Infomaximum

«Инфомаксимум»

(Общество с ограниченной ответственностью)

Система «Proceset»

Руководство прикладного администратора

2026 г.

Оглавление

Администратор

Раздел предназначен для специалистов, которые устанавливают и настраивают систему Proceset.

В разделе приведены инструкции по развертыванию серверной части, установке и подключению СУБД ClickHouse, управлению агентами и интеграциями, а также по работе с пользователями, ролевой моделью и конфигурацией системы.

Управление сервером Proceset

Раздел содержит инструкции по установке, обновлению и удалению серверной части Proceset, настройке кластерного режима и переносу сервера с Windows на Linux.

Инструкции по установке, обновлению и удалению системы

Приложение Infomaximum — программное решение для сбора, обработки и визуализации данных в реальном времени. Для того, чтобы начать работу с приложением, ознакомьтесь с инструкциями ниже.

- Установка, настройка и удаление приложения Infomaximum на Windows
- Установка приложения Infomaximum на Linux
- Обновление Proceset
- Лицензионный ключ

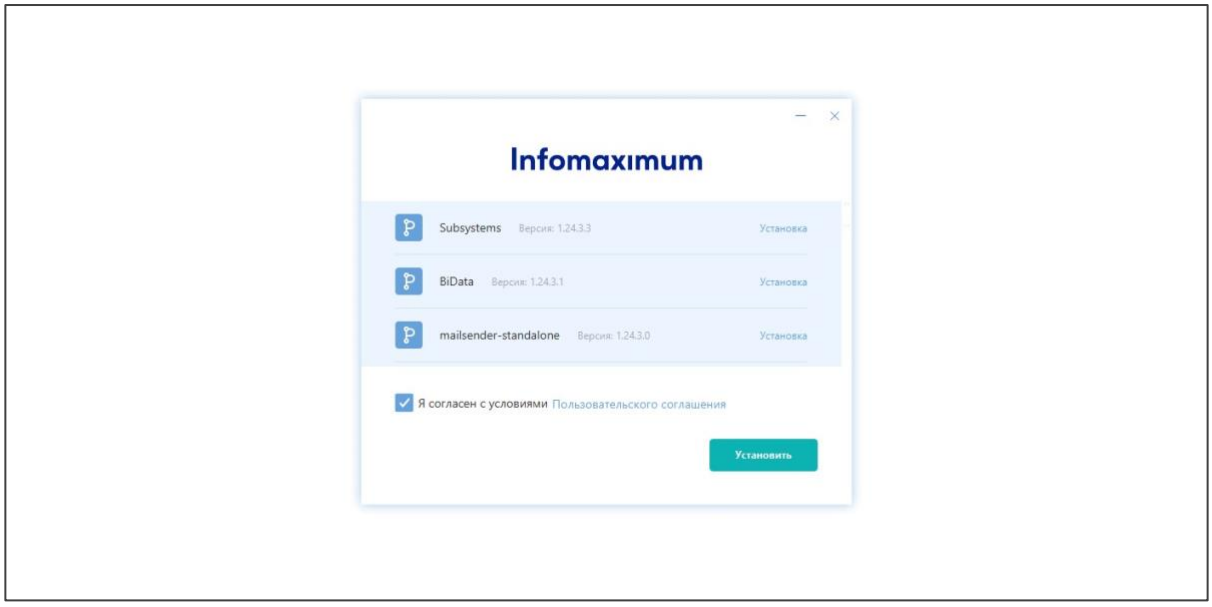
Установка, настройка и удаление приложения Infomaximum на Windows

Установка приложения Infomaximum на Windows

Для установки приложения Infomaximum может быть использован сервер под управлением ОС Windows Server 2012R2 (x64) или выше.

Выполните следующие шаги:

1. Запустите переданный вам exe-файл на сервере с правами локального администратора. Откроется окно установщика приложения.



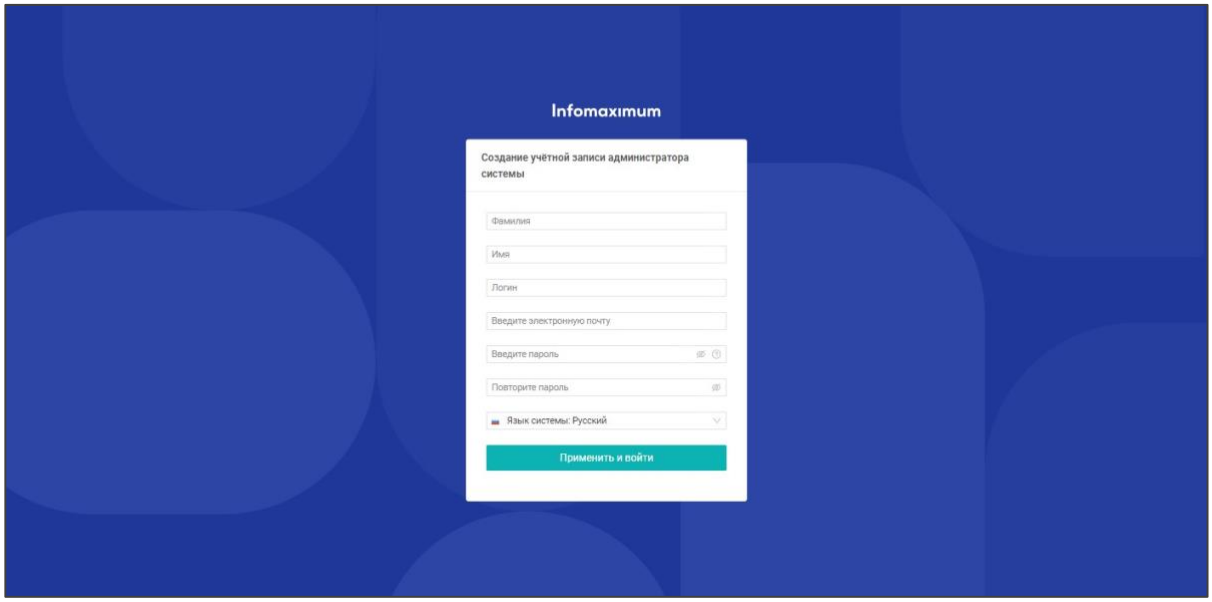
2. Ознакомьтесь с условиями пользовательского соглашения и примите их, поставив галочку в строке *Я согласен с Пользовательским соглашением*.

