



Einleitung.....	2
Ueberblick.....	2
Zugriff.....	2
WSDL.....	2
Schnittstellen-Hilfe.....	2
SOAP-Request.....	2
PHP-Beispiele.....	3
Basis-Script.....	3
Aufruf Schnittstellen-Hilfe.....	3
Login.....	3
Quickmessages lesen und neuen User anlegen.....	3
Vertrauensbeziehung (Trust Relationship).....	4
Beziehung aufbauen.....	4
Login.....	4

Diese Beschreibung stellt den aktuellen Stand verschiedener interner Strukturen zusammen, die fuer die Benutzung der SOAP-Schnittstelle von Webweaver Suite benoetigt werden. Sie erhebt weder Anspruch auf Vollstaendigkeit noch auf Richtigkeit. Details koennen sich ohne Vorankuendigung aendern.

Einleitung

Ueberblick

Funktionsweise

Alle Aufrufe werden ueber die Methode "request()" getaetigt. Diese Methode erwartet als Parameter ein mehrdimensionales Array mit den nacheinander auszufuehrenden Befehlen. Die Antwort ist wieder ein mehrdimensionales Array mit einem Sub-Array pro Befehl und einem einem zusaetzlichen Sub-Array, welches u.a. die Session-ID enthaelt, die beim naechsten Aufruf zu verwenden ist.

Es sollten moeglichst viele Befehle in einem Aufruf gebuendelt werden.

Sitzung

Normalerweise sollte eine typische Sitzung wie folgt ablaufen:

```
Erster Aufruf:      login()
weitere Aufrufe:   set_session()
                   set_focus()
                   (...)
Letzter Aufruf:   set_session()
                   logout()
```

Beispiele

Alle Beispiel-Aufrufe sind PHP-Code. Diese setzen eine aktuelle PHP5-Version mit aktiviertem SOAP-Support voraus. Diese Beispiele sollten sich aber relativ problemlos in jeder anderen (SOAP-faehigen) Sprache nachbauen lassen.

Eingabe

Normalerweise wird beim Aufruf ein assoziatives Array uebergeben. Sollte Ihre Umgebung diese Art von Arrays nicht unterstuetzen, so kann jedes Set [Parameter=>Value] durch ein Sub-Array [Parameter,Value] ersetzt werden.

Ausgabe

Die Ausgabe erfolgt als assoziatives Array. Falls es beim Entgegennehmen zu Problemen kommt, koennen Sie das Array serialisiert als Zeichenkette zurueckliefern lassen. `set_options(['output'=>'serialized'])`

Return Codes und Fehlermeldungen

Bei falsch formatierten Befehlen wird eine SOAP-Exception generiert. Ansonsten wird in der Antwort zu jedem Befehl ein Wert "return" mitgegeben:

OK	Ausfuehrung OK und abgeschlossen
RESUME	Ausfuehrung nur teilweise: Beim naechsten Aufruf weiter
RETRY	Ausfuehrung uebersprungen: Beim naechsten Aufruf neu
ERROR	Fehler in diesem Befehl, weiter mit naechstem Befehl
FATAL	Fehler und Abbruch weiterer Befehle

Bei den Return-Codes "ERROR" und "FATAL" werden zusaetzlich die Werte "error" mit einer kurzen Fehlerbeschreibung und "errno" mit einer eindeutigen Fehlernummer zurueckgeliefert. "RESUME" liefert den "offset" fuer den naechsten Aufruf.

Zugriff

URL

`https://HOST/soap.php`

Beschreibung

Die gesamte Kommunikation findet HTTPS-verschluesselt ueber das Skript `soap.php` im Root der Installation statt.

Beispiel

`https://www.webweaver.de/soap.php`

WSDL

URL

`https://HOST/soap.php?wsdl`

Beispiel

`https://www.webweaver.de/soap.php?wsdl`

Schnittstellen-Hilfe

URL

`https://HOST/soap.php?help` oder `$client->help()`; oder `https://HOST/wws/559185.php`

Beschreibung

Liefert die aktuelle Liste aller Objekte, Methoden und Parameter.

SOAP-Request

PHP

```
$client->request(array(...));
```

Beschreibung

Der eigentliche Remote-Aufruf

PHP-Beispiele

Basis-Script

PHP

```
<?
$client=new SoapClient('https://www.webweaver.de/soap.php?wsdl',
    array('encoding'=>'ISO-8859-1','trace'=>1,
        'features'=>SOAP_SINGLE_ELEMENT_ARRAYS,'exceptions'=>0));

$result=$client->request(); // Aufruf

print_r($result);
?>
```

Beschreibung

Dieser PHP-Block wird fuer alle Aufrufe verwendet, der eigentlich Aufruf variiert von Beispiel zu Beispiel.

Aufruf Schnittstellen-Hilfe

PHP

```
$result=$client->help();
```

Beschreibung

Ausgegeben wird ein mehrdimensionelles assoziatives Array, welches u.a. saemtliche aktuell unterstuetzten Objekte, Methoden und Eigenschaften und auch die jeweils benoetigten Rechte liefert.

