# Quassel IRC - Bug #1509

## trouble auto-joining restricted channels on OFTC after client re-connect

01/08/2019 10:09 PM - bill-auger

| | | | | |
|---|---|---|---|---|
| **Status:** | New | | **Start date:** | 01/08/2019 |
| **Priority:** | Normal | | **Due date:** | |
| **Assignee:** | | | **% Done:** | 0% |
| **Category:** | | | **Estimated time:** | 0.00 hour |
| **Target version:** | Some future release | | | |
| **Version:** | 0.13.0 | | **OS:** | Any |

### Description

if i sign out cleanly manually, then reconnect manually, it auto-joins the restricted channel as expected - the problem seems to happen only when the client re-connects due to network dropout ("Disconnecting. (No Ping reply in 180 seconds.)") - on freenode with SASL this often results in an underscore being appended to my nick - it seems that both servers still consider me as signed in when quassel concludes has that i am disconnected

the symptom i noticed is this:

when i sign out cleanly then re-connect manually, i see "You are connected using SSL and have provided a matching client certificate ....  You have been automatically identified." and everything proceeds as expected

but when the client re-connects itself automatically, i see "Nick already in use:" before it verifies the cert; and i do not see the "You have been automatically identified." message at all - in it's usual place is the error: "Cannot join channel (Need to be identified and verified to join this channel"

### History

#### #1 - 01/22/2019 08:51 PM - Darkstar

This happened before in 0.12 too with Nickserv-based registrations. Reason is that the re-join happens too early, long before the Nickserv IDENT message is sent and acknowledged. Thus you have to re-join. It would be good to have an option to delay joining channels either by a fixed amount, or at least until Nickserv registration is done

#### #2 - 01/26/2019 06:20 AM - digitalcircuit

*- Target version set to Some future release*

Sadly, OFTC **still** does not support SASL, which both of you noted as the proper network-dependent fix.

Adding a delay in Quassel might work this and other timing issues, though at this point it'd make more sense in 0.14+ as 0.13 is in maintenance and bugfixing (*this is a long standing bug, not a new regression*).

For now, my non-certificate workaround for OFTC is to put this in Commands to execute on connect…

```
/ns identify PASSWORD your_nick
/wait 5; /ns reclaim your_nick
/wait 6; /nick your_nick
```

Alternatively, you could uncheck Rejoin all channels after reconnect and specify a bunch of manual /join #channel lines to run on connect.

#### #3 - 07/20/2020 02:37 PM - jjakob

I'm having the same issues, but using exclusively certificate certfp-based identify. It's my understanding there's no need for SASL password-based identify, as using a client certificate will use SASL EXTERNAL (that's what quassel-client says next to the SASL settings). But still, quassel starts to join channels before the server identifies my nick. IMO, a plain time delay isn't robust, as different conditions may cause the server-side identify to take longer than usual. It should wait for the server to say the nick is successfully identified before starting to join channels.

This is how every successful OFTC connection looks:

```
[14:04:52] [coulomb.oftc.net] *** Connected securely via TLSv1.2 ECDHE-RSA-AES256-GCM-SHA384-256
[14:04:52] [coulomb.oftc.net] *** Your client certificate fingerprint is <REDACTED>
[14:04:52] * #suckless: Cannot join channel (Need to be identified and verified to join this channel, '/msg Ni
ckServ help' to learn how to register and verify.)
[14:04:52] *** Mode jjakob +R by jjakob
[14:04:52] [NickServ] This nickname is registered and protected.  If it is your nickname, you may
```

```
[14:04:52] [NickServ] authenticate yourself to services with the IDENTIFY command.  If you do
[14:04:52] [NickServ] not authenticate or change your nickname, I will change your nickname.  You
[14:04:52] [NickServ] are getting this notice because you are not on the access list for the jjakob
[14:04:52] [NickServ] nickname.
[14:04:52] [coulomb.oftc.net] Activating Cloak: <REDACTED>
[14:04:52] [NickServ] You are connected using SSL and have provided a matching client certificate
[14:04:52] [NickServ] for nickname jjakob.  You have been automatically identified.
```

I have to rejoin #suckless on every reconnect.

