**Техническое задание на разработку интеграционного решения ПО «Сигур» и САБ «ИРБИС64+» в целях контроля и управления турникетами в здании Иркутской государственной универсальной научной библиотеки им. И. И. Молчанова-Сибирского**

# Оборудование и ПО

1. Турникет № 1 – ВХОД (access point: 1)
2. Турникет № 2 – ВЫХОД (access point: 2)
3. Контроллеры управления турникетами «Сигур E2» (2 шт.)
4. Персональный компьютер с установленным ПО Сигур (далее **СИГУР**). Технические характеристики ПК: iCore 5, 4 Гб ОЗУ, ОС Windows 10 Pro. ПК введен в домен l.irklib.ru, установлен антивирус Kaspersky Lab.

Турникеты оборудованы считывателями карт Mifare и считывателями штрих-кодов. Контроллеры подключены к ПК с установленным СИГУР по ethernet. ПК подключен к ЛВС ИОГУНБ (ethernet 100 Mbps).

1. Проектируемое интеграционное ПО (далее «***Привратник***»), представляющее собой REST-сервис, размещается на том же ПК, что и СИГУР для управления контроллерами турникетов и имеет GUI с описанным

# Безопасность

«Привратник» принимает запросы по небезопасному протоколу HTTP, но только с разрешенных конечных точек, перечень которых задается в файле конфигурации.

«Привратник» устанавливается в отдельную папку, разрешения на чтение/запись в которую имеются лишь у следующих учетных записей:

* BUILTIN\System,
* BUILTIN\Network Service,
* Администраторы.

Настройки «Привратника» хранятся в файле **appsettings.json** без шифрования (включая учетные данные, необходимые для подключения к серверу ИРБИС64). Также без шифрования хранятся в файловой системе запросы на авторизацию прохода и журнал.

# Конечные точки подключения

«Привратник» предоставляет две конечные точки:

* **/auth** для авторизации прохода через турникеты,
* **/** для пользовательского интерфейса.

Запросы к этим точкам подключения принимаются только с адресов, заданных в конфигурационном файле (в т. ч. доступен вариант разрешения для всех возможных адресов).

# Взаимодействие с операционной системой

Привратник представляет собой системный сервис, автоматический запускаемый от учетной записи **BUILTIN\Network Service**. Он функционирует полностью автономно и не требует присутствия пользователя.

# Схема взаимодействия

1. ПО «Привратник» запускается при старте ПК и работает в режиме прослушивания порта NNNN[[1]](#footnote-1).
2. СИГУР получая от контроллеров запрос с данными считанного идентификатора пользователя (далее **keyHex**) обращается к локальному адресу 127.0.0.1:**NNNN**, передавая в запросе UID и номер турникета (далее **access point**)
3. «Привратник» анализирует полученный **access point** и, в зависимости от него, выполняет следующие действия

## Алгоритм работы «Привратника» для турникета на ВХОД

1. Если запрос от СИГУР содержит access point: 1[[2]](#footnote-2)
   1. Кэширует keyHex, дату и время запроса
   2. Обращается к серверу ИРБИС64+ (далее **ИРБИС**)[[3]](#footnote-3) с запросом количества записей в БД RDR по запросу (RI=*keyHex*^NR=$)[[4]](#footnote-4)
      1. В случае недоступности ответа от сервера в течение ??? сек
         1. отправляет СИГУР ответ с кодом *allow: false*
         2. подает звуковой сигнал и выводит в графический интерфейс Оператора сообщение вида: «ВНИМАНИЕ! Сервер ИРБИС64 недоступен. Примите решение о пропуске посетителя с картой № *keyHex* в ручном режиме»
      2. В случае возврата ИРБИС числа записей = 0
         1. отправляет СИГУР ответ с кодом *allow: false*
         2. подает звуковой сигнал и выводит в графический интерфейс Оператора сообщение вида: «ВНИМАНИЕ! Посетитель с картой № *keyHex* не найден в базе данных читателей, либо лишен права обслуживания. Попросите посетителя пройти на ресепшн»
      3. В случае возврата ИРБИС числа записей > 1
         1. отправляет СИГУР ответ с кодом *allow: false*
         2. подает звуковой сигнал и выводит в графический интерфейс Оператора сообщение вида: «ВНИМАНИЕ! В базе данных читателей обнаружена множественность идентификатора *keyHex*. Попросите посетителя подойти на ресепшн»
      4. В случае возврата ИРБИС числа записей = 1
         1. отправляет СИГУР ответ с кодом *allow: true*
         2. выводит в графический интерфейс Оператора сообщение вида: «Зафиксирован вход читателя с пропуском № *keyHex*»
         3. отправляет в ИРБИС запрос на изменение записи в БД RDR по запросу RI=keyHex
            1. в запись добавляет новое повторение поля 40 с содержанием: **^d*ГГГГММДД*^1*ЧЧ:ММ:СС*^c(Посещение)^isigur^v\***.[[5]](#footnote-5)
            2. В случае отсутствия в записи поля 999, добавляется поле 999 с содержанием: **1**