Login

PHP

```
$result=$client->request(array(
    array('method'=>'login','login'=>'test@webweaver.de','password'=>'pw_test'),
));
```

Beschreibung

Fuehrt den Login auf der Plattform durch. Das Passwort wird PLAIN uebergeben. Ausgegeben werden u.a. die Basisrechte des Users und alle Institutionen/Gruppen/Klassen inkl. der Rechte, in denen der User Mitglied ist.

PHP alternativ

```
$result=$client->request(array(
    array(array('method','login'),array('login','test@webweaver.de'),
        array('password','pw_test')),
));
```

Beschreibung

Der gleiche Aufruf wie zuvor, aber mit indizierten anstatt assoziativen Arrays.

PHP alternativ 2

```
$result=$client->request(array(
    array('method'=>'set_option','output'=>'serialized'),
    array('method'=>'login','login'=>'test@webweaver.de','crypt'=>md5('pw_test')),
));
```

Beschreibung

Ausgabe als serialisiertes PHP-Array (String). Login mit MD5-Hash. Diese Art des Logins ist nur dann moeglich, wenn die Plattform das Passwort der Nutzer selbst verwaltet und sollte deshalb nicht (mehr) verwendet werden.

Quickmessages lesen und neuen User anlegen

PHP

```
$result=$client->request([
    ['method'=>'set_session','session_id'=>'25578342991433259418310511'],
    ['method'=>'set_focus','object'=>'messenger'],
    ['method'=>'read_quick_messages'],
    ['method'=>'set_focus','login'=>'info@test.webweaver.de','object'=>'administration'],
    ['method'=>'add_user','user_name'=>'test','type'=>0,'fullname'=>'Test'],
]);
```

Beschreibung

1. Session-ID setzen
2. Focus auf den privaten Messenger
3. Lesen alle verfuegbaren Quickmessages
4. Focus auf die Institutions-Adminstration
5. Neuen User "test" mit Rolle 0 anlegen

Vertrauensbeziehung (Trust Relationship)

Damit das Passwort eines Nutzers nicht lokal abgelegt werden muss, kann ueber die SOAP-Schnittstelle eine Vertrauensbeziehung angelegt werden, die das Passwort beim Login ersetzt.

Beziehung aufbauen

PHP

```
$result=$client->request([
    ['method'=>'login','login'=>$l,'password'=>$p,'get_properties'=>[]],
    ['method'=>'set_focus','object'=>'trusts'],
    ['method'=>'register_master','remote_title'=>$t,'remote_application'=>$a],
    ['method'=>'logout']
]);
```

Beschreibung

trusts::register_master() legt eine neue Vertrauensbeziehung an. "remote_title" wird dem Nutzer in der Uebersicht seiner Vertrauensbeziehungen angezeigt. Ebenso der optionale Parameter "remote_ident". "remote_application" ist der interne Name der Applikation, die die Vertrauensbeziehung anlegt und sollte eindeutig sein (z. B. eine URL) und nur ASCII-Zeichen enthalten. Als Rueckgabe wird u. a. das Token geliefert:

```
['trust']['token'] => 'qVU1Xz2qCOqOURjmkG3i5u5j154XwDgj6TD7FQPWuyekAVesN4TZ1iSc5hrwhXBy'
```

Login

PHP

```
$url=false;
$result=$client->request([
    ['method'=>'get_nonce']
]);
if($result and ($result[0]['return']==='OK') and ($nonce=$result[0]['nonce'])) {
    $salt='salz (ASCII)';
    $algo='sha1';
    $token='qVU1Xz2qCOqOURjmkG3i5u5j154XwDgj6TD7FQPWuyekAVesN4TZ1iSc5hrwhXBy';
    $hash=sha1($nonce['key'].$salt.$token);
    $result=$client->request([
        ['method'=>'login','login'=>$l,'algorithm'=>'sha1','nonce_id'=>$nonce['id'],
        'salt'=>$salt,'hash'=>$hash,'application'=>$a,'get_properties'=>[]],
        ['method'=>'set_focus','object'=>'trusts'],
        ['method'=>'get_url_for_autologin','locale'=>'de'],
        ['method'=>'logout'],
    ]);
    if($result and ($result[0]['return']==='OK') and ($result[1]['return']==='OK')
        and ($result[2]['return']==='OK')) {
        $url=$result[2]['url'];
    }
}
echo $url;
```

Beschreibung

Der erste Request fordert eine Nonce vom Server an. Der zweite Request fuehrt den Login anhand des SHA1-Hashes vom Key der Nonce, einem Salz und dem Token durch. Der Parameter "application" muss mit "remote_application" beim Aufbau der Vertrauensbeziehung identisch sein. Anschliessend kann die SOAP-Schnittstelle wie nach einem "normalen" Login genutzt werden. Im Beispiel wird eine URL angefordert, die bei Aufruf im Browser den Nutzer einloggt.