This is how a normal Freenode connection looks:

```
[14:29:46] * Connecting to chat.freenode.net:6697...
[14:29:46] * Requesting capability list...
[14:29:46] [adams.freenode.net] *** Looking up your hostname...
[14:29:46] [adams.freenode.net] *** Checking Ident
[14:29:46] [adams.freenode.net] *** Couldn't look up your hostname
[14:29:54] * Ready to negotiate (found: extended-join, identify-msg, chghost, away-notify, account-notify, sas
l, multi-prefix, cap-notify, tls)
[14:29:54] * Negotiating capabilities (requesting: account-notify, away-notify, cap-notify, chghost, extended-
join, multi-prefix)...
[14:29:54] [adams.freenode.net] *** No Ident response
[14:29:54] * Capability negotiation finished (enabled: account-notify, away-notify, cap-notify, chghost, exten
ded-join, multi-prefix)
[14:29:54] * Welcome to the freenode Internet Relay Chat Network jjakob
[14:29:54] * Your host is adams.freenode.net[2a01:270:0:666f::1/6697], running version ircd-seven-1.1.9
<cut server info>
[14:29:54] *** Mode jjakob +Zi by jjakob
[14:29:54] [NickServ] This nickname is registered. Please choose a different nickname, or identify via /msg Ni
ckServ identify <password>.
[14:29:54] [NickServ] You are now identified for jjakob.
```

No errors here, but sometimes it still fails to join some channels (this is from a previous connection):

```
[19:36:29] [NickServ] This nickname is registered. Please choose a different nickname, or identify via /msg Ni
ckServ identify <password>.
[19:36:29] [NickServ] You are now identified for jjakob.
[19:36:36] * 470 #systemd-unregistered #systemd Forwarding to another channel
```

**#4 - 07/20/2020 08:20 PM - digitalcircuit**

After looking into this more, first I was mistaken, you do NOT have to enable the SASL checkbox jjakob, sorry for that.  You'd only need to do that if OFTC actually supported SASL, which it does not.  I overlooked that [Quassel sets the certificate regardless of SASL before it even connects](#).

Secondly, OFTC does not support SASL.

[6:15:28 pm] [coulomb.oftc.net] *** No Ident response
[6:15:28 pm] * Ready to negotiate (found: multi-prefix)

If SASL was supported (password or certificate), the IRC server itself should cause Quassel to wait before it allows Quassel to try to join any channels.

[IRCv3 Client Capability Negotiation](#)

> Upon receiving either a CAP LS or CAP REQ command during connection registration, the server MUST not complete registration until the client sends a CAP END command to indicate that capability negotiation has ended. This allows clients to request their desired capabilities before completing registration.

And Quassel does not [send CAP END/continue caps until SASL authentication has completed](#), password or certificate.

Capability negotiation with SASL **is** the reliable way of handling authentication without plain time delays, so I don't think there'd be any way around this on Quassel's side other than introducing a timer, new client-side UI, protocol change with feature flag, and database change to store it.

**#5 - 11/19/2020 01:12 AM - dmick**

OFTC **does** support SASL now, but the problem persists:

- Connecting to irc.oftc.net:6697...
- Requesting capability list...
  [charm.oftc.net] * **Looking up your hostname...**
  **[charm.oftc.net]** Checking Ident
  [charm.oftc.net] **Found your hostname**

