СИБРУС

Принципы развертывания системы

версия 1.2, 2018 г.

Оглавление

Введение

«СИБРУС» служит для построения системы внутрикорпоративной связи по каналам локальных вычислительных сетей и Интернет.

Отличительной особенностью является ориентированность на обеспечение высокой степени безопасности связи и хранения данных в системы «СИБРУС».

«СИБРУС» реализует следующие основные функции:

- Текстовая переписка между парами пользователей в реальном времени.
- Текстовая переписка в рамках групп пользователей в реальном времени.
- Обмен файлами между парами пользователей.
- Обмен файлами в рамках групп пользователей.
- Голосовая связь между парами пользователей.
- Видеосвязь между парами пользователей.
- Голосовые конференции в рамках групп пользователей.
- Видеоконференции в рамках групп пользователей.
- Долговременное защищенное хранение истории переписки и файлов как на клиентских устройствах пользователей, так и на сервере.

Помимо основных функций «СИБРУС» реализует ряд различных дополнительных функций, повышающих эффективность совместной работы, хранения и обмена информации.

1 Требования к среде окружения

1.1 Сервер

1.1.1. Операционные системы

Серверное ПО «СИБРУС» доступно для следующих операционных систем:

- Linux – основная рекомендованная ОС, рекомендуется использовать версии Debian GNU/Linux AMD64.
- Windows Server – портированная версия ПО «СИБРУС», не рекомендуется для крупных инсталляций.

1.1.2. Зависимости от стороннего ПО

Основное серверное ПО «СИБРУС»

Для работы серверного ПО «СИБРУС» используется следующее стороннее программное обеспечение:

- В качестве базы данных используется NoSQL БД MongoDB https://www.mongodb.com
- В качестве глобальной разделяемой памяти и очереди сообщений используется Redis https://redis.io .

Дополнительные необязательные зависимости

В случае приватной схемы управления обновлениями клиентского ПО потребуются какой-либо сторонний web-сервер, а также, возможно, система MDM (Mobile Device Management).

Для обеспечения функциональности звонков на телефоны по протоколам SIP/RTP потребуется сторонняя АТС (PBX). Протестирована совместимость «СИБРУС» с Asterisk PBX и 3CX PBX.

1.1.3. Аппаратные требования к узлам сервера

Принципы оценки аппаратных ресурсов кластера приведены в документе «СИБРУС — Конфигурация кластера».

1.1.4 Сетевые требования

Порты и настройки файерволов

Сервер «СИБРУС» по умолчанию использует следующие порты для связи:

- **Входящие**
  - TCP-порт 37210 для TLS-соединений клиент-сервер.
  - TCP-порт 37212 для TLS-соединений для автоматической конфигурации клиентов.
  - Диапазон UDP-портов 3000-9000 для аудио и видеосвязи и видеоконференций.
  - Также дополнительно может использоваться TCP-порт 443 для TLS-соединений клиент-сервер в случае, если из сети клиента разрешен доступ только по портам 80 и 443.
  - TCP-порт 443 для HTTPS-соединений со стороны клиентов для автоматического обновления клиентского ПО.
  - UDP-порт 5060 для SIP.
  - Настраиваемый диапазон UDP-портов для RTP.
- **Исходящие**
  - TCP-порт 443 (или другой, в зависимости от того, где push-шлюз) для HTTPS-соединений для отправки уведомлений на мобильные клиенты.
  - TCP-порт 389 для связи с AD/LDAP.
  - UDP-порт 5060 для SIP.
  - Настраиваемый диапазон UDP-портов для RTP.

Пропускная способность

Звонки точка-точка

	Клиентское приложение		Сервер (при звонке через релей)	
	Вх.	Исх.	Вх.	Исх.
Аудио	60-80 Кбит/сек	60-80 Кбит/сек	60-80 Кбит/сек	60-80 Кбит/сек
Видео	400-800 Кбит/сек	400-800 Кбит/сек	400-800 Кбит/сек	400-800 Кбит/сек

Конференции - M участников, из них N активных

Участник		Активный участник		Сервер	
Вх.	Исх.	Вх.	Исх.	Вх.	Исх.
аудиопоток	-	аудиопоток	аудиопоток	N*аудиопоток	M*аудиопоток
N*видеопоток	-	N*видеопоток	видеопоток	N*видеопоток	M*N*видеопоток

1.2 Клиентские устройства

1.2.1. Операционные системы

Клиентское ПО «СИБРУС» доступно для следующих операционных систем:

- Windows XP/Vista/7/8/10.
- MacOS версии 10.x и новее.
- Linux 32/64, протестированы Ubuntu, Debian, Mint, BaseAlt, AstraLinux.
- Android версии 4.0 и новее.
- iOS версии 8 и новее.

1.2.2. Дополнительные требования

Для корректной работы мобильных приложений на платформах iOS и Android необходимо использовать службы уведомлений:

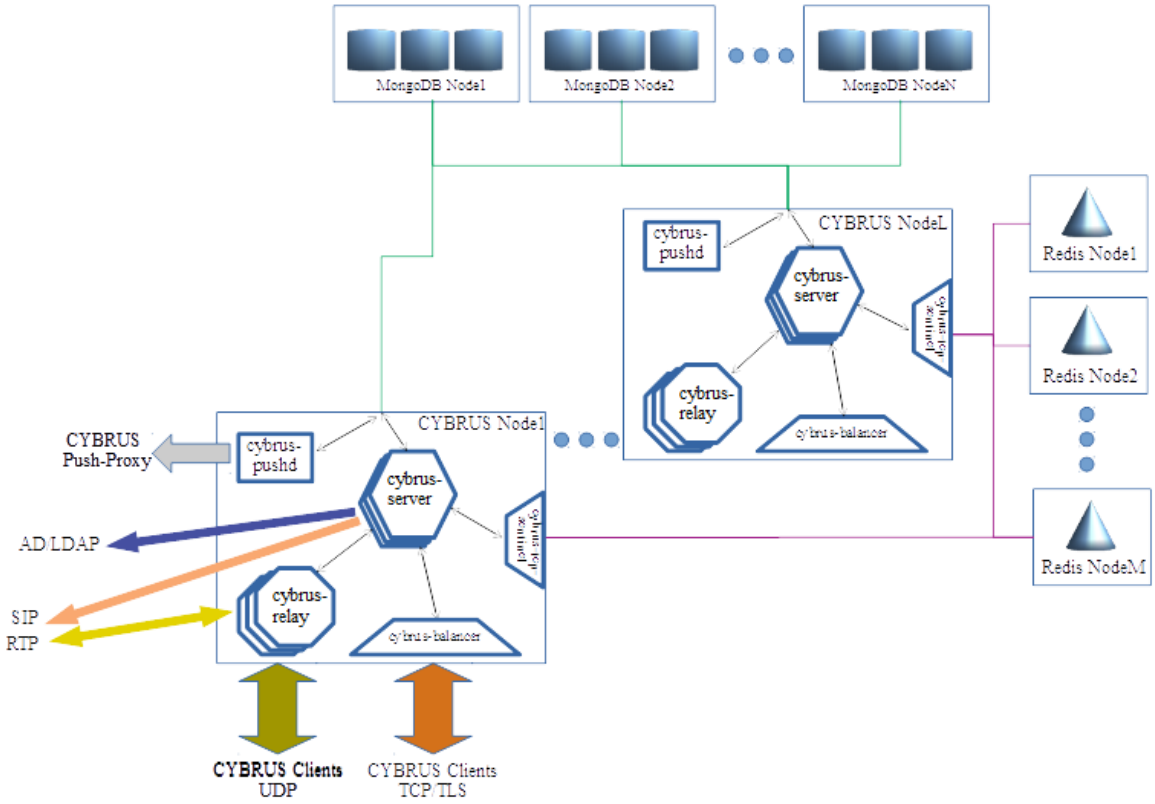
- iOS – Apple Push Notification System (APNS);
- Android – Firebase Cloud Messaging (FCM).

Для обеспечения доступа к данным службам:

- клиентские устройства должны иметь сетевой доступ к данным сервисам, предоставляемым производителями соответствующих платформ;
- серверы «СИБРУС» должны быть настроены на обращения к соответствующим сервисам одним из следующих способов:
  - либо через прокси-шлюзы Производителя ПО «СИБРУС»;
  - либо напрямую к сервисам Apple и Google, но в таком случае необходимо произвести предварительную подготовку клиентского ПО для мобильных платформ.

2 Архитектура серверной инсталляции

2.1 Кластер «СИБРУС»



2.1.1. Состав пакета серверного программного обеспечения

Состав пакета серверного программного обеспечения приводится для максимального набора компонент для кластера на Linux. Набор компонент для некластерных инсталляций, а также инсталляций Windows может отличаться от перечисленного ниже.

- Основной прикладной серверный процесс, приложение *cybrus-server*.
- Медиаретранслятор для аудио и видео связи, приложение *cybrus-relay*.
- Сервер внутренних видеоконференций, модуль в составе *cybrus-relay*.
- Шлюз в системы IP-телефонии по протоколам SIP/RTP:
  - сигнальный уровень SIP реализован как модуль в составе *cybrus-server;*

  - транспортный уровень RTP реализован как модуль в составе *cybrus-relay.*
- Балансировщик нагрузки на серверные процессы, приложение *cybrus-balancer*.
- Диспетчер кластеризации очереди сообщений и глобальной памяти, приложение *cybrus-tcp-sentinel*.
- Гипервизор серверных процессов, приложение *cybrus-launcher*.
- Утилита мониторинга серверных процессов, приложения *cybrus-ping* и *cybrus-test-sms*.
- Утилита администрирования сервера, приложение *cybrusadm*.
- Утилита администрирования базы данных сервера, приложение *cybrus-database-adm*.
- Утилита управления процессами сервера «на лету» (в runtime), приложение *cybrusctl*.
- Утилита сервиса нотификация мобильных платформ, приложение *cybrus-pushd*.
- Модуль интеграции с AD/LDAP, модуль в составе *cybrus-server.*
- Модули реализации API администрирования сервера для интеграции со сторонними системами — *экспериментальные, могут не входить в стандартную поставку*.
  - Утилита администрирования, приложение *cybrus-admin-cli*.
  - Библиотека PHP для администрирования сервера.
- Модули реализации API построения специализированных клиентов-шлюзов в сторонние протоколы — *экспериментальные, могут не входить в стандартную поставку*.
  - Базовый прокси между сервером «СИБРУС» и Redis Pub/Sub, приложение *cybrus-proxy-adapter*.

2.1.2. Взаимодействие компонентов серверного ПО

Основные функции сервера СИБРУС реализуются основными прикладными серверными модулями *cybrus-server*. В составе кластера может параллельно работать произвольное количество процессов *cybrus-server* на произвольном количестве физических узлов кластера. Число процессов ограничивается только условиями лицензионного соглашения и аппаратными ресурсами сервера.

