprueba si se ha pulsado el botón de "Lanzar
petición" o "Petición a medida"
$porDefecto = $_POST['default'];
if( $porDefecto )
       $extensions = StorkConstants::genDefaultAttrs($_POST);
else
       $extensions = StorkConstants::genAttrs($_POST);

//generar una Authentication Request
$ar = stork_saml_Message::buildAuthnRequest($extensions, $metadata,
$idpMetadata);
```

Después de que el SAML Request sea generados se debe realizar un POST hacia el IdP apropiado.

```
$b = new SAML2_StorkHTTPPost($_POST['country'], $_POST);
if ($porDefecto)
     $b->sendDefault($ar);
else
     $b->send($ar);
```

6.2 Validando y leyendo una respuesta de autenticación

Después de recibir un POST se debe validar la firma del SAML.

```
//obtener la response
$b = new SAML2_HTTPPost();
$response = $b->receive();

//obtener el metadata
$authSource = StorkConstants::SPID;
$as = SimpleSAML_Auth_Source::getById($authSource);
$metadata = $as->getMetadata();

//validar la firma
$retVal = stork_saml_Message::checkSign($metadata, $response);
if($retVal) {
    //obtener las assertions
    $assertions = $response->getAssertions();
```

```
    //obtener los atributos
    $attributes = $assertions[0]->getAttributes();
    //obtener el status de la saml response
    $status = $response->getStatus();
    if('urn:oasis:names:tc:SAML:2.0:status:Success' !== $status['Code']) {
          // Autenticación correcta!
    } else {
          // Autenticación Fallida!
    }
  } else {
     // Validación de firma fallida
     echo '<h2>Ha ocurrido un error’;
     echo '<p>La validación de la firma ha fallado.</p>';
}
```

7 Preguntas frecuentes

¿Cómo configurar un host virtual?

A continuación se muestra un ejemplo de las líneas que hay que añadir al fichero *httpd-vhosts.conf* para configurar un host virtual:

*<VirtualHost *:80>*

DocumentRoot "C:\apache\var\simplesamlphp\www"

ServerName localhost

</VirtualHost>

Configuraciones alternativas del virtual host de Apache:

En caso de que falle la configuración del host virtual, si sólo va a disponer de esta aplicación en su servidor, puede en su lugar modificar el fichero */conf/extra/httpd-ssl.conf* haciendo que la raíz del servidor sea la raíz del Service Provider. Para ello, defina bajo la directiva *<VirtualHost _default_:[puerto deseado]>* la siguiente información

DocumentRoot "[ruta donde vaya a alojarse el SP en el servidor]/var/simplesamlphp/www

¿Dónde puedo acudir a soporte?

mailto:stork@indra.es

8 ANEXO: Petición de Alta como Proveedor de Servicios con acceso al Intermediador de IdPs

Fecha_Petición	*Fecha actual*
Nombre	*Nombre del Proveedor de Servicio*
Descripción	
Entorno	*PRE o PRO*
Persona_Contacto	*Persona de Contacto del Proveedor de Servicio (teléfono y email)*
Cerificado_SP	*Parte pública del certificado usado por el Proveedor de Servicio en la firma SAML*
-----BEGIN CERTIFICATE----- -----END CERTIFICATE-----	
Fecha_ prevista	*Fecha de Inicio prevista para pruebas en el entorno indicado*

Nota que nombre del proveedor de servicio y el certificado de firma tienen que coincidir con los valores configurados en 4.1.1.