Smuxi Issues [FROZEN ARCHIVE] - Bug # 802: SSL IRC connections usually only work on the first try (local

Status:	Closed	**Priority:**	Normal
Author:	vith	**Category:**	Engine IRC
Created:	01/19/2013	**Assigned to:**	Mirco Bauer
Updated:	05/21/2015	**Due date:**	
Complexity:	High		
Found in Version:	0.8.11.0 (master/b14d3e4)		
Subject:	SSL IRC connections usually only work on the first try (local engine)		

Description:

I can connect to morgan.freenode.net:6697 SSL once, but if I disconnect and reconnect I get this:

```
09:52 -!- Connecting to morgan.freenode.net port 6697...
09:52 -!- Connection failed! Reason: Could not connect to: morgan.freenode.net:6697 The authentication or decryption has failed.
```

However, I can still connect to a different SSL IRC server (turing.nullirc.net:6697 SSL). However, if I open a second connection tab to turing I sometimes get:

```
09:53 -!- Connecting to turing.nullirc.net port 6697...
09:53 -!- Connection failed! Reason: Could not connect to: turing.nullirc.net:6697 BeginWrite failure
```

If I close all NullIRC tabs, I can again connect to turing successfully. Attempting a second connection results in the same BeginWrite failure.

There may be some randomness involved to when connections succeed or fail. If you just keep opening new server tabs, alternating between servers, it can start working again.

Debug log is attached.

smuxi-frontend-gnome is built from git master/b14d3e4

built with ./autogen.sh --prefix=$HOME MCS=/usr/bin/dmcs && make && make install

Mono JIT compiler version 2.10.8.1 (Debian 2.10.8.1-5ubuntu1)

Linux vith-ubuntu-vm 3.5.0-22-generic #34-Ubuntu SMP Tue Jan 8 21:47:00 UTC 2013 x86_64 x86_64 x86_64 GNU/Linux

Ubuntu 12.10 64-bit Desktop in VMWare Workstation

Associated revisions

05/21/2015 04:40 PM - Mirco Bauer

Server, Frontend-GNOME: workaround TLS/SSL negotiation caching issues on Mono (closes: #802)

The symptom usually is that the first TLS/SSL connect works but no following
ones to a specific server.

History

01/19/2013 05:30 PM - Mirco Bauer

- Complexity set to High

- Found in Version changed from GNOME frontend 0.8.11.0 (master/b14d3e4) to 0.8.11.0 (master/b14d3e4)

I can reproduce the 2nd SSL connect fails issue with morgan.freenode.net on Mono 3.0.1.

Also the issue that sometimes (tried it 6 times) the SSL connection to turing.nullirc.net fails, but I get this message instead:

<pre>

```
17:29:26 -!- Connection failed! Reason: Could not connect to: turing.nullirc.net:6697 The authentication or decryption has failed.
```

</pre>

01/19/2013 05:47 PM - Mirco Bauer

Here a minimal test-case to reproduce this issue with just Mono:

<pre>

```
wget https://raw.github.com/mono/mono/master/mcs/class/Mono.Security/Test/tools/tlstest/tlstest.cs
dmcs tlstest.cs -r:Mono.Security
mono tlstest.exe https://morgan.freenode.net:6697 https://morgan.freenode.net:6697
```

</pre>

Output:

<pre>

```
meebey@redhorse:~/tmp$ mono tlstest.exe https://morgan.freenode.net:6697 https://morgan.freenode.net:6697

https://morgan.freenode.net:6697
[Subject]
  CN=*.freenode.net, OU=Gandi Standard Wildcard SSL, OU=Domain Control Validated

[Issuer]
  CN=Gandi Standard SSL CA, O=GANDI SAS, C=FR

[Not Before]
  1/14/2013 1:00:00 AM

[Not After]
  1/15/2014 12:59:59 AM

[Thumbprint]
  9C326008AF581A62A97053852CB53A6F320F5E67


 Valid From:  1/14/2013 1:00:00 AM
 Valid Until: 1/15/2014 12:59:59 AM

Error #-2146762486: CERT_E_CHAINING 0x800B010A

https://morgan.freenode.net:6697
FAILED: #-2146233087
System.IO.IOException: The authentication or decryption has failed. ---> Mono.Security.Protocol.Tls.TlsException: The server stopped the handshake.
  at Mono.Security.Protocol.Tls.SslClientStream.SafeReceiveRecord (System.IO.Stream s) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.SslClientStream.OnNegotiateHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  --- End of inner exception stack trace ---
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
```