Для управления аудио- и видеосвязью используются медиаретрансляторы *cybrus-relay*, которые также выполняют роль серверов внутренних видеоконференций (ВКС). В составе кластера может быть запущено произвольное количество медиаретрансляторов и серверов ВКС. Медиаретрансляторы и серверы ВКС могут быть физически распределены по разным географически разнесенным дата-центрам.

Число процессов ограничивается только условиями лицензионного соглашения и аппаратными ресурсами сервера.

Сервер позволяет осуществлять связь со сторонними системами IP-телефонии с использованием протоколов SIP/RTP через соответствующие шлюзы, входящие в серверный пакет.

Балансировка запросов клиентских приложений между прикладными серверными модулями *cybrus-server* может выполняться либо неявно за счет DNS-балансировки, либо явно с использованием балансировщиков *cybrus-balancer*.

Серверные модули могут либо напрямую обращаться к глобальному менеджеру разделяемой памяти и очереди сообщений в составе кластера, либо через диспетчер кластеризации *cybrus-tcp-sentinel*, который автоматически выбирает работающий менеджер разделяемой памяти и очереди сообщений среди доступного пула в кластере.

На каждом узле кластера может быть запущен гипервизор серверных процессов СИБРУС *cybrus-launcher*, в задачи которого входит запуск и останов серверных процессов СИБРУС, а также мониторинг их возможного падения и автоматический перезапуск.

Утилиты мониторинга серверных процессов *cybrus-ping* и *cybrus-test-sms* могут быть запущены как на узлах в составе кластера, так и вне кластера. Монитор периодически подключается по очереди к различным процессам кластера, оценивает доступность серверных процессов кластера и скорость отклика. Результаты работы мониторов могут быть собраны в систему мониторинга (например, Zabbix). В случае недоступности каких-либо процессов, монитор отправляет оповещения о проблемах системным администраторам.

Утилита администрирования *cybrusadm* используется для базового администрирования сервера (управление лицензиями, назначение администраторов, загрузка конфигурация и т.д.). Для управления конфигурацией базы данных используется утилита *cybrus-database-adm.* Для управления пользователями используется отдельный модуль в составе клиентского ПО СИБРУС.

Утилита управления процессами сервера *cybrusctl* используется для управления серверными процессами кластера «на лету» (например, для частичной перезагрузки параметров, останова и перезапуска отдельного серверного процесса и т. д.).

Утилита сервиса нотификации мобильных платформ *cybrus-pushd* используется для отправки уведомлений на мобильные клиентские приложения через сервисы уведомлений Google и Apple.

Модуль интеграции с AD/LDAP используется для режима управления доступами пользователей к системе через стороннюю систему с использованием протокола LDAP.

Модуль реализации API администрирования сервера позволяет встраивать управление сервером и пользователями в сторонние системы управления АСУ через специальное API.

Модуль реализации API построения специализированных клиентов-шлюзов в сторонние протоколы предназначен для возможности построения так называемых

«ботов», а также для возможности обмена различными объектами на уровне данных со сторонними системами. Например, можно построить боты, которые будут получать текстовые команды в формате переписки от пользователя через обычный клиент и транслировать эти команды в какие-то сторонние системы. Либо, в обратную сторону, из сторонних систем может передаваться какая-то информация пользователям в клиентские приложения в формате текстовой переписки.

2.2. Использование Active Directory и LDAP

Сервер «СИБРУС» поддерживает интеграцию со сторонними службами каталогов пользователей по протоколу LDAP. Самая распространенная служба каталогов — это Active Directory (AD) от Microsoft.

LDAP/AD используется в «СИБРУС» следующим способом.

1. Импорт пользователей в базу данных «СИБРУС». Сервер «СИБРУС» содержит свою базу пользователей. Пользователей можно заводить как напрямую в «СИБРУС» с помощью инструментов администрирования, так и импортировать существующую пользовательскую базу из служб AD/LDAP, используемых в организации. При импорте из AD/LDAP копируются ФИО пользователей, адрес эл. почты, должность и телефон, а также фотография пользователя, если она загружена в AD/LDAP. В дальнейшем, при изменении данных пользователей в AD/LDAP они могут быть заново синхронизированы в базе «СИБРУС».
2. Авторизация пользователей на сервере «СИБРУС». При включенной интеграции с AD/LDAP процедура авторизации пользователей на сервере «СИБРУС» делегируется в AD/LDAP. В базе данных «СИБРУС» не хранятся пароли пользователей, импортированных из AD/LDAP.
3. Навигация по пользователям и поиск пользователей в клиентском приложении. Пользователи «СИБРУС» могут осуществлять навигацию по каталогам пользователей AD/LDAP и искать других пользователей для того, чтобы можно было добавлять друг друга в списки контактов и осуществлять общение.

2.3. Подключение к службам уведомлений мобильных платформ

Архитектура современных мобильных платформ Android и iOS ориентирована на обеспечение максимально возможной экономии электропитания и прочих ресурсов мобильных устройств. Поэтому в экосистеме Android и iOS приложение не может быть постоянно запущено и подключено к удаленному серверу по сети. Операционная система по своему усмотрению может либо отключить приложение от сети, либо вообще закрыть приложение.

Для того, чтобы клиенты таких приложений как «СИБРУС» всегда оставались на связи и получали уведомления о входящих сообщениях, звонках и прочих событиях,

используется система уведомлений, поставляемая производителем мобильной платформы.

«СИБРУС» поддерживает две системы уведомлений:

1. для iOS – Apple Push Notifications Service (APNS);
2. для Android – Firebird Cloud Messaging (FCM).

Обобщенно служба уведомлений для мобильной платформы в «СИБРУС» работает следующим образом.

