Smart Proxy - Bug #14719

Allow TLSv1 for compatibility with some clients.

04/19/2016 03:01 PM - Jason Smith

Status:	Rejected		
Priority:	Normal		
Assignee:			
Category:			
Target version:			
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	
Bugzilla link:		**Red Hat JIRA:**	
Pull request:	https://github.com/theforeman/smart-proxy/pull/408		

Description

After updating and testing foreman 1.11, our custom php scripts that talk to the foreman proxy through the REST API no longer work. After some debugging and looking at the php documentation, the problem is that TLSv1 is not allowed. According to some user comments in the php documentation:

http://php.net/manual/en/function.curl-setopt.php#115993

Setting php to use TLSv1 or above will only work if you have curl 7.34 or newer. Note, RHEL6 comes with curl 7.19 and RHEL7 comes with curl 7.29. To maintain compatibility with still supported RHEL versions and allow custom 3rd party scripts written in php to connect to the foreman-proxy REST API, need to allow TLSv1 also:

https://github.com/theforeman/smart-proxy/pull/408

Related issues:

Related to Smart Proxy - Bug #14387: SSLv3 remains enabled on Ruby 1.8.7	**Closed**	**03/29/2016**

History

#1 - 04/19/2016 03:01 PM - Jason Smith

- Related to Bug #14387: SSLv3 remains enabled on Ruby 1.8.7 added

#2 - 04/19/2016 03:02 PM - The Foreman Bot

- Status changed from New to Ready For Testing

- Pull request https://github.com/theforeman/smart-proxy/pull/408 added

#3 - 07/14/2016 06:47 PM - Daniel Gagnon

Having a similar issue communicating between foreman and a smart-proxy after an upgrade to 1.12 and debian8.

Foreman:
- debian 8 (upgraded from 7)
- OpenSSL 1.1.0-pre6-dev xx XXX xxxx (had to install custom version due to http://openssl.6102.n7.nabble.com/openssl-1-0-2h-Parsing-really-large-CRLs-fails-side-effect-of-change-in-x-name-c-tc65870.html#none)
- ruby 2.1.5p273 (2014-11-13) [x86_64-linux-gnu]
- foreman 1.12

Proxy:
- centos CentOS release 5.11 (Final)
- OpenSSL 0.9.8e-fips-rhel5 01 Jul 2008
- ruby 2.1.8p440 (2015-12-16 revision 53160) [x86_64-linux]
- smart proxy from git 1.12

Actual error in proxy log:

```
E, [2016-07-14T16:30:28.274669 #16809] ERROR -- : OpenSSL::SSL::SSLError: SSL_accept returned=1 errno=0 state=
SSLv2/v3 read client hello A: unknown
          protocol
```

Error from foreman:

```
From foreman:
Error: Unable to communicate with the proxy: ERF12-2530 [ProxyAPI::ProxyException]: Unable to detect features
([Errno::ECONNRESET]: Connection reset by peer - SSL_connect) for proxy https://pxesetup.clients.netelligent.c
a:8443/features and Please check the proxy is configured and running on the host.
```

The fix I have found, ont he proxy:
in lib/launcher.rb

```
ssl_options |= OpenSSL::SSL::OP_NO_SSLv2 if defined?(OpenSSL::SSL::OP_NO_SSLv2)
ssl_options |= OpenSSL::SSL::OP_NO_SSLv3 if defined?(OpenSSL::SSL::OP_NO_SSLv3)
ssl_options |= OpenSSL::SSL::OP_NO_TLSv1 if defined?(OpenSSL::SSL::OP_NO_TLSv1)
```

becomes:

```
#ssl_options |= OpenSSL::SSL::OP_NO_SSLv2 if defined?(OpenSSL::SSL::OP_NO_SSLv2)
#ssl_options |= OpenSSL::SSL::OP_NO_SSLv3 if defined?(OpenSSL::SSL::OP_NO_SSLv3)
#ssl_options |= OpenSSL::SSL::OP_NO_TLSv1 if defined?(OpenSSL::SSL::OP_NO_TLSv1)
```

I believe this indicates that foreman itself is trying to establish a connection with an older protocol.

#4 - 07/14/2016 07:11 PM - Daniel Gagnon

> I believe this indicates that foreman itself is trying to establish a connection with an older protocol.

correction. new guess is that openssl on centos 5 does not support anything above tls1, so that disabling v2, v3 and tls1 effectivly disables all available protocol.

#5 - 07/14/2016 07:16 PM - Brandon Weeks

That would be my guess as to what is happening CentOS 5 was not tested as part of this change and even early versions of CentOS 6 don't support TLS 1.2 completely.

#6 - 04/06/2018 08:51 AM - Ewoud Kohl van Wijngaarden

- Status changed from Ready For Testing to Rejected

The PR was closed and I don't think we should be allowing TLSv1 in this day and age. Clients should use current protocols.