Binary file operations

fstream output(file, std::ios::out | std::ios::binary | std::ios::trunc);

—->We open the binary file (std::ios::binary) for writing (std::ios::out).

For the SecureData::backup member function, we read the file from the text.dat to copy the same context.

fstream input(file, std::ios::in | std::ios::binary);

—->We open the binary file (std::ios::binary) for reading (std::ios::in).

Binding a function to its arguments

—->To bind the function and its arguments.

std::bind(converter, text, key, nbytes / 4, Cryptor());

std::bind(converter, text + nbytes / 4, key, nbytes / 4, Cryptor())]);

std::bind(converter, text + nbytes / 2, key, nbytes / 4, Cryptor());

std::bind(converter, text + 3\*nbytes / 4, key, nbytes / 4, Cryptor());

Multi-threading

void SecureData::code(char key) {

//converter(text, key, nbytes, Cryptor());

std::thread t1(std::bind(converter, text, key, nbytes / 4, Cryptor()));

std::thread t2(std::bind(converter, text + nbytes / 4, key, nbytes / 4, Cryptor()));

std::thread t3(std::bind(converter, text + nbytes / 2, key, nbytes / 4, Cryptor()));

std::thread t4(std::bind(converter, text + 3\*nbytes / 4, key, nbytes / 4, Cryptor()));

t1.join();

t2.join();

t3.join();

t4.join();

encoded = !encoded;}

—->We create the multi-thread to process the file faster.