

# Программа дуплексной связи оператора технической поддержки с клиентом

Команда “АПКШ <<Континент>>”

## 1 Цель кейса:

Разработка программного обеспечения для дуплексной связи, позволяющего абоненту устанавливать видеосвязь с оператором технической поддержки с использованием терминала на базе ОС Windows.

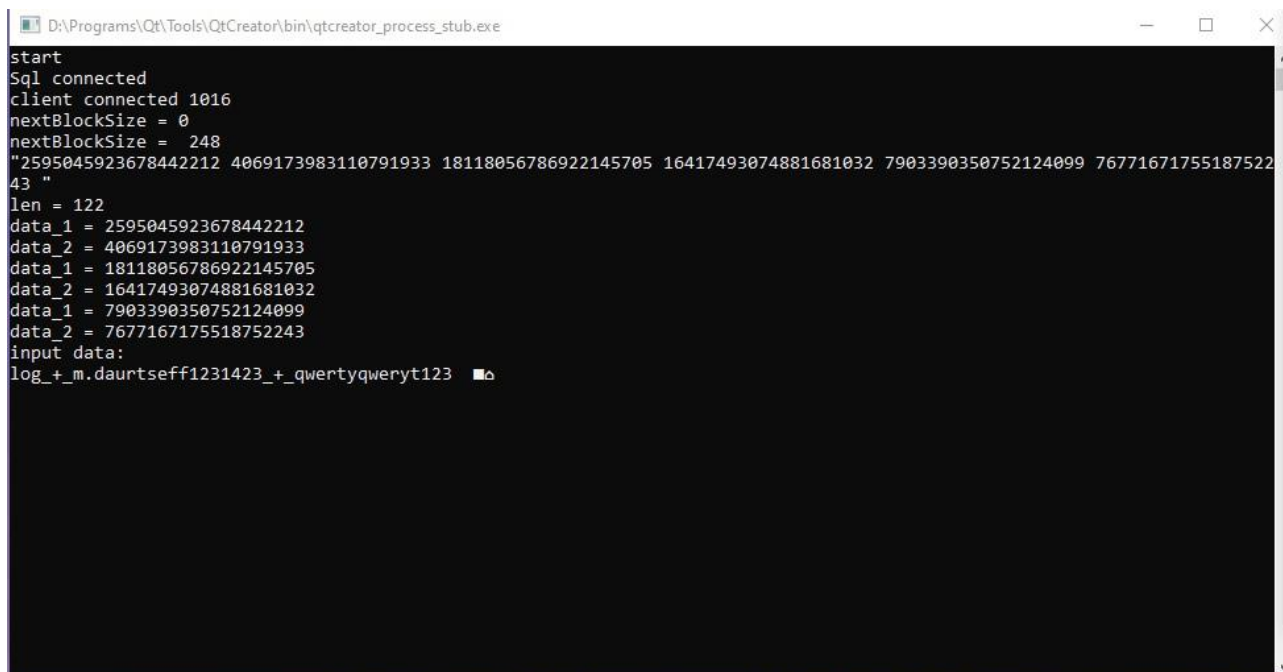
## 2 Отчет о выполненной работе:

Выполнить все задания кейса у команды не получилось, т.к. все силы были направлены на изучения новых компетенций и закрепления их на практике, а именно изучение криптографических алгоритмов шифрования данных.

Эта часть была успешно реализована в работе, на примере логирования пользователя.

## 3 Алгоритм работы программы:

Со стороны сервера запускается файл “server.exe” (рис. 1), а со стороны клиента файл “ZVKS.exe”(рис.2), имеющие следующие интерфейсы пользователей:



```
start
Sql connected
client connected 1016
nextBlockSize = 0
nextBlockSize = 248
"2595045923678442212 4069173983110791933 18118056786922145705 16417493074881681032 7903390350752124099 7677167175518752243 "
len = 122
data_1 = 2595045923678442212
data_2 = 4069173983110791933
data_1 = 18118056786922145705
data_2 = 16417493074881681032
data_1 = 7903390350752124099
data_2 = 7677167175518752243
input data:
log+_m.daurtseff1231423+_qwertyqweryt123
```

Рисунок 1 – файл “server.exe”