3. Нажмите **Установить**.

4. В процессе установки может появиться запрос на создание разрешающего правила для «Брандмауэра Windows». Разрешите его создание.

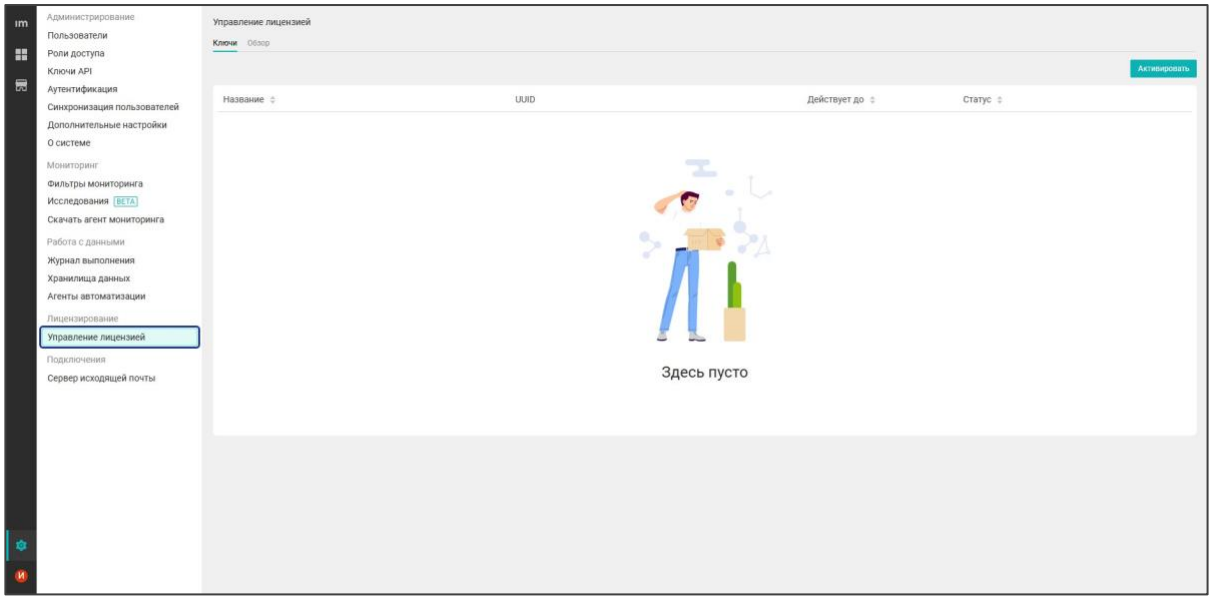
5. После завершения процесса установки нажмите **Готово**.

6. В браузере, используемом по умолчанию, откроется страница для создания учетной записи администратора системы.



7. Заполните поля на странице и нажмите **Применить и войти**. Указанные данные будут использоваться для входа первого администратора системы в веб-интерфейс.

8. После успешного ввода данных откроется веб-интерфейс системы. Перейдите в *Настройки* и нажмите **Управление лицензией**.



9. Активируйте лицензионный ключ системы.

10. После успешной активации приложение готово к дальнейшей настройке.

Настройка работы веб-сервера приложения по защищенному протоколу HTTPS

После установки приложения Infomaximum работа с веб-интерфейсом и передача данных от агентов мониторинга происходит по протоколу HTTP. Чтобы настроить передачу данных по защищенному протоколу HTTPS, выполните следующие действия:

1. Подготовьте SSL-сертификат и закрытый ключ для сервера, на котором установлено приложение, в виде pfx-файла (PKCS#12). Допустимо, чтобы в pfx-файл были также включены

сертификаты центров сертификации. Важно, чтобы этот сертификат сервера был доверенным на компьютерах, где будут установлены агенты мониторинга и где будет происходить работа с веб-интерфейсом. Для получения сертификата рекомендуем обратиться к администраторам инфраструктуры открытых ключей (PKI) в вашей организации.

2. На сервере отключите службу Infomaximum: нажмите *Win+R*, откройте *services.msc*, в открывшемся окне кликните правой кнопкой мыши по **Infomaximum** и выберите **Остановить**.

3. Запустите текстовый редактор, например, «Блокнот», от имени администратора и отредактируйте файл конфигурации веб-сервера приложения C:\ProgramData\Infomaximum\config\com.infomaximum.subsystem.frontend.json.

4. Рекомендуется сделать резервную копию файла перед редактированием.

