

Клиент-серверное приложение для синхронизации файлов bFileServer/FileClient

Дата обновления документа: 05.04.2023

Терминология

«**Сервер**» bFileServer (далее «Сервер») – java-приложение для синхронизации файлов. Сервер запускается в Windows или Linux и ожидает подключения клиентов FileClient.

«**Клиент**» FileClient (далее «Клиент») – java-приложение для синхронизации локальных файлов клиента с сервером bFileServer.. Текущая версия Клиента (версия 0.1.0) проверена для работы с среде Windows.

«**Рабочая папка Клиента**» - место, где клиент хранит свои локальные документы. По-умолчанию "C:/temp/JavaFileClient/".

Сервер

Сервер протестирован в работе на Windows 10 и Debian 11.

Преимущества запуска на Windows или Linux не обнаружено.

Серверу нужны две директории:

1) **файлы для запуска программы** (где угодно в файловой системе, главное, чтобы прав доступа хватало).

Исполняемые файлы, конфиг server.conf и др. Например, это может быть «/opt/FileServer/bin», или «C:/temp/JavaFileServer/».

2) **директория, в которой будут создаваться папки с файлами пользователей при синхронизации.** Эта директория должна быть указана в конфигурационном файле server.conf. Например, «/opt/FileServer/storage».

Не принципиально, где находится директория с исполняемыми файлами.

Запуск сервера:

```
> cd /opt/FileServer  
> java -jar bFileServer.jar
```

Сервер ожидает подключения Клиентов на порт tcp 5556 (по-умолчанию). Номер порта может быть изменен.

Клиент

Клиент после запуска отслеживает изменения файлов в рабочей папке клиента. Клиент версии 0.1.0 хорошо себя показал в Windows 10. Работа в Linux проверена в Debian 11.

Запуск клиента:

```
C:\Users\user\FileClient > java -jar bFileServer.jar
```

При запуске Клиент устанавливает соединение с Сервером, проходит аутентификацию и синхронизирует содержимое своей рабочей папки с Сервером. Клиент запускает GUI-приложение. Если все прошло успешно, Клиент готов к работе. При изменении файлов Клиента такие файлы синхронизируются с Сервером.

Директория с рабочими файлами (документы, изображения и т.п.) Клиента по-умолчанию: "C:/temp/JavaFileClient/".

Адрес:порт Сервера по-умолчанию: "127.0.0.1:5556", протокол tcp.

Основные события, касающиеся работы с файлами клиента, отображаются в графической оболочке:

Menu

Activity log О приложении

```
05.04.23 14:50:22 Соединение с сервером "192.168.88.2":"5556" установлено.
05.04.23 14:50:22 Рабочая папка: "C:/temp/JavaFileClient/
05.04.23 14:50:22 Выполняется синхронизация с сервером...
05.04.23 14:50:23 Текущая версия базы сервера: 3457
05.04.23 14:50:23 Текущая версия базы клиента: 3457
05.04.23 14:50:23 Новая версия базы синхронизации: 3458
05.04.23 14:50:23 Начальная синхронизация с сервером завершена.
```

Особенности работы клиента на Linux

Для работы клиента в Linux необходимо скопировать в одну из папок библиотек (например, «/usr/lib64»), которые доступны java, библиотеку «libjnotify.so». Эта библиотека необходима для работы модуля, отслеживающего изменения в рабочей папке пользователя.

Особенности работы с редактируемыми документами

Практика показала, что программы, работая с файлами (например, при редактировании документов) создают временные файлы в той же директории, в которой находится исходный файл. Особенно это актуально для программ типа LibreOffice и для программ Linux в общем. Внесенные изменения сначала записываются во временный файл, который потом переименовывается в исходный. Для пользователя это прозрачно, но с точки зрения работы модуля, отсеживающего изменения в директории, постоянно происходят операции открытия, чтения записи в файлы, причем часто эти файлы недоступны для чтения или записи со стороны клиента, так как открыты редактором или системой. Поэтому чисто практикой подбираются по маске временные файлы, которые клиент должен игнорировать и не синхронизировать с сервером. Например, это файлы «.-lock.имя_файла», «".имя_файла.swp"», «имя_файла.TMP.tmp» и другие. С выходом новых версий программы будут обновляться фильтры. Список фильтров будет обновляться и публиковаться. При работе необходимо учитывать, что созданный вручную файл myData.TMP.tmp не будет синхронизирован с сервером. Таких названий следует избегать.

В Windows аналогичные проблемы тоже есть, хотя встречаются реже. Например, LibreOffice создает временные файлы вида «что_то_там~Wf_что_то_тут» в папке с исходным документом.

