

strongSwan - Bug #692

Curl fails to fetch HTTPS CRL URL

28.08.2014 06:42 - John Doe

Status:	Closed	Start date:	28.08.2014
Priority:	Normal	Due date:	
Assignee:	Martin Willi	Estimated time:	0.00 hour
Category:	libstrongswan	Resolution:	Fixed
Target version:	5.2.1		
Affected version:	5.2.0		
Description			
If the certificate contains a HTTPS url to connect to the CRL curl fails to connect. Here is a patch to make it work: --- src/libstrongswan/plugins/curl/curl_plugin.c 2013-10-18 12:38:17.000000000 1100 ++ src/libstrongswan/plugins/curl/curl_plugin.c 2014-08-27 15:15:19.277802550 +1000 @@ -79,7 +79,7 @@ },); - res = curl_global_init(CURL_GLOBAL_NOHING); + res = curl_global_init(CURL_GLOBAL_DEFAULT); if (res != CURLE_OK) { DBG1(DBG_LIB, "global libcurl initializing failed: %s",			
Related issues:			
Related to Issue #2570: libcurl http request failed: SSL: couldn't create a c...			Closed

Associated revisions

Revision 0c8c965c - 24.09.2014 17:37 - Martin Willi

Merge branch 'curl-features'

Enable missing [https:// support](https://support) in the curl plugin by initializing libcurl appropriately.

To initialize the SSL backend properly as required, we rely on our specific crypto backends (openssl, gcrypt) that already provide this functionality.

Fixes #692.

History

#1 - 28.08.2014 06:47 - John Doe

Here is the patch again because the formatting was screwed up in the original:

```
--- src/libstrongswan/plugins/curl/curl_plugin.c 2013-10-18 12:38:17.000000000 +1100
+++ src/libstrongswan/plugins/curl/curl_plugin.c 2014-08-27 15:15:19.277802550 +1000
@@ -79,7 +79,7 @@
     },
 );

- res = curl_global_init(CURL_GLOBAL_NOHING);
+ res = curl_global_init(CURL_GLOBAL_DEFAULT);
if (res != CURLE_OK)
{
    DBG1(DBG_LIB, "global libcurl initializing failed: %s",
```

#2 - 28.08.2014 10:41 - Martin Willi

- Assignee set to Martin Willi

#3 - 28.08.2014 11:22 - Martin Willi

- *Tracker changed from Issue to Bug*
- *Status changed from New to Assigned*
- *Target version set to 5.2.1*

Hi,

If the certificate contains a HTTPS url to connect to the CRL curl fails to connect.

I agree, seems that HTTPS support was missing in the curl fetcher backend.

Here is a patch to make it work:

Thanks for the patch. I'd prefer to use CURL_GLOBAL_SSL only, as we initialize Winsock2 anyway by ourselves. Also, we should probably have a fallback in case SSL initialization fails.

It's probably a good idea to dynamically query the protocols supported by the libcurl build, and announce supported plugin features only. I've pushed three patches to the [curl-features](#) branch, queued for master.

Regards
Martin

#4 - 28.08.2014 13:17 - Martin Willi

Seems that there is a bad interaction issue between our openssl plugin and libcurl if it uses the OpenSSL based SSL backend.

The problem is that the threading callbacks get registered, but can't be unregistered in our plugin. libcurl makes use of them during cleanup, but the openssl plugin is already gone. Not sure yet how to fix this.

#5 - 24.09.2014 17:48 - Martin Willi

- *Category set to libstrongswan*
- *Status changed from Assigned to Closed*
- *Resolution set to Fixed*

The associated merge commit fixes the issue.

When using OpenSSL with curl, we have to register appropriate threading callbacks for the SSL backend used by libcurl. To avoid any conflicts with our openssl and gcrypt plugins already doing that, we rely on these plugins to provide the appropriate functionality.

To fetch over HTTPS, this implies that you'll need the appropriate strongSwan plugin enabled for the libcurl SSL backend you are using. Refer to the [curl](#) Wiki page for details.

Regards
Marting

#6 - 01.03.2018 16:49 - Tobias Brunner

- *Related to Issue #2570: libcurl http request failed: SSL: couldn't create a context: error:140A90A1:lib(20):func(169):reason(161) added*