

Использование для сетевой передачи голоса стандартных протоколов.

Введение.

Программное обеспечение диспетчерского комплекса «Обь» практически полнофункциональное, в том смысле, что его можно установить на компьютер под Windows, возможно, дополнительно установить на компьютер MS SQL сервер и начать использовать лифтовые блоки, т.е. начать диспетчеризацию лифтов.

Однако, как и при использовании большинства ПО, появляются требования пользователей, которые сложно или не рационально реализовать средствами нашего ПО и нужно иметь возможность интеграции со сторонним ПО — готовым или разрабатываем самими пользователями.

Под интеграции со сторонним ПО будем понимать обмен информацией между ПО ДК «Обь» и иным ПО.

Информация, которой обменивается ПО, можно разделить на два существенно разных класса:

1. Данные о работе оборудования, людей - диспетчеров, механиков, данные об изменении среды работы системы, назовём их для дальнейшего использования телеметрические данные;
2. Голосовые данные — разговоры диспетчеров, операторов Call центров, с пользователями нашего оборудования через наше оборудование (лифтовые блоки, переговорные устройства).

Хотя оба эти класса информации — цифровые данные, требования к ним разные.

Для обмена телеметрическими данными в ПО ДК «Обь» уже достаточно давно используется стандартный протокол прокол OPC DA <https://opcfoundation.org/> Это протокол имеет существенные ограничения:

1. стороннее ПО должно работать по Windows
2. ПО ДК «Обь» и стороннее ПО должно работать на одном компьютере

Что бы обойти эти ограничения в LKDSDomain был реализован начальный вариант OPC UA <https://opcfoundation.org/> сервера. Протокол OPC UA изначально много платформенный и сетевой. OPC UA сервер, реализованный в LKDSDomain, будет развиваться, вплоть до работы с базой данных (возможность запроса исторических данных). Возможно OPC UA будет реализован и в нашем облачном сервере LKDSCloud.

Протокол OPC UA вполне достаточен для обмена телеметрическими и историческими данными. Использование стандартизированного протокола даёт значительные преимущества по сравнению с использованием «самописных» протоколов:

1. наличие разнообразных готовых SCADA систем, позволяющих отобразить информацию ДК «Обь» в нужном для пользователя виде
2. хорошая документированность
3. возможность обучения
4. наличие специалистов, которые могут помочь в использовании и поддержке готовых SCADA систем, в разработке специфического ПО

5. знания, полученные в результате освоения OPC UA, пригодятся на долго, может и на всю жизнь.

Но OPC UA пока не предоставляет возможность обмена голосовыми данными.

Делалась попытка опубликовать передачу голоса, которая реализована внутри ПО ДК «Обь» путем обмена звуковыми пакетами через COM интерфейсы. При этом подходе есть всё те же проблемы, что и при использовании OPC DA:

1. стороннее ПО должно работать по Windows
2. ПО ДК «Обь» и стороннее ПО должно работать на одном компьютере

С другой стороны, у наших пользователей всегда было желание направлять вызовы от оборудования ДК «Обь» на обычные телефоны. Эти желания превратились в требования при реализации нашего оборудования за рубеж, т.к. как таковых диспетчерских пунктов в нашем понимании там нет. Есть Call центры, возможно где-то очень далеко, где стоимость рабочей силы меньше, а квалификация достаточная, например, в Индии. Вызовы направляются в эти Call центры, а телеметрические данные обрабатываются уже на месте на штатном ПО ДК «Обь» или стороннем ПО.

Таким образом, передача голосовых данных нужна по двум причинам:

1. обмен со сторонним ПО
2. использование сторонних Call центров для обработки заявок

Т.к. современная телефония уже давно цифровая, то нужно просто обеспечить в ПО ДК «Обь» реализацию протоколов цифровой телефонии и это должно закрыть эти обе причины. В настоящее время для цифровой телефонии используется группа протоколов (SIP, SDP, RTP, RTSP, транспортные протоколы UDP, TCP и т.п.) Все эти протоколы используются вместе для обеспечения одного сеанса голосовой связи. Исторически цифровая телефония называется SIP телефонией, хотя протокол SIP используется не только в телефонии. Используется и другое название VoIP (Voice over Internet Protocol) телефония или просто IP телефония, может это и правильно, но в дальнейшем буду использовать термин SIP телефония, SIP телефон и SIP протокол.

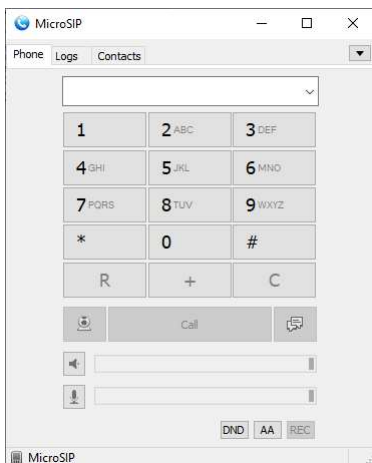
SIP телефон — это по сути программа на компьютере или микропрограмма в устройстве с микрофоном и громкоговорителем, которые реализуют SIP протокол.

