Let CURL handle cookie list

A client asked about how to handle best a SOAP webservice, which handles authorization and session handling via cookie. You can query and set cookies in CURL sessions easily with the cookie functions. But the best here may be to just call MBS("CURL.SetOptionCookieFile"; $curl; "") and let CURL handle them. So each CURL session can include an internal cookie storeage activated by CURL.SetOptionCookieFile function. When a cookie is set on the server, CURL parses the response and adds the cookie to the cookie storage in memory. For the next query, matching cookies are automatically added to headers and send with the request.

Let's take the following script to query a SOAP servcice with first a login query and than an information query. As authorization is handled via cookie, we reuse the same $curl session and benefit from automatic cookie handling:

The URL of the service
Set Variable [$url ; Value: "http://somedomain.com/PartnerService17.asmx"]
Build XML for SOAP Login request, e.g. with MBS XML functions or simply subsitute()
Set Variable [$xml ; Value: "<?xml version=\"1.0\" encoding=\"utf-8\"?> <soap:Envelope xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" xmlns:soap=\"http://schemas.xmlsoap.org/soap/envelope/\>..."]
Set Field [Curl_Test::XML ; $xml]
The SOAP action to use
Set Variable [$action ; Value: "http://somedomain.com/partner/loginWithLicenceUnlockInfo"]
#
Setup transfer
Set Variable [$curl ; Value: MBS("CURL.New")]
Set Variable [$result ; Value: MBS("CURL.SetOptionURL"; $curl; $url)]
Put SOAP action and XML data type in the headers
Set Variable [$result ; Value: MBS("CURL.SetOptionHTTPHeader"; $curl; "Content-Type: text/xml; charset=utf-8"; "Expect:"; "SOAPAction: \""& $action & "\"")]
Ask CURL to manage cookies
Set Variable [$result ; Value: MBS("CURL.SetOptionCookieFile"; $curl; "")]
Set the payload for the login transfer
Set Variable [$result ; Value: MBS("CURL.SetOptionPostFields"; $curl; $xml)]
Run transfer
Set Variable [$result ; Value: MBS("CURL.Perform"; $curl)]
Let's take a look on the cookie list
Set Variable [$$cookie ; Value: MBS("CURL.GetCookieList"; $curl)]

```
Set Field [ Curl_Test::Cookie ; $$cookie ]
# Check result
Set Field [ Curl_Test::Debug ; MBS("CURL.GetDebugAsText"; $curl) ]
Set Variable [ $result ; Value: MBS("CURL.GetResultAsText"; $curl) ]
Set Field [ Curl_Test::Resultat ; $result ]
#
# Reuse same $curl for second request reusing existing cookies
#
# Make new XML for second request
Set Variable [ $npk ; Value: "103" ]
# Build XML for SOAP Login request, e.g. with MBS XML functions or simply subsitute()
Set Variable [ $xml ; Value: "<soap:Envelope xmlns:xsi=\"http://www.w3.org/2001/
XMLSchema-instance\" xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\"
xmlns:soap=\"http://schemas.xmlsoap.org/soap/envelope/\">..." ]
Set Field [ Curl_Test::XML ; $xml ]
Set Variable [ $action ; Value: "http://somedomain.com/partner/queryInfo" ]
#
# Setup transfer
Set Variable [ $result ; Value: MBS("CURL.SetOptionURL"; $curl; $url) ]
# Put SOAP action and XML data type in the headers
Set Variable [ $result ; Value: MBS("CURL.SetOptionHTTPHeader"; $curl; "Content-Type:
text/xml; charset=utf-8"; "Expect:"; "SOAPAction: \""& $action & "\"") ]
# Set the payload for the second transfer
Set Variable [ $result ; Value: MBS("CURL.SetOptionPostFields"; $curl; $xml) ]
# Run transfer
Set Variable [ $result ; Value: MBS("CURL.Perform"; $curl) ]
# Check result
Set Field [ Curl_Test::Debug ; MBS("CURL.GetDebugAsText"; $curl) ]
Set Variable [ $result ; Value: MBS("CURL.GetResultAsText"; $curl) ]
Set Field [ Curl_Test::Resultat ; $result ]
# Finally end curl session
Set Variable [ $r ; Value: MBS("CURL.Cleanup"; $curl) ]
```

If you have questions about our plugin, please do not hesitate to contact us.