1. Клиентская часть
    1. Клиент «СИБРУС» получает от службы уведомлений мобильной платформы уникальный идентификатор уведомлений — токен.
    2. Клиент «СИБРУС» отправляет токен на сервер «СИБРУС».
2. Серверная часть
    1. Токен сохраняется на сервере «СИБРУС» на некоторый период времени. По умолчанию срок хранения токена — 7 дней.
    2. На сервере «СИБРУС» запущена служба отправки уведомлений пользователям, которая входит в состав серверного ПО «СИБРУС».
        1. При приходе сообщения пользователю производится рассылка уведомлений о сообщении по всем токенам пользователя.
        2. Уведомление только информирует о самом факте прихода сообщения, но не содержит каких-либо данных из тела сообщения.
        3. Уведомления отправляются по протоколу HTTPS на прокси-шлюз уведомлений «СИБРУС», а не напрямую на серверы Apple и Google.
3. Инфраструктурная часть.
        1. Прокси-шлюз уведомлений «СИБРУС» не входит в серверное ПО «СИБРУС», а является частью инфраструктуры Производителя ПО «СИБРУС».
        2. Прокси-шлюз уведомлений «СИБРУС» получает уведомления от всех серверных инсталляций «СИБРУС» всех Заказчиков ПО «СИБРУС» и транслирует их от своего имени на серверы Apple или Google.
        3. Для каждой серверной инсталляции «СИБРУС» выдаются свои собственные сертификаты и реквизиты доступа к Прокси-шлюзу уведомлений «СИБРУС».
4. В случае использования кастомизированных версий клиентского ПО на условиях White Label Заказчик может самостоятельно взаимодействовать с сервисами Apple и Google без использования Прокси-шлюза уведомлений «СИБРУС».

Таким образом, для корректной работы служб уведомлений мобильных платформ должны быть выполнены следующие требования.

1. Мобильные устройства должны иметь доступ к Интернет и, в частности, к облачным сервисам производителей мобильных платформ Apple и Google.
2. На сервере «СИБРУС» необходимо настроить и запустить службу отправки уведомлений «СИБРУС», предварительно получив от Производителя ПО «СИБРУС» реквизиты доступа к Прокси-шлюзу уведомлений «СИБРУС».
3. На сервере «СИБРУС» должны быть разрешены исходящие соединения по протоколу HTTPS к Прокси-шлюзу уведомлений «СИБРУС».

Обратите внимание, что сервисы Apple и Google могут располагаться за пределами Российской Федерации.

2.4. Подключение к IP-телефонии

Пользователи «СИБРУС» могут осуществлять голосовые звонки на сторонние телефоны и службы через шлюз SIP/RTP, входящий в состав серверного ПО «СИБРУС». В состав клиентского ПО «СИБРУС» входит номеронабиратель и соответствующая телефонная книга.

Требования для интеграции с системами IP-телефонии следующие.

1 Шлюз SIP/RTP должен подключаться к сторонней АТС (PBX). Протестирована совместимость со следующими PBX:

1.1 Asterisk;

1.2 3CX.

2 Поддерживаются следующие протоколы и стандарты:

2.1 SIP в качестве сигнального протокола;

2.2 RTP в качестве транспорта голосового трафика.

2.3 Голосовые кодеки PCMU/PCMA.

3 Сетевые требования:

3.1 Используется только протокол UDP как для SIP, так и для RTP. Соответствующим образом должны быть настроены файерволы для разрешения такого трафика.

3.2 Шлюз SIP/RTP и сторонняя PBX должны быть взаимно доступны по прямым IP-адресам. NAT не поддерживается.

4 Шлюз SIP/RTP распределен между серверными приложениями «СИБРУС» следующим образом:

4.1 сигнальный уровень SIP реализуется в рамках основных серверных процессов *cybrus-server*;

4.2 транспортный уровень RTP реализуется в рамках серверных процессов медиаретрансляторов *cybrus-relay*.

5 Каждый пользователь «СИБРУС» подключается к сторонней PBX через шлюз SIP/RTP под своей собственной учетной записью. Поэтому, необходимо:

5.1 На сторонней PBX должны быть заведены учетные записи пользователей, которым разрешено осуществлять исходящие звонки.

5.2 Параметры учетных записей пользователей PBX должны быть занесены в настройки соответствующих пользователей сервера «СИБРУС»:

5.2.1 либо через импорт соответствующих полей из AD/LDAP;

5.2.2 либо вручную через графическую утилиту администрирования «СИБРУС».

2.5. Особенности конфигурации сети для UDP-трафика

Для того, чтобы уменьшить голосовой и видео трафик через сервер «СИБРУС», а также уменьшить задержки между клиентами при голосовой и видеосвязи, по умолчанию клиентское ПО «СИБРУС» пытается установить прямой (p2p) канал связи между парой абонентов. Если удается установить такой канал, то голосовой и видео трафик при двухсторонней связи между двумя клиентскими устройствами будет проходить напрямую, не попадая на сервер.

В большинстве текущих конфигураций локальных сетей клиенты подключаются к сети Интернет не с прямыми IP-адресами, а через трансляцию адресов NAT. В таком случае если два клиента находятся в разных сетях, они не могут узнать IP-адреса друг друга. Для того, чтобы установить прямой канал связи между ними, требуется посредник с прямым IP-адресом, к которому могут обратиться оба клиента. В ПО «СИБРУС» таким посредником выступает медиаретранслятор *cybrus-relay*. Для того, чтобы в системе была возможна прямая аудио и видеосвязь между клиентскими приложениями необходимо, чтобы медиаретрансляторы располагались на серверах с прямыми IP-адресами.