5. Внесите в файл следующие изменения:

- "protocol":"https" — использование HTTPS
- "port":443 — укажите порт, на котором будет работать веб-сервер
- "ssl_cert_store_password":"password" — пароль от pfx-файла. Если пароль пустой, укажите ""
- "ssl_cert_store":"C:\\ProgramData\\Infomaximum\\certs\\file.pfx" — путь к pfx-файлу. Косую черту необходимо экранировать еще одной косой чертой
- "url" — используется системой для формирования всех входящих ссылок. Например, на основе параметра создаются ссылки в рассылаемых почтовых сообщениях для сброса пароля и приглашения

6. Сохраните файл.

7. Запустите службу Infomaximum.

8. Проверьте доступность веб-сервера по протоколу HTTPS по указанному порту. Если после запуска возникают затруднения, обратитесь к логу C:\ProgramData\Infomaximum\logs\main.log.

Примечание. Агенты мониторинга также передают данные в приложение через веб-сервер. При изменении параметров веб-сервера необходимо изменить настройки ранее установленных агентов или выполнить переустановку агентов.

Указание доступного для приложения Infomaximum объема оперативной памяти

Максимальный объем оперативной памяти сервера (ОЗУ), который может использовать приложение Infomaximum, строго определен в параметрах запуска службы.

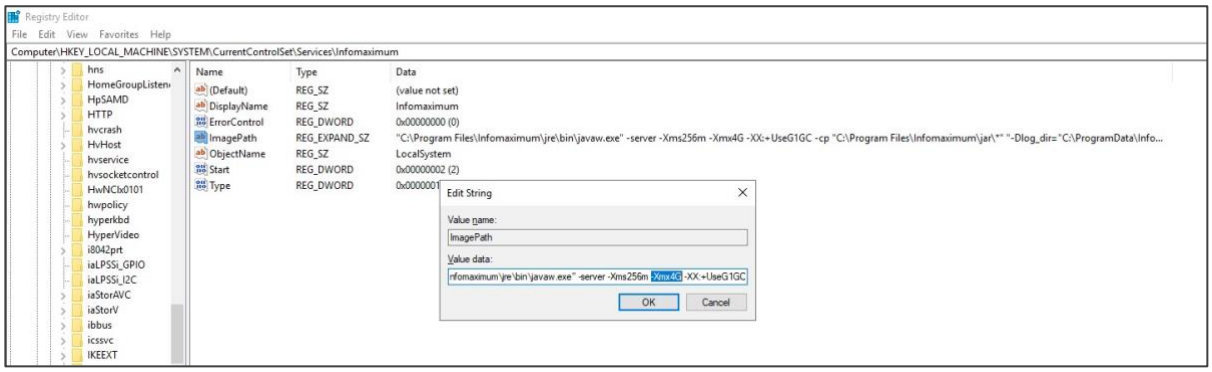
Чтобы изменить выделенный объем ОЗУ для службы Infomaximum выполните следующие шаги:

1. Откройте *Редактор реестра* с правами администратора.

2. В редакторе реестра откройте ветку Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Infomaximum.

3. Дважды кликните по *ImagePath*.

4. В поле **Value Data** измените параметр -Xmx2G p на нужное значение в гигабайтах, например, на Xmx8G, чтобы выделить 8 Гб памяти.



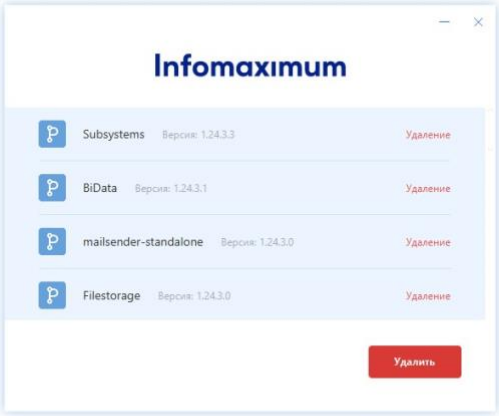
5. Сохраните изменения и перезапустите службу Infomaximum в оснастке службы или *Диспетчере задач*.

Совет. Выделяйте не более 80 % от общего объема оперативной памяти сервера. Также обратите внимание, что необходимо оставить 4 Гб свободной ОЗУ для стабильной работы операционной системы.

Удаление приложения Infomaximum

Для удаления приложения Infomaximum запустите файл *uninstall.exe* на сервере, где оно установлено (по умолчанию файл расположен в *C:\Program Files\Infomaximum\uninstall.exe*).

Откроется окно с предложением удалить программу.



Нажмите **Удалить**.

Логи системы и инсталлятора

Логи установки и службы необходимы для диагностики ошибок и мониторинга приложения в реальном времени. Их доступность помогает быстро выявлять и устранять нарушения в работе.

- Лог работы службы Infomaximum по умолчанию сохраняется в *«C:\ProgramData\Infomaximum\logs\main.log»*
- Лог процесса установки сохраняется в каталоге %TMP% пользователя, в сеансе которого был запущен процесс установки/обновления. По умолчанию хранится в *«C:\Users%USERNAME%AppData\Local\Temp\infomaximum_stdout.log»*

Установка приложения Infomaximum на Linux

Развертывание Proceset на Linux с использованием Docker обеспечивает надежность, отказоустойчивость и простоту управления.

Чтобы установить приложение Infomaximum на Linux выполните следующие действия:

1. Установите на сервер ПО Docker, следуя официальной документации *https://docs.docker.com/engine/install/.*

2. Загрузите предоставленный архив с Docker-образом на нужный сервер.

3. Распакуйте архив и загрузите его в локальное хранилище образов Docker.

```
$ gunzip infomaximum_docker_app_d230809.tar.gz
```

```
# docker load < infomaximum_docker_app_d230809.tar
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

4. Создайте тома Docker (volume) для постоянного хранения данных работы программы.