Возможно, в будущем эту проблему выйдут решить другим способом, без фильтров по временным файлам, но пока это так, как есть.

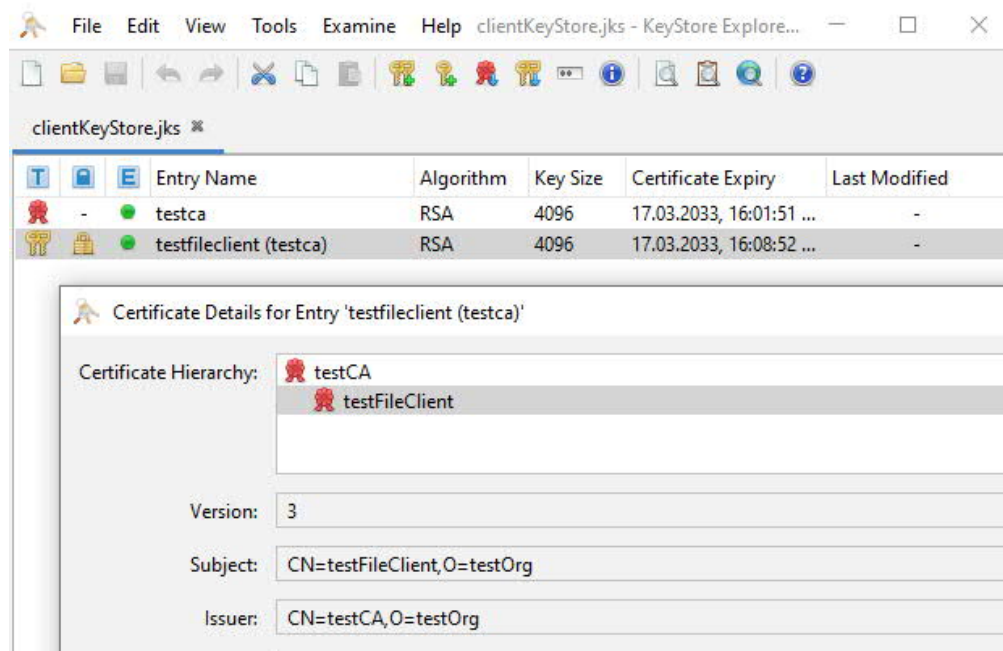
Аутентификация пользователей

Клиент и Сервер используют SSL-сертификаты и ключи для аутентификации и защиты трафика.

Аутентификация пользователя производится только по ключевой паре, без ввода логина/пароля.

Клиент использует контейнер ключей формата PKCS #12 для работы с Сервером.

Примечание: для подготовки или работы с контейнерами ключей в среде Windows рекомендую бесплатную программу KeyStore Explorer.



Сертификаты клиента и сервера должны быть выданы одним и тем же удостоверяющим центром (не обязательно нужен сертифицированный CA, для работы достаточно использовать самоподписанные сертификаты).

В контейнере Клиента должны быть:

- 1) сертификат CA;
- 2) ключевая пара из приватного ключа и сертификата клиента.

В контейнере Сервера должны быть:

- 1) сертификат CA;
- 2) ключевая пара из приватного ключа и сертификата сервера.

Важно! При успешной синхронизации с Сервером на сервере создается директория с именем пользователя, взятым из сертификата (поле CN, CommonName).

В тестовом сертификате пользователя CN=testFileClient. Соответственно, на сервере будет создана директория testFileClient. Поэтому **поле CN сертификата должно быть уникальным в рамках клиент-серверной инфраструктуры.**

Таким образом, подбор паролей невозможен в принципе. Это не исключает отсутствие уязвимостей в ПО, но снижает вероятность получения доступа к документам пользователя со стороны третьих лиц.

Заявление о конфиденциальности

Клиент и Сервер не отправляют никаких данных никаким третьим лицам, не собирают и не отправляют статистику использования. Сервер ждет входящих подключений на порт по-умолчанию 5556, а клиент может совершать исходящие соединения только на порт и адрес сервера. Других соединений с любыми другими узлами локальной или глобальной сетей не совершается. При обнаружении аномальной активности программы прошу немедленно связаться с разработчиком программы.

Отказ от ответственности

Данное ПО предоставляется бесплатно и «как есть». Разработчик не несет ответственности за прямой и/или косвенный ущерб, причиненный использованием данного ПО.

Использованные сторонние библиотеки

В разработке ПО использованы библиотека jNotify (<https://jnotify.sourceforge.net/>) , sqlite-jdbc.

Контакты разработчика

Email: ivanov.ilya@bozza.ru

<https://bozza.ru>