В случае, если нет возможности расположить медиаретранслятор на сервере с прямым IP-адресом, для большинства клиентов, особенно мобильных, будет невозможно установить прямое соединение друг с другом. Однако, сама голосовая и видеосвязь при этом может быть доступна через медиаретранслятор в случае правильной настройки сети и медиаретранслятора в соответствии с разделами руководства администратора, в котором описывается ситуация «медиаретранслятор за NAT или в DMZ».

3. Конфигурирование клиентских приложений

3.1. Выбор и настройки сервера

Каждая серверная инсталляция обладает рядом уникальных конфигурационных параметров, которые должны быть переданы в клиентское ПО для того, чтобы клиенты могли подключаться и работать с данным экземпляром серверного ПО «СИБРУС». Например, как минимум, IP-адрес и/или домен сервера должны быть известны клиенту, чтобы он мог подключиться к данному серверу.

Клиентское ПО «СИБРУС» предлагает несколько вариантов конфигурирования клиента при первом запуске.

- Ручной ввод настроек сервера в клиенте. Пользователь вручную заносит параметры сервера в диалоге настроек в клиентском ПО.
- Загрузка в клиент файла с настройками сервера. Пользователь загружает в клиентское ПО конфигурационный файл, полученный ранее от администратора.
- Автоматическое конфигурирование клиента по сети. Пользователю известен только его полный логин и пароль, а клиент самостоятельно по сети получает все необходимые конфигурационные данные.

Самый простой способ первого старта, с точки зрения обычного пользователя, это — использование автоматического конфигурирования клиента. В этом случае пользователю достаточно указать в стартовом диалоге только свой полный логин вида user@example.com, где user – уникальный идентификатор пользователя, а example.com – уникальное имя домена серверной инсталляции. После чего клиент по значению домена найдет соответствующий ему сервер автоконфигурирования и загрузит с него конфигурационные параметры клиента.

Для включения режима автоконфигурирования необходимо:

1. Настроить и запустить на серверной инсталляции сервер автоконфигурирования «СИБРУС».
2. Опубликовать IP-адрес и порт сервера автоконфигурирования, соответствующего домену данной серверной инсталляции одним из возможных способов:
    1. Либо с использованием записи DNS SRV вида _tcp._cybrusc.example.com.
    2. Либо с использованием бесплатного сервиса от Производителя ПО «СИБРУС», передав Производителю параметры сервера автоконфигурирования и имя домена для публикации в данном сервисе.
3. Клиент «СИБРУС» при первом старте по имени домена пробует найти запись DNS SRV для данной доменной зоны. В случае, если запись DNS SRV не

найдена, клиент обращается к сервису Производителя ПО «СИБРУС». Если соответствующий домен не найден и там, то клиент выведет пользователю ошибку, и пользователю придется вручную указать параметры сервера в настройках клиента.

3.2. Особенности установки клиентского ПО на платформы персональных компьютеров

3.2.1. Пути установки и пути хранения данных

По умолчанию клиентское ПО при самостоятельной установке ставится в домашнюю директорию пользователя. При этом пользователи могут самостоятельно обновлять клиентское ПО. Однако, администратор может выбрать централизованную установку клиента для всех пользователей компьютера. В таком случае управлять обновлениями сможет только администратор.

На одном устройстве могут храниться данные различных учетных записей пользователей «СИБРУС». Пользователь может переключаться между учетными записями в клиенте. При первой загрузке учетной записи с сервера пользователь может выбрать папку, в которой должны размещаться данные этой учетной записи. По умолчанию данные размещаются в домашней директории пользователя. Отдельно пользователь может выбрать в настройках путь, куда сохранять принятые и отправленные файлы.

3.2.2. Виртуальные диски

Клиентское ПО на платформах Windows/macOS/Linux реализует функции виртуального диска. Виртуальный диск служит для защищенного хранения принятых и отправленных файлов. Для работы виртуального диска необходимо учесть следующее.

1 Windows

1.1 Для работы виртуального диска используется компонент Dokany (https://github.com/dokan-dev/dokany). Инсталлятор драйвера Dokany входит в состав пакета «СИБРУС», однако иногда политики безопасности Windows могут быть таковы, что потребуется ручная установка драйвера администратором.

1.2 По умолчанию при подключении виртуального диска в системе появляется диск с некоторой буквой, например Z:\, который доступен для всех пользователей компьютера. В случае, если к компьютеру одновременно может быть подключено несколько пользователей, например, на сервере терминалов, то рекомендуется при установке клиента выбрать опцию «Монтировать виртуальный диск в домашние папки пользователей». В таком

случае другие пользователи не получат доступ к данным виртуальных дисков друг друга.

2 MacOS

2.1 Для работы виртуального диска используется компонент OSXFuse (https://osxfuse.github.io). Инсталлятор OSXFuse входит в состав пакета «СИБРУС». Однако, при некоторых комбинациях настроек безопасности может потребоваться, чтобы данный компонент был установлен вручную администратором.

3 Linux

3.1 Для виртуальных дисков используется стандартный компонент FUSE. Необходимо, чтобы ядро Linux было собрано с поддержкой FUSE.

4. Обеспечение отказоустойчивости

4.1. Отказоустойчивость в рамках одного дата-центра

«Горячая» отказоустойчивость в рамках одного дата-центра обеспечивается в кластерной конфигурации серверной инсталляции ПО «СИБРУС». В такой конфигурации при выходе из строя отдельных компонентов либо вообще не произойдет паузы в обслуживании, и выход из строя компонента приведет лишь к частичной деградации производительности всей системы, либо система автоматически восстановится после сбоя через небольшой период паузы в обслуживании.

