

Article

[Timothy Leavitt](#) · Mar 9, 2016 3m read

Example: Connecting to a Caché web service from PHP

There have been a few use cases recently within InterSystems where we've needed to connect to Caché-based web services from PHP. The first of these was actually the Developer Community itself, which uses web services as part of Single Sign-On with other InterSystems sites/applications. The following example demonstrates how to connect to a Caché-based web service (particularly, the web service in the SAMPLES namespace) from PHP, using password authentication.

(Note: This assumes password authentication is enabled for /csp/samples.)

```
<?php
// Standard SOAP header for username/password
// From http://stackoverflow.com/questions/13465168/php-namespaces-in-soapheader-child-nodes
class WSSESecurityHeader extends SoapHeader {
    public function __construct($username, $password)
    {
        $wsseNamespace = 'http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd';
        $security = new SoapVar(
            array(new SoapVar(
                array(
                    new SoapVar($username, XSD_STRING, null, null, 'Username', $wsseN
amespace),
                    new SoapVar($password, XSD_STRING, null, null, 'Password', $wsseN
amespace)
                ),
                SOAP_ENC_OBJECT,
                null,
                null,
                'UsernameToken',
                $wsseNamespace
            )),
            SOAP_ENC_OBJECT
        );
        parent::__construct($wsseNamespace, 'Security', $security, false);
    }
}

// Key parameters for connecting to the web service:
$location = "http://localhost:57772/csp/samples/SOAP.Demo.cls";
$uri = "http://tempuri.org";
$username = "_SYSTEM";
$password = "SYS";

// If the soap client is to be constructed from the WSDL, or if the WSDL is to be use
d to generate
// PHP classes (using various tools that exist for that purpose), ?WSDL=1 and the use
rname/password
// need to be tacked on to the end of the URL:
```

```
$wsdl = $location . "?WSDL=1&CacheUserName=" . $username . "&CachePassword=" . $password;

// Ideally, we'd then just be able to do something like:
// $client = new SoapClient($wsdl, array("trace"=>1));
// In many cases, that will be sufficient.
// However, the WSDL for SOAP.Demo in the SAMPLES namespace references additional schemas
// defined in other classes - for example:
// <s:import namespace="http://tempuri.org/QueryByName_DataSet" schemaLocation="http://localhost:57772/csp/samples/SOAP.Demo.QueryByName.DS.cls?XSD"/>
// There doesn't seem to be any way to make PHP's WSDL consumption use the CSP session cookie or continue
// to pass along the credentials in the URL when retrieving these, so it fails. The workaround is to create
// the SoapClient without the WSDL, using location/uri instead.

// Create SoapClient Object
$client = new SoapClient(null, array(
    "trace" => 1,
    "location" => $location,
    "uri" => $uri,
    "style" => SOAP_DOCUMENT,
    "use" => SOAP_LITERAL));

// Add security header.
$client->__setSoapHeaders(new WSSESecurityHeader($username, $password));

// Get a person by ID - we'll let this come from the URL but default to ID 1 so something's always shown.
$id = 1;
if (isset($_GET["id"])) {
    $id = $_GET["id"];
}

// Object with parameter names and values for the SOAP call
$request = new stdClass();
$request->id = $id;

// This goes in an associative array with key FindPersonSoapIn (see WSDL)
$params = array();
$params['FindPersonSoapIn']=$request;

// In the array of options for the soap call, soapaction must be specified since there's no WSDL to refer to.
// PHP provides a default value of <uri>#<method>, but that's incorrect.
$options = array('soapaction'=>$uri.'/SOAP.Demo.FindPerson');

// Actually call the web service
$result = $client->__soapCall("FindPerson",$params,$options);

// Just dump out the response to the page.
var_dump($result);
?>
```

[#Code Snippet](#) [#Frontend](#) [#SOAP](#) [#Caché](#)

Source URL: <https://community.intersystems.com/post/example-connecting-cach%C3%A9-web-service-php>