```
# docker volume create infomaximum-app-data
```

```
# docker volume create infomaximum-app-log
```

Примечание. Необходим сервер с возможностью установки ПО Docker версии 17.06 или выше.

Запуск приложения Infomaximum с HTTP

После создания томов приложение можно запустить в режиме HTTP — этот способ подходит для тестовых сред, внутренних серверов или случаев, когда шифрование трафика не требуется.

Чтобы запустить контейнер с приложением, которое будет работать по HTTP, выполните следующую команду:

```
# docker run -d --name infomaximum-app \
--mount source=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount source=infomaximum-app-log,target=/var/log/infomaximum/ \
-p 0.0.0.0:8010:8010 -d --restart=always \
dockerhub.office.infomaximum.com/infomaximum/infomaximum_app:d230809
```

Примечание. Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

Запуск приложения Infomaximum с HTTP (Swarm)

Для промышленной эксплуатации требуется отказоустойчивая и масштабируемая среда. Используйте Docker Swarm для запуска Infomaximum как службы с возможностью управления несколькими узлами и распределения нагрузки.

Чтобы запустить с помощью Swarm контейнер с приложением, которое будет работать по HTTP, выполните следующие действия:

1. Запустите локальный Docker Swarm.

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

2. Создайте службу (service).

```
# docker service create --name infomaximum-app \
--mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
-e JVM_MAX_MEMORY='4G' \
--publish published=8010,target=8010,mode=host \
--limit-memory 30G \
--limit-cpu 2 \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
dockerhub.office.infomaximum.com/infomaximum/infomaximum_app:d230809
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

Запуск приложения Infomaximum с HTTPS (Swarm)

При работе в корпоративной среде необходимо обеспечить безопасную передачу данных. Для этого запустите приложение в режиме HTTPS с использованием Docker Swarm и механизма secrets для хранения сертификатов и паролей.

Чтобы запустить с помощью Swarm контейнер с приложением, которое будет работать по HTTPS, выполните следующие действия:

1. Запустите Docker Swarm.

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

2. Создайте секреты (secrets).

```
# docker secret create infomaximum_app_https_certificate ${PATH_FILE}
```

```
# echo -n "${PASSWORD_CERTIFICATE}" | docker secret create infomaximum_app_https_certificate_password -
```

где:

- ${PATH_FILE} — полный путь к файлу с сертификатом и приватным ключом
- ${PASSWORD_CERTIFICATE} — пароль к файлу с сертификатом

3. Создайте службу (service).

```
# docker service create --name infomaximum-app \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
--publish published=443,target=8010,mode=host \
--restart-max-attempts 5 \
```

```
--restart-condition "on-failure" \
-e JVM_MAX_MEMORY='4G' \
-e FE_URL="https://server-name.domain.com" \
-e FE_CORS_POLICY="*" \
infomaximum/infomaximum_app:d230809
```

Совет.

- Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.
- Рекомендуется указать порт для веб-интерфейса стандартный для HTTPS — 443. При необходимости можно указать другой порт.

4. В переменной JVM_MAX_MEMORY укажите максимальный объем ОЗУ в гигабайтах, который сможет использовать приложение Infomaximum. Рекомендуется указать значение в пределах 60-90 % от свободного объема ОЗУ на сервере. В переменной FE_URL укажите рабочий URL, который будет использоваться для входа в веб-интерфейс. В переменной FE_CORS_POLICY укажите значение *, чтобы настроить CORS для междоменного обмена данными с любыми сайтами. Для ограничения доступа перечислите через запятую список URL-адресов, с которыми необходим междоменный обмен данными, например, "https://infomaximum.com, https://infomaximum.ru".

5. Приложение запущено. В веб-браузере откройте страницу https://<fqdn-сервера>/ для создания учетной записи администратора системы.

Если страница не открывается:

- Проверьте, что контейнер работает, выполнив команду:

```
# docker ps
```

- Обратитесь к логам приложения, выполнив команду:

```
# docker service logs -f infomaximum-app
```

- Убедитесь, что с компьютера, где открыт браузер, есть возможность соединения с сервером Proceset, например при помощи telnet.

Обновление Proceset

Каждое обновление ПО Proceset содержит определенный набор изменений функционала системы, запланированный к реализации. Изменения описаны в списке изменений и документе «Change List», который передается вместе с обновлением.

Название версии включает в себя префикс, обозначающий семейство ОС, для которой предназначено обновление: w — Windows, d — Linux (Docker). Также в названии указывается год, порядковый номер версии в рамках года, порядковый номер сборки. Например, версия с названием w221201 — это обновление системы на Windows, двенадцатый релиз в 2022 году, первая сборка текущей версии.

Обновление ПО Proceset должно производиться последовательно по релизам. Например, если нужно обновить систему с версии w2211xx до версии w2301xx, необходимо последовательно устанавливать обновления w2212xx и w2301xx. Выполнить сквозное обновление сразу с w2211xx до w2301xx невозможно. При этом порядковый номер сборки в рамках обновления не влияет на порядок обновления. Например, с версии w221201 можно обновиться до w230103.

Инструкции по обновлению системы на ОС:

- Обновление Proceset на Windows
- Обновление Proceset на Linux

Обновление Proceset на Windows

Порядок действий при обновлении системы следующий:

1. На сервере Windows, где установлено ПО Proceset, остановите службу Infomaximum, чтобы завершить работу приложения.

2. Сделайте резервную копию каталога *%ProgramData%\Infomaximum*.

3. Сделайте резервную копию каталога *%ProgramFiles%\Infomaximum*.