1. Отказоустойчивость по базе данных обеспечивается за счет использования встроенных механизмов репликации базы данных MongoDB.
2. Отказоустойчивость по глобальной памяти и очереди сообщений Redis обеспечивается за счет использования утилиты «часового», который мониторит доступность серверов Redis и, в случае сбоя основного сервера Redis, переключается на один из резервных.
3. Отказоустойчивость по доступности серверных процессов ПО «СИБРУС» обеспечивается за счет нескольких механизмов:
    1. В рамках одной операционной системы может быть запущено несколько процессов ПО «СИБРУС». В случае аварийного завершения какого-либо процесса, он будет автоматически перезапущен супервизором процессов. В качестве супервизора процессов может использоваться либо супервизор, входящий в серверное ПО «СИБРУС», либо произвольный сторонний супервизор, например, systemd в Linux.
    2. Процессы ПО «СИБРУС» могут быть параллельно запущены на произвольном количестве аппаратных или виртуальных серверов. Число процессов ограничено только условиями лицензии и аппаратными ресурсами системы. При выходе из строя любого сервера все его задачи автоматически перераспределятся между оставшимися серверами.
    3. Для целей отказоустойчивости и балансировки нагрузки между серверными процессами «СИБРУС» используется два механизма.
        1. В рамках одного внешнего IP-адреса и TCP-порта балансировка осуществляется на прикладном уровне с использованием утилиты балансировщика, входящего в состав серверного ПО «СИБРУС».
        2. В рамках нескольких внешних IP-адресов и портов балансировка может также осуществляться за счет DNS-балансировки, при которой каждый клиент «СИБРУС» подключается к одному из IP-адресов для данного домена, выбирая его случайным образом.

4.2. Отказоустойчивость в рамках нескольких дата-центров

Отказоустойчивость в рамках нескольких дата-центров обеспечивает толерантность к риску выхода из строя всего основного дата-центра, на котором работает основная серверная инсталляция ПО «СИБРУС».

Восстановление системы после выхода из строя основного дата-центра не является автоматическим и требует ручного «холодного» рестарта системы в резервном дата-центре.

Для обеспечения отказоустойчивости системы в рамках нескольких дата-центров необходимо выполнить следующие предварительные действия:

1. Зарезервировать аппаратные ресурсы в резервном дата-центре, которые удовлетворяли бы тем же самым требованиям по производительности, что и в основном дата-центре.
2. Провести установку и настройку ПО «СИБРУС» и соответствующих зависимостей в резервном дата-центре.
3. Настроить репликацию базы данных MongoDB с основной системы в базу данных MongoDB в резервном дата-центре. Репликация должна работать в режиме скрытой теневой реплики с постоянным фоновым копированием данных по сети. Для этого между необходимо обеспечить достаточную производительности сетевого канала между дата-центрами.
4. Подготовить резервную настройку DNS-зоны домена сервера «СИБРУС», чтобы в случае переключения на резервный дата-центр, можно было бы достаточно быстро переключить DNS-настройки на адреса резервных серверов.

В случае переключения на резервный дата-центр для того, чтобы произвести обратное переключение при восстановлении основного дата-центра, необходимо провести ту же самую обратную процедуру так, как если бы основной дата-центр был резервным, предварительно выполнив обратную репликацию данных MongoDB.

4.3. Мониторинг

Мониторинг работоспособности инсталляции «СИБРУС» может осуществляться на трех уровнях.

1. Уровень доступности системы. На данном уровне мониторятся параметры доступности самой системы, например, откликается ли операционная система по сети на команды *ping*, какое состояние ОЗУ, дисковой систему, ЦПУ и т. п. Данный мониторинг осуществляется сторонними средствами по отношению к ПО «СИБРУС».
2. Уровень доступности процессов серверного ПО «СИБРУС». На данном уровне мониторится доступность серверных процессов ПО «СИБРУС» по сети. Для

осуществления данного вида мониторинга в состав серверного ПО «СИБРУС» входят консольные утилиты мониторинга.

3. Уровень статистики параметров работы серверного ПО «СИБРУС». На данном уровне мониторятся различные параметры, показывающие картину использования ПО «СИБРУС» во времени. Для осуществления данного вида мониторинга используется внутренний сбор статистики работы серверными процессами ПО «СИБРУС» и опрос статистики консольной утилитой, входящей в состав серверного ПО «СИБРУС».

Метрики мониторинга, собираемые на каждом из уровней могут быть переданы в стороннюю систему мониторинга. Производитель ПО «СИБРУС» может предоставить типовые настройки агентов мониторинга для системы мониторинга Zabbix (://www.zabbix.com).

5. Лицензирование

5.1. Схема лицензирования

Программное обеспечение «СИБРУС» лицензируется по следующей схеме.

1. На платной основе лицензируется только серверное ПО «СИБРУС». Возможны два варианта лицензирования:
    1. Ежегодная лицензия с ограничением времени использования, включающая постоянные бесплатные обновления серверного ПО в течение срока действия лицензии. Лицензию необходимо продлевать на ежегодной основе.
    2. Постоянная лицензия без ограничения времени использования. В лицензию включены бесплатные обновления ПО в течение года. Обновления ПО по истечении первого года предоставляются со скидкой на платной основе.
2. Базовый расчет стоимости лицензии осуществляется исходя из числа активных пользовательских учетных записей. Активная пользовательская учетная запись — это учетная запись пользователя, который не заблокирован в системе. Базовая стоимость лицензии определяется как «стоимость лицензии 1 учетной записи для данной конфигурации сервера», помноженная на число учетных записей.
2. Клиентское ПО «СИБРУС» под маркой «СИБРУС» распространяется бесплатно без ограничения количества копий.
3. В случае, если разрабатывается и выпускается клиентское ПО «СИБРУС» под сторонней маркой, то использование и распространение такого ПО лицензируется отдельным соглашением.