Аппаратный SIP телефон просто имеет вместо разъёма RJ11 разъём RJ45 для подключения в компьютерную сеть. Мобильные и обычные телефоны также могут быть доступными через SIP протокол, но не непосредственно по IP адресу, а через специальный шлюз, доступный по IP сети и требующий дополнительной авторизации и договорных отношений с провайдером, например, МТС, Билайн и т.п.

Начальный вариант реализации SIP протокола в сервере LKDSDomain.

Любой путь начинается с первого шага и это первый шаг — реализация SIP протокола для пере направления вызова от ЛБ непосредственно на SIP телефон, т.е. не используются SIP шлюзы и дополнительная авторизация. SIP телефон это:

1. Программа, способная принимать SIP вызовы, например, MicroSIP
<http://www.microsip.org/>



2. Аппаратный SIP телефон, например, Grandstream GXP1615
<http://www.grandstream.com/>

3. Возможно и цифровая АТС, хотя экспериментов не проводилось.

Сам SIP протокол реализован в отдельной службе LKDSSIPsrv. Сделано это для уменьшения влияния ошибок реализации SIP протокола на сам сервер приложений LKDSDomain.

Расположена служба LKDSSIPsrv в папке \LKDSSIP\ установленного дистрибутива. Эта папка содержит следующие файлы:

LKDSSIP.dll

LKDSSIP1.dll

LKDSSIPsrv.exe

LKDSSIPsrvPS.dll

Сейчас служба LKDSSIPsrv не требует настройки. В дальнейшем настройка, конечно, потребуется и будет сделан конфигуратор. Для единообразия, конфигуратор будет иметь имя LKDSSIPsrvCfg.exe.

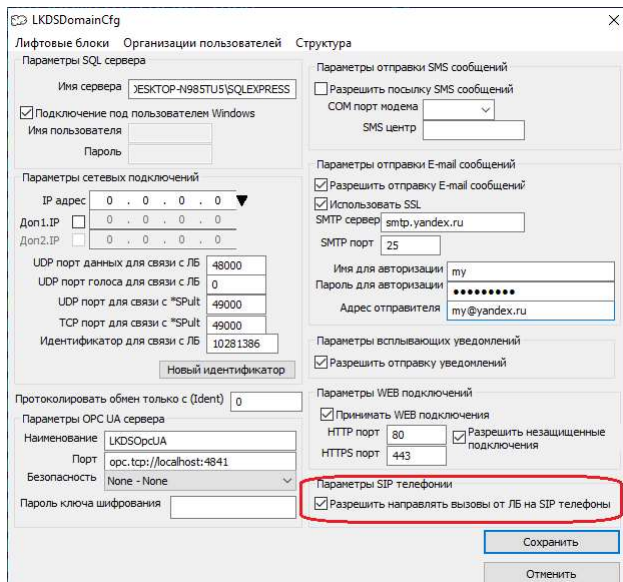
Сейчас служба LKDSSIPsrv одновременно поддерживает не более двух SIP соединений.

Сам SIP протокол реализован с использованием свободно распространяемой библиотеки с открытым кодом PJSIP <https://www.pjsip.org/>. На основе этой же библиотеки написано приложение MicroSIP <http://www.microsip.org/>. Исходные тексты приложения MicroSIP так же доступны, библиотека PJSIP содержит примеры использования и вполне можно написать своё приложение для приёма голосовых данных от ПО ДК «Обь».

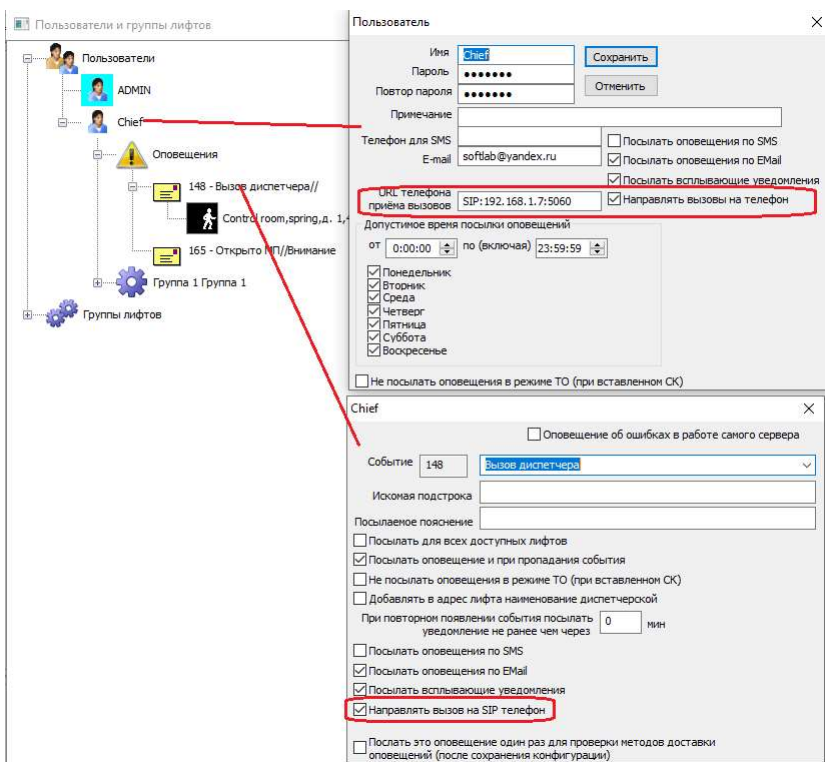
Пока LKDSSIPsrv обеспечивает только исходящие вызовы, т.е. позвонить на ЛБ нельзя.

