Цифровая подпись

Необходимо создать пару программ — клиент и сервер — на языке C/C++, которые позволяют подписывать произвольное текстовое сообщение ключом, или проверять имеющуюся цифровую подпись сообщения.

Клиентская часть (frontend), позволяет пользователю выполнить одну функцию из списка путём отправки команды серверу. Сервер, в свою очередь, выполняет команду, и отвечает результатом (успешным или описанием ошибки).

Команда пользователя задаётся как первый аргумент командной строки, а сообщения поступают на вход через stdin, а результаты (положительные) — через stdout, ошибки — через stderr.

Сервер запускается в единственном экземпляре, и обрабатывает поступающие запросы от множества клиентов в произвольном порядке. Для каждого подписанного документа сервер должен делать запись в журнал (log-файл) с отметкой времени запроса на подпись, и хешем подписанного содержимого.

С точки зрения пользователя эта пара программ позволяет делать следующее:

1. Сгенерировать ключевую пару (собственный ключ), и сохранить её в файл в формате BER на сервере. Если ключ уже был создан ранее, то прервать процесс создания с ошибкой;
2. Подписать текстовое сообщение (из stdin) собственным ключом;
3. Проверить подпись сообщения, и выдать отпечатки ключей, использованных для его подписания в stdout. Если подписи не были идентифицированы или повреждены, тогда выдать сообщение об этом в stderr.
4. Уничтожить сгенерированный ключ.

Требования

1. Серверу нельзя передавать закрытый ключ клиентам.
2. Сервер должен стабильно работать как минимум с двумя клиентами;
3. Реализацию криптографических функций взять из библиотеки на выбор (OpenSSL, PolarSSL, или д.р.).
4. Исходный код решения должен включать CMakeLists.txt достаточный для сборки под Linux. Сама сборка должна быть осуществима сразу после клонирования репозитория с решением, и требовать только наличия установленного компилятора, и пакета CMake версии 3.15, и требуемых библиотек.
5. Допустимо использовать стандартную библиотеку, и небольшие библиотеки, выполняющие свои специфичные функции. Использование Qt крайне нежелательно.

Опции

1. + за аутентификацию клиентов посредством PSK;
2. + за унификацию списка команд в единой кодовой базе (и выделение общего модуля сборки);
3. + за корректное поведение программы при краевых, но вероятных случаях (сервер не запущен, сервер долго отвечает и т. п.), то же — для сервера.