5.2. Процедура лицензирования

Лицензия серверного ПО «СИБРУС» привязывается к экземпляру базы данных без какой-либо привязки к оборудованию. Таким образом, в процессе эксплуатации сервер «СИБРУС» может полностью или частично переноситься с одного оборудования на другое без необходимости обновления лицензии.

В процессе первичной инсталляции серверного ПО «СИБРУС» создается запрос на лицензию сервера, который однозначно идентифицирует уникальный экземпляр серверной инсталляции «СИБРУС». В дальнейшем, на основе этого запроса производителем ПО выдаются все последующие лицензионные файлы в соответствии со спецификациями на конфигурацию серверного ПО и число активных пользовательских учетных записей, являющимися приложениями к Лицензионному

соглашению. Производитель ПО создает лицензионный файл и передает его Заказчику или Интегратору, обслуживающему информационные системы Заказчика. Заказчик или Интегратор должен активировать данный лицензионный файл в соответствии с инструкциями.

Если Заказчику в процессе эксплуатации требуется расширить лицензию, например, за счет добавления дополнительных активных пользовательских учетных записей, то Заказчик отправляет письменный запрос на приобретение дополнительных лицензий. Производитель ПО создает дополнительный лицензионный файл и передает его Заказчику или Интегратору, обслуживающему информационные системы Заказчика. Заказчик или Интегратор должен активировать данный лицензионный файл в соответствии с инструкциями.

6. Управление дистрибуцией и обновлениями

6.1. Серверное ПО

Дистрибуция и обновление серверного ПО выполняется в ручном режиме. Производитель серверного ПО передает Заказчику или Интегратору, обслуживающему информационные системы Заказчика, экземпляр пакета серверного ПО для выполнения инсталляции или обновления ПО на оборудовании Заказчика. Инсталляция и обновление ПО осуществляется в соответствии с инструкциями администратора для данной версии серверного ПО.

Производитель уведомляет Заказчика по электронным каналам связи о выходе новых версий серверного ПО «СИБРУС».

6.2. Клиентское ПО

Существует несколько вариантов дистрибуции и обновления клиентского ПО «СИБРУС».

1. Публичная дистрибуция и обновления. Дистрибуцией и обновлением управляет Производитель ПО.
2. Приватная дистрибуция и обновления. Дистрибуцией и обновлением управляет Заказчик.
3. Дистрибуция и обновление кастомизированных клиентских приложений White Label. Дистрибуцией и обновлением управляет Заказчик.

6.2.1. Публичная дистрибуция и обновления

Дистрибуция и управление обновлениями клиентского ПО «СИБРУС» осуществляются силами Производителя ПО «СИБРУС».

1 Клиенты для персональных компьютеров на платформах Windows/macOS/Linux

1.1 Самостоятельная установка и обновление ПО пользователями.

1.1.1 Инсталляторы доступны для самостоятельного скачивания пользователями с веб-сайта Производителя ПО www.cybrus.ru

1.1.2 Клиентское ПО периодически обращается к веб-сайту Производителz www.cybrus.ru для проверки наличия обновлений. В случае наличия обновлений пользователю предлагается обновить клиентское ПО. При согласии пользователя обновление автоматически загружается с веб-сайта производителя, после чего производится автоматическое обновление и перезапуск клиентского ПО.

1.2 Централизованная инсталляция и обновление клиентского ПО на компьютерах пользователей.

1.2.1 Администратор скачивает инсталляторы с веб-сайта Производителя ПО www.cybrus.ru

1.2.2 Администратор доступным ему способом выполняет установку клиентского ПО на компьютеры пользователей. Например, производит установку пакета cybrus.msi внутри домена Active Directory Windows через GPO.

2 Клиенты для мобильных платформ iOS/Android.

2.1 Инсталляторы доступны для самостоятельного скачивания пользователями из магазинов приложений производителей мобильных платформ.

2.1.1.1 AppStore для iOS.

2.1.1.2 Google Play для Android.

2.1.2 Производитель ПО «СИБРУС» выкладывает обновления в соответствующие магазины приложений.

2.1.3 Обновление ПО «СИБРУС» осуществляется либо автоматически, либо вручную пользователем, в зависимости от настроек мобильного устройства пользователя.

6.2.2. Приватная дистрибуция и обновления

Дистрибуция и управление обновлениями клиентского ПО «СИБРУС» осуществляются силами Заказчика:

1 Производитель ПО «СИБРУС» передает Заказчику или Интегратору, обслуживающему Информационные системы Заказчика, пакеты инсталляторов и обновлений клиентского ПО «СИБРУС» для всех поддерживаемых платформ.

2 Дистрибуция и обновление клиентского ПО «СИБРУС» осуществляется с внутренних серверов Заказчика.

2.1 Клиенты для персональных компьютеров на платформах Windows/macOS/Linux.

2.1.1 Самостоятельная установка и обновление ПО пользователями.

2.1.1.1 Пакеты инсталляторов и обновлений размещаются на web-сервере Заказчика.

2.1.1.2 Клиентское ПО автоматически конфигурируется таким образом, чтобы осуществлять проверку обновлений с веб-сайта Заказчика.

2.1.1.3 Инсталляторы доступны для самостоятельного скачивания пользователями с веб-сайта Заказчика.

