Quassel IRC - Bug #1857

The DH1080 protocol is missing the encryption type in the response

05/14/2023 01:03 AM - bakasura

Status:	New	**Start date:**	05/14/2023
Priority:	Urgent	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Version:	0.14.0	**OS:**	Any

Description

Hi,

When analyzing the communications using Wireshark during a "/keyx" operation, it was observed that when initiated from a Quassel client, the encryption type is not sent in the DH1080_INIT message. Similarly, when initiated from an external client, the DH1080_FINISH message does not return the encryption type.

Here are a couple of examples:

Example 1: Initiated by Quassel client:

Quassel: DH1080_INIT XXX < No encryption type sent
Other: DH1080_FINISH XXX ??? < Unpredictable encryption type received

Example 2: Initiated by external client:

DH1080_INIT XXX CBC
DH1080_FINISH XXX < No encryption type sent

DH1080_INIT XXX ECB
DH1080_FINISH XXX < No encryption type sent

Upon reviewing the code, I found the lines responsible for receiving and sending these messages:

DH1080_INIT:
https://github.com/quassel/quassel/blob/b2deed91ef0275ec42aec717294dad01b33e8ded/src/core/coreuserinputhandler.cpp#L451

DH1080_FINISH:
https://github.com/quassel/quassel/blob/b2deed91ef0275ec42aec717294dad01b33e8ded/src/core/coresessioneventprocessor.cpp#L855

As you can see, both are missing the final step of sending the encryption type.

Without this, external clients will be either compatible or incompatible based on their default parameters and their ability to detect errors when sending or receiving encrypted messages (through analysis of "+OK" and "+OK *" types). In other words, compatibility will be highly inconsistent.

Best regards.

History

#1 - 10/09/2024 12:36 PM - sm1rky

Hello,

I have been annoyed by this very same problem for years.
Quassel is completely missing CBC key exchange functions. And when doing keyx it will do a ECB handshake like you described but via the default cbc mode mechanism in cipher::setKey() it will use CBC while the other party uses ECB leading to obvious decoding errors.

I fixed all of this and some more stuff (topic stack overflow) in the pull request on github https://github.com/quassel/quassel/pull/620
you may wanna use both if you're using the qt client.

Not sure if any mainters will integrate that so build your own quassel meanwhile. It should be enough to build core, replace it on your server and still use mobile clients. Only qt client is broken with the topic decryption.

Best Regards