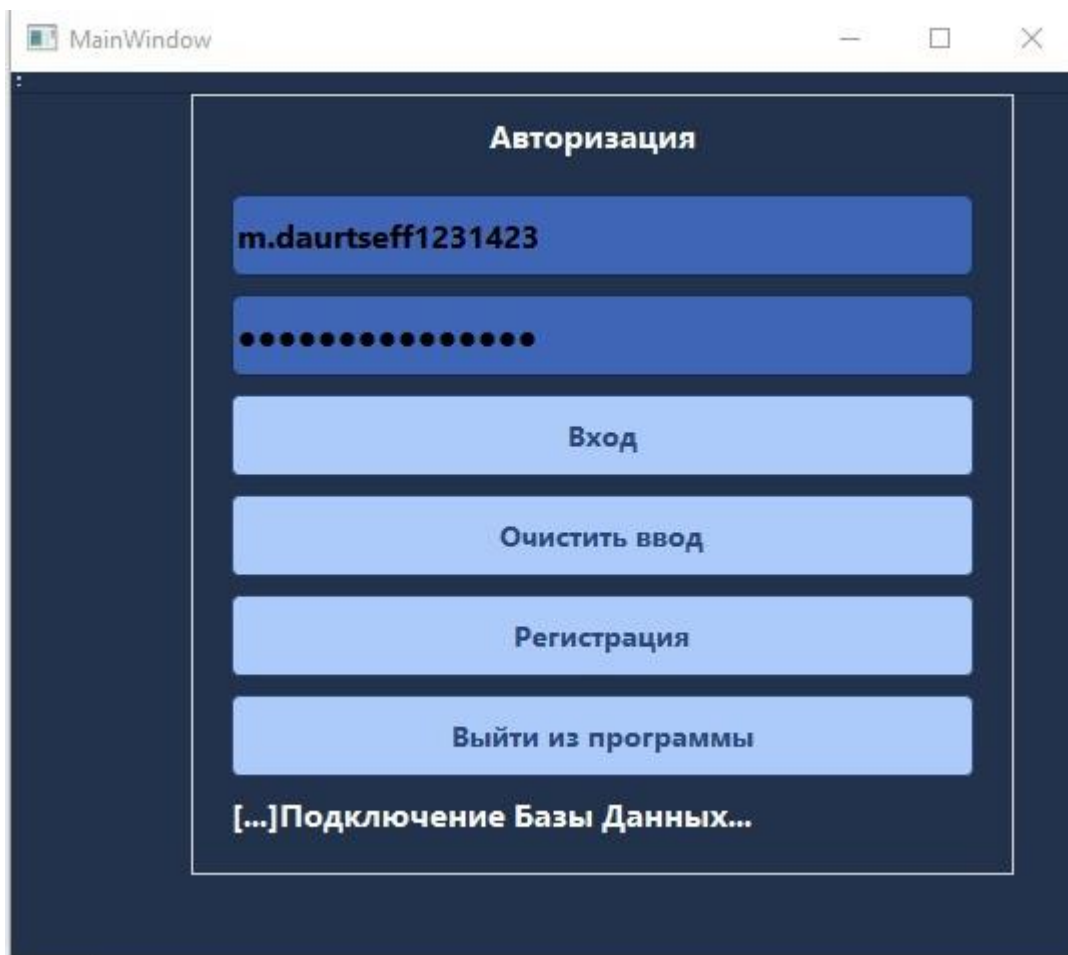


Рисунок 2 – файл “ZVKS.exe”

Между клиентом и сервером при запуске создается сокет TCP соединения по порту 2323 (IP адрес прописан 127.0.0.1, так что обе программы необходимо запускать на одном устройстве в следующей последовательности: сначала файл “server.exe” затем файл “ZVKS.exe”)

Сервер был подключен к базе данных SQL (Папка “Database”)

Со стороны пользователя открывается интерфейс авторизации в систему с функциональными кнопками (Работает только одна кнопка “Вход”), куда пользователь может вносить свои данные.

При нажатии кнопки “Вход” данные пользователя шифруются криптографический алгоритм блочного шифрования «Кузнечик» с длиной блока  $n = 128$  бит (определенный в ГОСТ Р 34.12-2015)

Со стороны сервера отслеживаются все отправления пользователя и происходит дешифровка данных (Процесс отображен более наглядно):

- Сначала идут блоки байт, которые рассчитаны для TCP соединения
- Затем строка шифр-текста
- Длина шифр-текста
- Разбиение шифр-текста на группу блоков для дешифровки

- После строки “input\_data:” дешифрованное сообщение.

Так как была использована IDE Qt Creator, то это означает возможность выхода данного приложения на другие ОС (Linux, Android).

#### **4. Запланированные реализации**

Реализовав шифрование данных, следующем пунктом для работы было бы построение чата-диалога между пользователем и тех.сотрудником, ведь текст намного проще шифровать чем файлы и поточные данные.

Следующим шагом было бы изучение классов QCamera и QMultimedia для постройки шифрованной закрытой видеоконференции.