**[charm.oftc.net]** * No Ident response

- Ready to negotiate (found: multi-prefix)
- Negotiating capabilities (requesting: multi-prefix)...
- Capability negotiation finished (enabled: multi-prefix)
- Welcome to the OFTC Internet Relay Chat Network dmick
- Your host is charm.oftc.net[64.62.190.36/6697], running version hybrid-7.2.2+oftc1.7.3
- This server was created Sep 22 2018 at 19:20:28
- charm.oftc.net hybrid-7.2.2+oftc1.7.3 CDGPRSabcdfgijklnorsuwxyz bciklmnopstvzeIMRS bkloveI
- CALLERID CASEMAPPING=rfc1459 DEAF=D KICKLEN=160 MODES=4 NICKLEN=30 PREFIX=(ov)+ STATUSMSG=+ TOPICLEN=391 NETWORK=OFTC MAXLIST=beI:100 MAXTARGETS=1 CHANTYPES=# are supported by this server
- CHANLIMIT=#:120 CHANNELLEN=50 CHANMODES=eIqb,k,l,cimnpstzMRS ELIST=CMNTU SAFELIST AWAYLEN=160 KNOCK EXCEPTS=e INVEX=I are supported by this server

[charm.oftc.net] **\* Connected securely via TLSv1.2 ECDHE-RSA-AES256-GCM-SHA384-256**
**[charm.oftc.net]**  Your client certificate fingerprint is <fingerprint>
.
.
.
[16:06:21] * #sepia: Cannot join channel (Need to be identified and verified to join this channel, '/msg NickServ help' to learn how to register and verify.)
[16:06:22] [NickServ] This nickname is registered and protected.  If it is your nickname, you may
[16:06:22] [NickServ] authenticate yourself to services with the IDENTIFY command.  You are
[16:06:22] [NickServ] getting this message because you are not on the access list for the
[16:06:22] [NickServ] dmick nickname.
[16:06:22] ** Mode dmick +R by dmick
[16:06:22] [NickServ] You are connected using SSL and have provided a matching client certificate
[16:06:22] [NickServ] for nickname dmick.  You have been automatically identified.

It seems like the order of requests is just wrong.  I'm using the Windows core and client; how can I see a log of what was actually exchanged between server and client?

**#6 - 11/19/2020 03:54 AM - genius3000**

dmick wrote:

> OFTC **does** support SASL now, but the problem persists:
>
> ...

Sorry but what is shown in your log snippet is **not** SASL authentication. The server is accepting your client certificate in the TLS connection and **after** connection passes that certificate onto the services server which matches it to your nick/account and identifies you. Unfortunately that still happens after you're connected, Quassel has no control over it and therefore the attempt to join before identification issue remains.

To be able to see the raw messages back and forth you have to start the core with a 'debug-irc' or similar flag enabled. I don't recall the exact flag nor which version it was first released in, it will show in the core's help output on the command line though. I think you can then log these lines or just run in a non-daemon mode, outputting to the command window.

**#7 - 11/23/2020 10:59 AM - digitalcircuit**

Confirming what genius3000 mentions, SASL is only used if the Use SASL Authentication checkbox is enabled.  The Quassel client interface in Networks incorrectly claims that SASL EXTERNAL is used regardless of the checkbox, which is confusing and being addressed in https://github.com/quassel/quassel/pull/547 - sorry about that (it's something I overlooked when reorganizing the SASL settings a few years ago).

If SASL was supported, you would see something like this in the Status Buffer:

```
* Negotiating capabilities (requesting: sasl, account-notify, away-notify, cap-notify, chghost, extended-join,
 multi-prefix)...
* SASL: You are now logged in as digitalcircuit.
* Capability negotiation finished (enabled: sasl, account-notify, away-notify, cap-notify, chghost, extended-j
oin, multi-prefix)
* Welcome to the freenode Internet Relay Chat Network digitalcircuit
[...]
```

SASL is requested as part of capability negotiation - if it's not shown in the "enabled" line, it's not enabled.

For logging all messages to/from the IRC network, launch the core with --debug-irc --loglevel=Debug --logfile="C:\path\to\writable\logfile" - the latter part might not be required (I don't have my Windows VM ready at the moment to test this).