## Алгоритм работы «Привратника» для турникета на ВЫХОД

1. Если запрос от СИГУР содержит access point: 2[[6]](#footnote-6)
   1. Кэширует keyHex, дату и время запроса
   2. Отправляет СИГУР ответ с кодом *allow: true*
   3. выводит в графический интерфейс Оператора сообщение вида: «Зафиксирован выход читателя с пропуском № *keyHex*»
   4. Обращается к ИРБИС с запросом на корректировку записи БД RDR по запросу RI=*keyHex*
   5. В случае если запрос возвращает ноль или более одной записи в БД RDR ничего не пишется
   6. отправляет в ИРБИС запрос на изменение записи в БД RDR по запросу RI=keyHex
      1. В записи анализируются повторения поля 40 на одновременное наличие хотя бы в одном повторении данных вида **^d*ГГГГММДД*** и **^c(Посещение)^isigur^v\*[[7]](#footnote-7)**
      2. Если такое повторение поля 40 в записи имеется, то все повторения поля 40 отвечающие указанному условию (^dГГГГММДД и ^c(Посещение)^isigur^v\*), проверяются на наличие в них строки **^f*ГГГГММДД*^2**
         1. если такая строка отсутствует, то каждое такое повторение поля 40 ДОПОЛНЯЕТСЯ следующими данными: **^f*ГГГГММДД*^2*ЧЧ:ММ:СС***.
         2. если такая строка присутствует во всех повторениях поля 40 отвечающих указанному условию (^dГГГГММДД и ^c(Посещение)^isigur^v\*), то в записи создается новое повторение поля 40 следующего содержания **^d*ГГГГММДД*^1*ЧЧ:ММ:СС* ^c(Посещение)^isigur^v\*^f*ГГГГММДД*^2*ЧЧ:ММ:СС***
         3. В случае присутствия в записи поля 999 оно ***удаляется***.
      3. если такое повторение поля 40 отсутствует, то создается новое повторение поля 40 следующего содержания **^d*ГГГГММДД*^1*ЧЧ:ММ:СС* ^c(Посещение)^isigur^v\*^f*ГГГГММДД*^2*ЧЧ:ММ:СС***
   7. в случае сбоев связи и ошибок обновления записи записывает в log-файл ошибок выхода (logouterror.log) три строки следующего содержания: **#10: *keyHex*\n#40: ^f*ГГГГММДД*^2*ЧЧ:ММ:СС*\n\*\*\*\*\*\n**

# Пользовательский интерфейс

Пользовательский интерфейс представляет собой обычную веб-страницу по адресу **http://localhost:NNNN/**, содержимое которой формируется «Привратником».

Помимо элемента предназначенного для оперативного информирования оператора о событиях приложения, должны присутствовать четыре счетчика с динамически обновляемыми данными

1. число входов (allow: true на access point 1),
2. число отказов во входе (allow: false на access point 1)
3. число выходов (allow: true на access point 2)
4. число запросов к серверу ИРБИС в очереди на обработку

Сброс значений счетчиков происходит в 00:00:00 по локальному времени

# Другие детали реализации

Полученные запросы аутентификации прохода через турникеты сериализуются и сохраняются в виде отдельных JSON-файлов без шифрования. По мере синхронизации с сервером ИРБИС64 файлы с запросами удаляются. Такой подход позволяет не беспокоиться об отключении питания или зависании операционной системы.

ВИЖУ ПРОБЛЕМЫ из-за сериализации заданий на входе:

сервер недоступен, создано задание на проверку, оператору отправлено сообщение о ситуации с предложением решить вопрос самостоятельно.

А) Оператор решает пропускать и жмет кнопку на открытие турникета. Читатель проходит. Сервер отвис, а читатель оказывается лишен права обслуживания/дублетный/отсутствует в БД. В обычной логике, оператору опять вылетает сообщение о необходимости принять решение? но решать уже нечего.

Б) Оператор решает не пропускать. Читатель не проходит. Сервер отвис, а с читателем в базе все в порядке. Фиксируется посещение, которого фактически не было.

Т. е. по идее надо бы, конечно, как-то уведомлять «Привратника» об операторском произволе, но кнопка у нас автономна.

Возможно, имеет смысл выполнение сериализованных (отложенных) запросов делать в “quiet”-режиме. При этом «проблемных» (не найденных/лишенных права обслуживания/ дублетных) по умолчанию не фиксировать в RDR «вход». У нас на выходе логика все равно настроена таки образом, что если покидающий зону пользователь не имеет в 40 полях «незакрытого» входа в тот же день, ему автоматом добавляется вход/выход. Так что «лишенцы» будут учтены, а вот с «дублетниками» и «чужаками» проблема – не вижу вариантов кроме как игнорировать.

# Журналирование

Журнал «Привратника» ведется в виде текстовых файлов, отдельный файл на каждые сутки. Это позволяет просто удалять слишком старые файлы журнала без прерывания работы «Привратника».

В журнал записываются следующие события:

* Запуск и (штатный) останов «Привратника»,
* Результат тестирования подключения к серверу ИРБИС64 при запуске «Привратника»,
* Полученные запросы на авторизацию прохода через турникеты, факт получения ответа от сервера ИРБИС64 (или факт сбоя), принятое решение об авторизации.

Дополнительно ведется отдельный журнал отказов авторизации, позволяющий быстро реагировать на инциденты (в т. ч. через пользовательский интерфейс).

1. Значение прослушиваемого порта должен браться из конфигурационного файла [↑](#footnote-ref-1)
2. Значение должно браться из конфигурационного файла [↑](#footnote-ref-2)
3. Данные для подключения к ИРБИС (ip-адрес, порт, логин и пароль для входа) должны браться из конфигурационного файла [↑](#footnote-ref-3)
4. Запрос должен браться из конфигурационного файла [↑](#footnote-ref-4)
5. где ГГГГММДД и ЧЧ:ММ:СС – дата и время запроса полученного от СИГУР в указанных форматах [↑](#footnote-ref-5)
6. Значение должно браться из конфигурационного файла [↑](#footnote-ref-6)
7. Формат для анализа: val((if p(v40) then if v40:'**^d*ГГГГММДД****'* and v40:'**^c(Посещение)^isigur^v\***' then '1;',break,else '0;' fi fi))>0 [↑](#footnote-ref-7)