Настройка LKDSDomain для использования SIP подключений.

В конфигураторе LKDSDomainCfg необходимо разрешить SIP соединения:



Затем в Spult, запущенном от имени пользователя - администратора организации сервера LKSDomain, нужно в параметрах пользователя указать URL адрес SIP телефона обычного пользователя, на который будут перенаправляться вызовы, и разрешение на это перенаправление. Кроме того, в параметрах оповещения о вызове нужно указать необходимость направления этого вызова на SIP телефон:



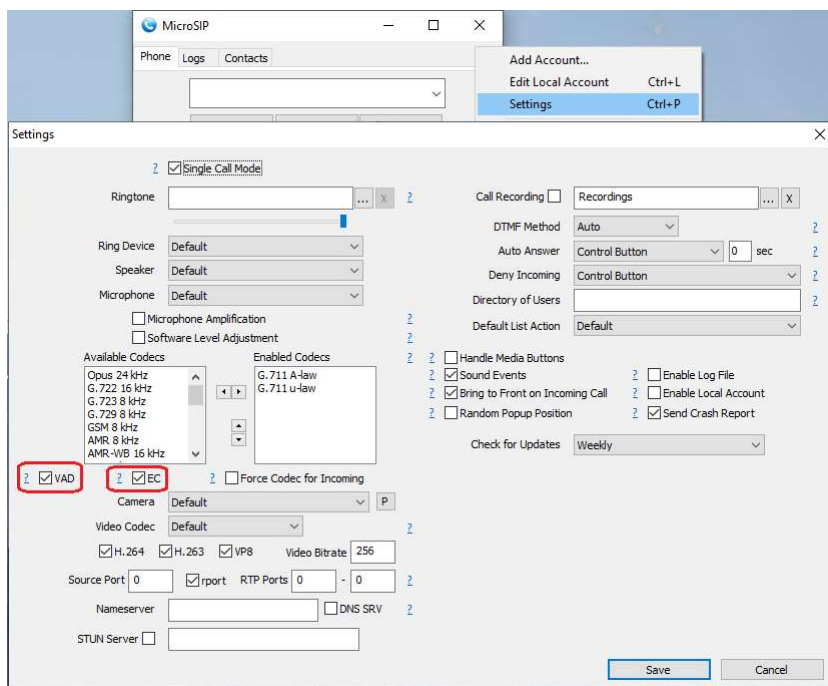
Если направление вызова с данного лифта разрешено на SIP телефоны нескольких пользователей, то будет использован только один, первый попавшийся. Перебора телефонов, если не удалось дозвониться, не будет. Перебор телефонов, наверное, может быть реализован цифровой АТС, если указать групповой номер.

Сейчас существует проблема, может быть не столь существенна и реально, но она есть - если один ЛБ описан в нескольких подъездах (в разных организациях или одновременно в разных ПКЛШ, LKDSPro, в прямом подключении) и в этих подъездах разрешено направление вызова на SIP телефоны, то результат будет некорректен.

Требования к используемому SIP телефону.

Внутри ДК «Обь» голосовые данные могут ходить только в одном направлении, т.е. либо от пассажира лифта (от ЛБ) в диспетчерскую, либо от диспетчерской пассажиру лифта (в ЛБ). Управляет этим диспетчер - либо с помощью кнопки на трубке, либо с помощью клавиатуры.

Если вызов направляется на SIP телефон, то этот SIP телефон должен сообщать LKDSSIPsrv – произносит ли что-нибудь диспетчер в трубку SIP телефона или молчит. В зависимости от этого LKDSSIPsrv формирует голосовые пакеты, содержащие направление разговора, для оборудования. Существует специальный алгоритм VAD (voice activity detection), т.е. определение голосовой активности, позволяющий это сделать. Это алгоритм встроен в LKDSSIPsrv, но желательно его включить и в SIP телефоне. В приложении MicroSIP это делается следующим образом:



Кроме того, можно включить алгоритм ЕС (Enable echo cancellation) - Включить эхо подавление.

Заключение.

Как я уже писал — это описание первого реализованного шага в направлении работы с голосом. И конечно нужна совместная отладка, определение следующих шагов, что бы сделать подобную работу с голосом практически нужной, либо удостовериться, что это не нужно (не может быть реально использовано в нынешней нашей архитектуре). В любом случае данное направление перспективно.