4. Сделайте резервную копию данных сервера системной СУБД Clickhouse. При наличии других экземпляров СУБД ClickHouse, подключенных к Proceset, рекомендуем выполнить их резервное копирование тоже.

5. Настройте сервисный режим работы системы Proceset, чтобы при следующем запуске системы исключить прием данных от агентов мониторинга и вход рядовых пользователей в систему.

6. Установите переданный файл обновления на сервер Windows. Дождитесь успешной установки.

7. Если обновление предполагает установку нескольких файлов, остановите службу Infomaximum и выполните установку следующего по порядку обновления. Повторите данный шаг при необходимости.

8. После установки всех нужных файлов обновлений убедитесь, что веб-интерфейс системы доступен и работает.

9. Проверьте функционирование важных дашбордов.

10. Если все корректно работает, можно отключить сервисный режим и сделать систему доступной для пользователей и приема активности от агентов.

В случае проблем с обновлением выполните откат системы.

Создание резервной копии данных ClickHouse

Выполните следующие действия на сервере Linux, на котором запущен сервис СУБД ClickHouse:

1. Остановите службу СУБД. Для этого удалите docker service.

```
# docker service rm infomaximum-clickhouse`
```

2. С помощью команды ниже запустите создание резервной копии данных СУБД.

```
# docker run -it --rm \
--mount source=infomaximum-clickhouse,target=/clickhouse \
-v /tmp:/target \
infomaximum/infomaximum-clickhouse:22.8.3.13 \
/bin/bash -c "tar -cvf /target/clickhouse-$(date -u +%d.%m.%Y).tar /clickhouse"
```

Примечание.

- **infomaximum/infomaximum-clickhouse:22.8.3.13** — имя docker-образа с ClickHouse. Оно может отличаться в зависимости от версии.

- Резервная копия сохраняется в каталоге */tmp* операционной системы хоста.

3. Выполните запуск службы СУБД ClickHouse. Команда запуска может отличаться в зависимости от некоторых условий. Чтобы найти последнюю используемую команду запуска сервиса в истории команд bash, воспользуйтесь командой ниже:

```
# history | grep “docker service create --name infomaximum-clickhouse”
```

Если найти ранее используемую команду не удалось, обратитесь к документации по установке Proceset для формирования нужной команды запуска. Пример команды запуска службы ClickHouse:

```
# docker service create --name infomaximum-clickhouse \
--secret infomaximum_app_user \
--secret infomaximum_app_user_password_hash \
--secret infomaximum_external_user \
--secret infomaximum_external_user_password_hash \
--secret infomaximum_clickhouse_dhparam.pem \
--secret infomaximum_clickhouse.crt \
--secret infomaximum_clickhouse.key \
--publish published=8123,target=8123,mode=host \
--mount type=volume,src=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount type=volume,src=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
--no-resolve-image \
infomaximum/infomaximum-clickhouse:22.8.3.13
```

Примечание. Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

Сервисный режим работы Proceset

Для включения сервисного режима работы Proceset выполните следующее:

1. Убедитесь, что служба Infomaximum отключена.

2. Сделайте резервную копию файла *%ProgramData%Infomaximum\config\com.infomaximum.subsystem.frontend.json*.

3. Внесите изменения в этот файл. Установите значения ниже:

- "service_mode":true
- "service_mode_message": "Ваше сообщение об обслуживании"

Изменения вступят в силу при запуске службы Infomaximum.

Чтобы отключить сервисный режим, откатите изменения в файле и выполните перезапуск службы.

Откат системы в случае проблем с обновлением

1. На сервере, где установлен Proceset, остановите службу Infomaximum.

2. Удалите или переименуйте каталог *%ProgramData%Infomaximum*\. На его место подложите ранее сделанную резервную копию.

3. Удалите или переименуйте каталог *%ProgramFiles%Infomaximum*\. На его место подложите ранее сделанную резервную копию.

4. На сервере, где запущена служба ClickHouse, остановите сервис СУБД.

```
# docker service rm infomaximum-clickhouse
```

5. Удалите и заново создайте volume с данными ClickHouse.

```
# docker volume rm infomaximum-clickhouse
```

```
# docker volume create infomaximum-clickhouse
```

6. Выполните команду восстановления данных в volume (предполагается, что архив с резервной копией данных расположен в */tmp* хостовой ОС).

```
# docker run -it --rm \
--mount source=infomaximum-clickhouse,target=/clickhouse \
-v /tmp:/source infomaximum/infomaximum-clickhouse:22.8.3.13 \
/bin/bash -c “tar -xvf /source/clickhouse-$(date -u +%d.%m.%Y).tar -C /”
```

7. Выполните запуск службы ClickHouse ранее используемой для этого командой.

8. Выполните запуск службы Infomaximum на сервере Windows.

Обновление Proceset на Linux

Перед выполнением обновления серверных компонентов системы рекомендуем уточнить их текущие параметры запуска.

Уточнение текущих настроек запуска

Чтобы получить список запущенных docker-сервисов, на серверах с установленными компонентами системы выполните команду:

```
$ docker service ls
```

Сверьте имена служб в столбце NAME и текущие версии используемых образов в столбце IMAGE. Чтобы получить более подробные сведения, выполните команду для каждого из docker-сервисов:

```
$ docker service inspect --pretty [ имя_сервиса]
```

Если ранее служба запускалась от имени этого же пользователя операционной системы, выполните команду ниже, чтобы получить пример последних команд запуска службы:

```
$ history | grep 'docker service create'
```

Чтобы получить список доступных в системе docker-образов, выполните команду:

```
$ docker images
```

Создание резервных копий данных

Перед обновлением необходимо создать резервные копии данных серверных компонентов системы. Используйте удобные вам способы резервного копирования, например, создание снимков виртуальных машин.