</pre>

01/19/2013 05:52 PM - Mirco Bauer

When I tried to connect too often I get this:

```
<pre>
meebey@redhorse:~/tmp$ mono tlstest.exe https://morgan.freenode.net:6697 https://morgan.freenode.net:6697

https://morgan.freenode.net:6697
FAILED: #-2146233087
System.IO.IOException: EndWrite failure ---> System.Net.Sockets.SocketException: The socket has been shut down
  at System.Net.Sockets.Socket.EndSend (IAsyncResult result) [0x00000] in <filename unknown>:0
  at System.Net.Sockets.NetworkStream.EndWrite (IAsyncResult ar) [0x00000] in <filename unknown>:0
  --- End of inner exception stack trace ---
  at System.Net.Sockets.NetworkStream.EndWrite (IAsyncResult ar) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.RecordProtocol.EndSendRecord (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.RecordProtocol.SendRecord (ContentType contentType, System.Byte[] recordData) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.RecordProtocol.SendAlert (Mono.Security.Protocol.Tls.Alert alert) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0

https://morgan.freenode.net:6697
FAILED: #-2146233087
System.IO.IOException: The authentication or decryption has failed. ---> Mono.Security.Protocol.Tls.TlsException: The authentication or decryption has failed.
  at Mono.Security.Protocol.Tls.RecordProtocol.ReadRecordBuffer (Int32 contentType, System.IO.Stream record) [0x00000] in <filename unknown>:0

  at Mono.Security.Protocol.Tls.RecordProtocol.InternalReceiveRecordCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  --- End of inner exception stack trace ---
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
</pre>
```

This makes it pretty clear that the 2nd connect issue is not really about SSL handshake but more about some invalid state, probably cached.

01/19/2013 06:27 PM - Mirco Bauer

After studying some of Mono's SSL code, I found the cache and the default lifetime of it (180 seconds) and a way to change that default lifetime:

```
<pre>
MONO_TLS_SESSION_CACHE_TIMEOUT=0 mono --debug tlstest.exe https://morgan.freenode.net:6697 https://morgan.freenode.net:6697
</pre>
```

```
<pre>
meebey@redhorse:~/tmp$ MONO_TLS_SESSION_CACHE_TIMEOUT=0 mono --debug tlstest.exe https://morgan.freenode.net:6697 https://morgan.freenode.net:6697

https://morgan.freenode.net:6697
[Subject]
  CN=*.freenode.net, OU=Gandi Standard Wildcard SSL, OU=Domain Control Validated

[Issuer]
  CN=Gandi Standard SSL CA, O=GANDI SAS, C=FR

[Not Before]
  1/14/2013 1:00:00 AM

[Not After]
  1/15/2014 12:59:59 AM
```

```
[Thumbprint]
  9C326008AF581A62A97053852CB53A6F320F5E67



 Valid From:  1/14/2013 1:00:00 AM
 Valid Until: 1/15/2014 12:59:59 AM

Error #-2146762486: CERT_E_CHAINING 0x800B010A

https://morgan.freenode.net:6697
[Subject]
  CN=*.freenode.net, OU=Gandi Standard Wildcard SSL, OU=Domain Control Validated

[Issuer]
  CN=Gandi Standard SSL CA, O=GANDI SAS, C=FR

[Not Before]
  1/14/2013 1:00:00 AM

[Not After]
  1/15/2014 12:59:59 AM

[Thumbprint]
  9C326008AF581A62A97053852CB53A6F320F5E67



 Valid From:  1/14/2013 1:00:00 AM
 Valid Until: 1/15/2014 12:59:59 AM

Error #-2146762486: CERT_E_CHAINING 0x800B010A
```

</pre>

Trying it too often, the IRCd server starts to hate us again and simply closes the socket after accepting the connection:

<pre>

```
meebey@redhorse:~/tmp$ MONO_TLS_SESSION_CACHE_TIMEOUT=-1 mono --debug tlstest.exe https://morgan.freenode.net:6697

https://morgan.freenode.net:6697
FAILED: #-2146233087
System.IO.IOException: EndWrite failure ---> System.Net.Sockets.SocketException: The socket has been shut down
  at System.Net.Sockets.Socket.EndSend (IAsyncResult result) [0x0002d] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/System/System.Net.Sockets/Socket_2_1.cs:2032
  at System.Net.Sockets.NetworkStream.EndWrite (IAsyncResult ar) [0x0002f] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/System/System.Net.Sockets/NetworkStream.cs:346
  --- End of inner exception stack trace ---
  at System.Net.Sockets.NetworkStream.EndWrite (IAsyncResult ar) [0x0003d] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/System/System.Net.Sockets/NetworkStream.cs:348
  at Mono.Security.Protocol.Tls.RecordProtocol.EndSendRecord (IAsyncResult asyncResult) [0x00040] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/RecordProtocol.cs:721
  at Mono.Security.Protocol.Tls.RecordProtocol.SendRecord (ContentType contentType, System.Byte[] recordData) [0x0000b] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/RecordProtocol.cs:729
  at Mono.Security.Protocol.Tls.RecordProtocol.SendAlert (Mono.Security.Protocol.Tls.Alert alert) [0x00027] in /tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/RecordProtocol.cs:625
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00019] in
```

```
/tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/SslStreamBase.cs:98

https://morgan.freenode.net:6697
FAILED: #-2146233087
System.IO.IOException: The authentication or decryption has failed. ---> Mono.Security.Protocol.Tls.TlsException: The authentication or decryption
has failed.
     at Mono.Security.Protocol.Tls.RecordProtocol.ReadRecordBuffer (Int32 contentType, System.IO.Stream record) [0x00036] in
/tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/RecordProtocol.cs:458
       at Mono.Security.Protocol.Tls.RecordProtocol.InternalReceiveRecordCallback (IAsyncResult asyncResult) [0x0004b] in
/tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/RecordProtocol.cs:355
  --- End of inner exception stack trace ---
        at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x0002a] in
/tmp/buildd/mono-3.0.1+dfsg/mcs/class/Mono.Security/Mono.Security.Protocol.Tls/SslStreamBase.cs:100
```

</pre>

This means there still seems to be some caching involved, as the 2nd attempt does not fail like the 1st attempt...

01/19/2013 06:32 PM - Mirco Bauer

Disabling the session cache is not good enough to make turing.nullirc.net work, see here:

<pre>

```
meebey@redhorse:~/tmp$ MONO_TLS_SESSION_CACHE_TIMEOUT=0 mono tlstest.exe https://turing.nullirc.net:6697 https://turing.nullirc.net:6697



https://turing.nullirc.net:6697
[Subject]
  E=ms@nullirc.net, CN=turing.nullirc.net, OU=turing.nulirc.net, O=NullIRC, L=Dallas, S=Texas, C=US

[Issuer]
  E=ms@nullirc.net, CN=turing.nullirc.net, OU=turing.nulirc.net, O=NullIRC, L=Dallas, S=Texas, C=US

[Not Before]
  7/26/2009 6:16:04 AM

[Not After]
  8/25/2009 6:16:04 AM

[Thumbprint]
  9FF1725B5119B73CAD176BA545439BA78FD241FA



 Valid From:  7/26/2009 6:16:04 AM
 Valid Until: 8/25/2009 6:16:04 AM

Error #-2146762487: CERT_E_UNTRUSTEDROOT 0x800B0109

https://turing.nullirc.net:6697
FAILED: #-2146233087
System.IO.IOException: The authentication or decryption has failed. ---> Mono.Security.Protocol.Tls.TlsException: The server stopped the
handshake.
  at Mono.Security.Protocol.Tls.SslClientStream.SafeReceiveRecord (System.IO.Stream s) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.SslClientStream.OnNegotiateHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
  --- End of inner exception stack trace ---
  at Mono.Security.Protocol.Tls.SslStreamBase.AsyncHandshakeCallback (IAsyncResult asyncResult) [0x00000] in <filename unknown>:0
```

</pre>

05/14/2015 09:18 AM - Mirco Bauer

Smuxi has now a FAQ entry dedicated to SSL issues, see here: https://smuxi.im/faq/troubleshooting/linux-tls/

05/14/2015 09:19 AM - Mirco Bauer

- Target version set to 1.0

Since disabling the SSL caching is an easy quick fix, Smuxi should do that by default. Smuxi is not making many SSL connections, thus this shouldn't be a performance issue.

05/21/2015 05:16 PM - Mirco Bauer

- Status changed from New to Closed

- % Done changed from 0 to 100

Applied in changeset commit:"a68dfb5fdf5fb425b2eef043dffb753b8aebbdce".

Files

smuxi-local_engine_ssl_bug.txt	152.9 KB	01/19/2013	vith