2.1.1.4 Клиентское ПО периодически обращается к веб-сайту Заказчика для проверки наличия обновлений. В случае наличия обновлений

пользователю предлагается обновить клиентское ПО. При согласии пользователя обновление автоматически загружается с веб-сайта Заказчика, после чего производится автоматическое обновление и перезапуск клиентского ПО.

2.1.2 Централизованная инсталляция и обновление клиентского ПО на компьютерах пользователей.

2.1.2.1 Администратор доступным ему способом выполняет установку клиентского ПО на компьютерах пользователей. Например, производит установку пакета cybrus.msi внутри домена Active Directory Windows через GPO.

3 Клиенты для мобильных платформ iOS/Android.

3.1 Клиентское ПО размещается в MDM Заказчика или на внутреннем веб-сервере Заказчика.

3.2 Устройства пользователей автоматически устанавливают или обновляют клиентское ПО «СИБРУС» из MDM, в соответствии с политиками управления приложениями. Либо пользователи в ручном режиме скачивают обновления с внутреннего веб-сервера Заказчика.

6.2.3. Кастомизированное клиентское ПО на условиях White Label

Клиентское ПО может быть кастомизировано и выпущено на условиях White Label с именем бренда и стилем Заказчика.

Дистрибуция и обновление такого клиентского ПО совпадает со схемой приватной дистрибуции и обновлений с некоторыми уточнениями для мобильных платформ.

1. Для выпуска кастомизированного клиента на iOS Заказчик или другое лицо, от имени которого будет выпускаться приложение, должен заключить с Apple соглашение о вступлении в Apple Enterprise Developer Program.
2. Кастомизированные клиентские приложения на iOS не могут публиковаться в Apple AppStore для массового скачивания и использования.
3. Для публикации кастомизированного клиентского приложения на Android Заказчик или другое лицо, от имени которого будет выпускаться приложение, должен зарегистрировать соответствующий Google Account, от имени которого будет публиковаться приложение.

7. Администрирование

7.1. Конфигурирование сервера

Конфигурирование серверных процессов осуществляется с использованием текстовых конфигурационных файлов, параметров командной строки, а также консольных утилит администрирования.

У каждого приложения в серверном пакете есть набор конфигурационных параметров, которые могут быть переданы приложению либо в виде параметров командной строки, либо в составе конфигурационного файла. Для просмотра доступных параметров того или иного приложения необходимо запустить данное приложение с ключом —help или -h, например, «cybrus-server –help». Для просмотра версии приложения необходимо запустить приложение с ключом —version или -v, например, «cybrus-server -v».

7.2. Журналы работы

На сервере «СИБРУС» ведется два вида журналов.

1. Технические журналы работы серверных процессов. Каждый серверный процесс выводит логи своей работы в файл логов. Можно также направить вывод логов в консоль, либо в syslog.
2. Журналы действий в системе.
    1. Данные журналы сохраняются в базе данных сервера и доступны для просмотра через графическую утилиту администрирования.
    2. Журналы действий могут включать в себя следующие журналы:
        1. Журнал подключений пользователей к серверу.
        2. Журнал подключений и действий администраторов.
        3. Журнал операций с параметрами безопасности пользователей.

7.3. Администраторы

В «СИБРУС» выделяется два уровня функций администрирования.

1. Системные администраторы.
2. Администраторы безопасности.

7.3.1. Системные администраторы

Системный администратор выполняет задачи по обеспечению работоспособности сервера. Системный администратор имеет доступ к аппаратными ресурсам и базе

данных сервера. В задачи системного администратора входит следующий список, который может быть расширен:

1. Развертывание и настройка аппаратных ресурсов сервера.
2. Установка и настройка системного программного обеспечения и среды окружения.
3. Настройка сети.
4. Установка и обновления серверного ПО «СИБРУС».
5. Управление лицензиями серверного ПО «СИБРУС».
6. Настройка интеграции ПО «СИБРУС» со сторонними системами, например, с AD/LDAP.
7. Заведение первых пользователей.
8. Создание Доменов Безопасности.
9. Назначение Администраторов безопасности.
10. Мониторинг сервера.
11. Управление резервированием и обеспечением отказоустойчивости.
12. Просмотр и анализ журналов сервера.

Помимо различных системных инструментов для администрирования системный администратор использует специальные консольные утилиты в составе серверного ПО «СИБРУС», а также графический модуль администрирования, который входит в состав клиентского ПО «СИБРУС» для версий на персональных компьютерах Windows/macOS/Linux.

7.3.2. Администраторы безопасности

Пользователи «СИБРУС» могут входить в Домены Безопасности. Домен Безопасности — это специальный вид группы пользователей, управление настройками безопасности, которых может осуществляться централизованно администратором данной группы. Администратор безопасности не имеет прямого доступа к аппаратным ресурсам и базе данных сервера. Администратор безопасности может управлять только настройками безопасности тех пользователей, которые входят в Домен Безопасности, которым управляет данный администратор.

В задачи администратора безопасности входит следующий список, который может быть расширен:

1. Добавление новых пользователей в Домен Безопасности.
2. Блокировка пользователей в Домене Безопасности.
3. Просмотр и анализ журнала логов подключений и действий пользователей в Домене Безопасности.
4. Управление списками контактов пользователей в Домене Безопасности.
5. Управление параметрами безопасности пользователей в Домене Безопасности.

6. Управление ключами шифрования пользователей в Домене Безопасности.
7. Просмотр истории переписки пользователей в Домене Безопасности.

Администратор безопасности для работы использует специальный графический модуль администрирования, который входит в состав клиентского ПО «СИБРУС» для версий на персональных компьютерах Windows/macOS/Linux.