Пример создания резервных копий служб Proceset и ClickHouse:

1. Остановите службы на серверах с помощью команды:

```
$ docker service rm infomaximum-app
$ docker service rm infomaximum-clickhouse
```

2. Сделайте копии данных.

- Копирование данных службы Proceset:

```
$ docker run -it --rm \
--mount source=infomaximum-app-data,target=/app-data \
-v /var/tmp:/target \
infomaximum/infomaximum_app:VERSION \
/bin/bash -c "tar -cvf /target/infomaximum-app-backup.tar /app-data"
```

Где VERSION — тег текущей версии Proceset.

- Копирование данных службы ClickHouse:

```
$ docker run -it --rm \
--mount source=infomaximum-clickhouse,target=/clickhouse-data \
-v /var/tmp:/target \
infomaximum/infomaximum-clickhouse:CH_VERSION \
```

```
/bin/bash -c "tar -cvf /target/clickhouse-backup.tar /clickhouse-data"
```

Где CH_VERSION — тег текущей версии ClickHouse.

В результате выполнения команд будет создано 2 архива в каталоге */var/tmp* операционной системы хоста. При необходимости переместите их в более надежное место.

Установка обновления Proceset

Чтобы установить обновление:

1. Загрузите образ новой версии Proceset в локальное хранилище образов Docker:

```
$ gunzip infomaximum_docker_app_d241208.tar.gz
$ docker load < infomaximum_docker_app_d241208.tar
```

Обратите внимание: Версия в имени файла приведена для примера. Укажите файл, соответствующий устанавливаемой версии.

2. Выполните запуск службы СУБД ClickHouse. Для восстановления последней команды запуска используйте раздел *Уточнение текущих настроек запуска*. Если найти ранее используемую команду не удалось, для формирования необходимой команды запуска изучите документацию по установке Proceset.

Пример команды запуска службы ClickHouse:

```
$ docker service create --name infomaximum-clickhouse \
--secret infomaximum_app_user \
--secret infomaximum_app_user_password_hash \
--secret infomaximum_external_user \
--secret infomaximum_external_user_password_hash \
--secret infomaximum_clickhouse_dhparam.pem \
--secret infomaximum_clickhouse.crt \
--secret infomaximum_clickhouse.key \
--publish published=8123,target=8123,mode=host \
--mount type=volume,src=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount type=volume,src=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
--no-resolve-image \
infomaximum/infomaximum-clickhouse:24.3.2.23
```

Обратите внимание: Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

3. Запустите службу Proceset с использованием docker-образа новой версии. Чтобы восстановить последнюю команду запуска, изучите раздел *Уточнение текущих настроек запуска*.

Пример команды:

```
$ docker service create --name infomaximum-app \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
```

```
--publish published=8010,target=8010,mode=host \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
-e JVM_MAX_MEMORY='4G' \
infomaximum/infomaximum_app:NEW_VERSION
```

Где NEW_VERSION — тег текущей версии Proceset.

Совет. Чтобы исключить доступ агентов мониторинга и пользователей к серверу, пока вы не убедитесь в работоспособности системы, первый запуск службы рекомендуется выполнить на другом сетевом порте. Для этого в параметре "published=" укажите номер любого свободного порта системы. Если система Proceset работает корректно, перезапустите сервис с указанием номера порта, который соответствует настройкам агентов мониторинга.

Обновление установлено. Проверьте работу веб-интерфейса системы, протестируйте подключение к ClickHouse в разделе *Хранилища данных*. Чтобы проверить отсутствие ошибок в логах Proceset, выполните команду:

```
$ docker service logs infomaximum-app 2>&1 | grep 'ERROR'
```

В случае критических проблем с работой обновления выполните откат обновления системы.

Откат обновления системы

Если обновление системы привело к критическим ошибкам, которые не позволяют продолжить работу системы, выполните откат обновления:

1. Остановите службы Proceset и ClickHouse.

- Остановка службы Proceset: *$ docker service rm infomaximum-app*
- Остановка службы ClickHouse: *$ docker service rm infomaximum-clickhouse*

2. При необходимости создайте копии файлов *docker volume* с поврежденными данными.

- Создание копий на сервере службы Proceset:

```
# tar -czv --same-permissions --same-owner -f infomaximum-app-volume-$(date -u +%d.%m.%Y).tar.gz /var/lib/docker/volumes/infomaximum-app-data/
```

- Создание копий на сервере службы ClickHouse:

```
# tar -czv --same-permissions --same-owner -f infomaximum-ch-volume-$(date -u +%d.%m.%Y).tar.gz /var/lib/docker/volumes/infomaximum-clickhouse/
```

3. Удалите и заново создайте *volume* с данными.

- Удаление и повторное создание *volume* с данными на сервере службы Proceset:

```
$ docker volume rm infomaximum-app-data
$ docker volume create infomaximum-app-data
```

- Удаление и повторное создание *volume* с данными на сервере службы ClickHouse:

```
$ docker volume rm infomaximum-clickhouse
$ docker volume create infomaximum-clickhouse
```

4. Восстановите данные служб Proceset и ClickHouse.

- Восстановление данных службы Proceset:

```
$ docker run -it --rm \
--mount source=infomaximum-app-data,target=/app-data \
-v /var/tmp:/target \
infomaximum/infomaximum_app:VERSION \
/bin/bash -c "tar -xvf /target/infomaximum-app-backup.tar -C /"
```

- Восстановление данных службы ClickHouse:

```
$ docker run -it --rm \
--mount source=infomaximum-clickhouse,target=/clickhouse-data \
-v /var/tmp:/target \
infomaximum/infomaximum-clickhouse:24.3.2.23 \
/bin/bash -c "tar -xvf /target/clickhouse-backup.tar -C /"
```

5. Запустите службы Proceset и ClickHouse с помощью команды, которая использовалась для этого ранее.

Настройка кластерного режима

Кластерный режим Proceset позволяет объединить основной сервер приложения Proceset с

дополнительными компонентами — такими как Агент автоматизации, Агент Webhook и AI Агент — в единую распределенную систему. Взаимодействие между компонентами осуществляется по протоколу gRPC с обязательным использованием взаимной аутентификации на основе X.509-сертификатов и шифрования трафика.

Этот режим повышает отказоустойчивость, масштабируемость и гибкость системы при обработке больших объемов данных или выполнении ресурсоемких задач.

Общий принцип настройки кластера

На каждой ноде кластера нужно указать:

- Уникальное имя ноды в рамках кластера (строка, например: "proceset")
- Порт для входящих gRPC-соединений (по умолчанию — 7000)
- Сертификат X.509 и приватный ключ в формате PEM/PKCS#8 для текущей ноды
- Список адресов других нод в формате host:port
- Сертификаты других нод для взаимной аутентификации и доверия

Примечание. Все соединения между нодами обязательно шифруются. HTTP-режим не поддерживается — только HTTPS/mTLS поверх gRPC.

Настройка на Windows (Legacy-установка)

Если сервер Proceset установлен на ОС Windows:

1. Создайте файл конфигурации: C:\ProgramData\Infomaximum\config\cluster.json

2. Заполните его содержимым:

```
{
  "network": {
    "current": {
      "name": "proceset",
      "port": 7000,
      "ssl": {
        "cert_chain_path": "ssl/proceset.crt",
        "private_key_path": "ssl/proceset.key",
        "trust_certs": ["ssl/agent01.crt", "ssl/agent02.crt"]
      }
    },
    "nodes": ["agent01.domain.local:7000", "agent02.domain.local:7000"]
  }
}
```

Где:

Параметр	Описание
name	Уникальное имя ноды в кластере (отображается в логах и веб-интерфейсе)
port	Порт для входящих gRPC-соединений

Параметр	Описание
cert_chain_path	Путь до сертификата X.509(PEM-сертификату) ноды Proceset (относительно C:\ProgramData\Infomaximum)
private_key_path	Путь к приватному ключу X.509 ноды Proceset в формате PKCS#8
trust_certs	Массив путей к сертификатам других нод для доверия
nodes	Массив, с адресами других нод в кластере (включая порт подключения к gRPC)

3. Перезапустите службу **Infomaximum**, чтобы применить изменения.

Настройка в Docker-контейнере (Linux)

При запуске Proceset в Docker (например, в Docker Swarm) конфигурация передается через переменные окружения и Docker secrets.

Переменные окружения

Переменная	Описание
CL_NAME	Уникальное имя текущей ноды (например, "proceset")
CL_PORT	Порт для gRPC (опционально, по умолчанию 7000)
CL_REMOTE_NODES	Список адресов других нод, разделенных точкой с запятой: "agent01.domain.local:7000;agent02.domain.local:7000"

Docker secrets для настройки кластера (файлы в контейнере)

Secret	Назначение
cluster_current.crt	Сертификат текущей ноды, используемый для защищенного взаимодействия между нодами кластера (PEM)
cluster_current.key	Закрытый ключ текущей ноды, соответствующий сертификату cluster_current.crt (PKCS#8)
cluster_remote_node_*.crt	Сертификаты удаленных нод кластера (например, cluster_remote_node_agent01.crt)

Пример запуска в Docker Swarm

```
docker service create --name infomaximum-app \
  --secret infomaximum_app_https_certificate \
  --secret infomaximum_app_https_certificate_password \
  --secret cluster_current.crt \
  --secret cluster_current.key \
  --secret cluster_remote_node_agent01.crt \
  --secret cluster_remote_node_agent02.crt \
  --mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
  --mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
  --publish published=443,target=8010,mode=host \
  --restart-max-attempts 5 \
  --restart-condition "on-failure" \
  -e JVM_MAX_MEMORY='4G' \
  -e FE_URL="https://server-name.domain.com" \
  -e FE_CORS_POLICY="*" \
  -e CL_NAME="proceset" \
  -e CL_REMOTE_NODES="agent01.domain.local:7000;agent02.domain.local:7000" \
  infomaximum/infomaximum_app:d250603
```

Примечание.

- Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

- Настройка агентов (автоматизации, Webhook, AI) осуществляется аналогичным образом — с собственным name, сертификатом и ссылками на другие ноды, включая Proceset.

Используемые сертификаты

В таблице ниже приведены все сертификаты, используемые в приведенном примере запуска Proceset, включая сертификаты для взаимодействия между нодами кластера и для HTTPS веб-интерфейса.

Secret	Назначение
infomaximum_app_https_certificate	Сертификат для шифрования HTTPS-соединения веб-интерфейса Proceset в формате PKCS#12 (.pfx)
infomaximum_app_https_certificate_password	Пароль от сертификата infomaximum_app_https_certificate
cluster_current.crt	Сертификат текущей ноды, используемый для шифрования соединения между Proceset и агентами автоматизации
cluster_current.key	Закрытый ключ текущей ноды, используемый для шифрования соединения между Proceset и агентами автоматизации
cluster_remote_node_agent01.crt	Сертификат (открытый ключ) агента автоматизации agent01
cluster_remote_node_agent02.crt	Сертификат (открытый ключ) агента автоматизации agent02

Проверка корректности настройки

После настройки всех нод:

1. Войдите в веб-интерфейс Proceset под учетной записью администратора.

2. Перейдите в раздел **Настройки → О системе**.

3. Убедитесь, что в списке отображаются все ноды кластера.

4. Для каждой из нод можно раскрыть подробности, чтобы просмотреть номера используемых версий компонентов

Перенос сервера Proceset с Windows на Linux

Важно.

- Компания «Инфомаксимум» переводит серверную часть системы и ее компоненты на Linux. Выпуск Windows-сборок при этом постепенно прекращается.
- План перехода:
  - С версии 2506 Windows-сбороки не предоставляются новым клиентам
  - Версия 2609 — последняя, которая может быть предоставлена существующим клиентам, использующим систему на Windows
  - Для версии 2609 в рамках LTS-обслуживания будут выпускаться патчи для Windows.
  - Предоставление патчей для этой версии планируется до 01.07.2027
  - С версии 2610 и далее Windows-сборки не выпускаются и не предоставляются
- Использование Linux позволяет:
  - Снизить зависимость от внешних ограничений, связанных с лицензированием и политикой обновлений
  - Применять открытые решения, соответствующие требованиям импортозамещения
  - Быстро получать исправления уязвимостей и обновления безопасности
  - Гибче адаптировать развертывание под различную инфраструктуру и оборудование (включая контейнеризацию)
  - Сфокусировать ресурсы разработки и поддержки на развитии продукта и качестве релизов
- После отказа от Windows-сборок работа агента мониторинга с аутентификацией через Active Directory будет сохранена.

Чтобы сохранить функциональность Proceset при переходе на Linux в Docker, корректно перенесите данные и настройте окружение. Эта инструкция описывает процесс переноса с сохранением всех настроек и данных, включая подключение к внешним системам (ClickHouse, Active Directory и др.).

Настройка аутентификации через Active Directory (Kerberos)

Важно. На Windows-сервере может быть настроена Стандартная Windows-аутентификация, однако на Linux эта опция недоступна. Если такая аутентификация была активна на исходном сервере (Сервер №1), отключите ее до переноса, иначе приложение на Linux-сервере не сможет запуститься корректно.

Если в вашей системе использовалась Стандартная Windows-аутентификация, выполните следующие действия до начала переноса:

1. В веб-интерфейсе Proceset перейдите в раздел *Аутентификация*.

2. Удалите или отключите аутентификацию типа Стандартная Windows-аутентификация.

3. Настройте вместо нее Kerberos-аутентификацию — это единственный поддерживаемый способ интеграции с Active Directory на Linux.

Подготовка к переносу

1. Убедитесь, что версия Proceset, установленная на Windows-сервере (далее — Сервер №1), совпадает с версией, планируемой к установке на Linux-сервере (далее — Сервер №2), включая все компоненты и патчи.

2. На Сервере №1 создайте полную резервную копию каталога: C:\ProgramData\Infomaximum.

3. Остановите службу *Infomaximum* через оснастку *Службы*.

4. Измените тип запуска службы на *Отключено*, чтобы предотвратить случайный запуск после переноса.

Установка и подготовка Linux-сервера

1. На Сервере №2 выполните первый запуск приложения Proceset согласно инструкции по установке. Это нужно для инициализации структуры каталогов и Docker-томов.

2. После первого запуска остановите службу Proceset

```
sudo docker service rm infomaximum-app
```

3. Очистите Docker-том с данными приложения.

```
sudo rm -rf /var/lib/docker/volumes/infomaximum-app-data/_data/*
```

Перенос данных

1. Из резервной копии на Сервере №1 извлеките каталоги databases и secret_key.

2. Создайте архив (например, Infomaximum.tar), содержащий эти каталоги.

```
tar -cvf Infomaximum.tar databases secret_key
```

3. Перенесите архив на Сервер №2, например, с помощью scp.

```
scp "Infomaximum.tar" user@linux-server:/tmp/
```

4. Распакуйте архив в целевой Docker-том.

```
sudo tar -xvf /tmp/Infomaximum.tar -C /var/lib/docker/volumes/infomaximum-app-data/_data/
```

5. Убедитесь, что структура тома корректна — в папке находятся каталоги databases/ и secret_key/

```
ls -la /var/lib/docker/volumes/infomaximum-app-data/_data/
```

6. Приложение внутри контейнера работает от пользователя с UID 1001, поэтому назначьте корректного владельца.

```
sudo chown -R 1001:1001 /var/lib/docker/volumes/infomaximum-app-data/_data/
sudo chown -R 1001:1001 /var/lib/docker/volumes/infomaximum-app-log/_data/
```

Завершение переноса

Важно.

- Не запускайте одновременно две копии Proceset, подключенные к одному экземпляру ClickHouse — это нарушит целостность данных.
- После переноса перепроверьте все настройки, включая подключение к внешним системам.
- Для соответствия функциональности Windows-версии корректно настройте Kerberos при использовании Active Directory.

1. Запустите приложение на Linux — выполните стандартный запуск Proceset по инструкции.

2. Убедитесь, что приложение запущено — контейнер с приложением находится в статусе Running.

docker-compose ps

3. Проверьте доступ к данным, настройкам, подключение к ClickHouse.

4. Убедитесь, что пользователи могут проходить аутентификацию (в том числе через AD, если настроено).

5. После подтверждения стабильной работы на Linux удалите приложение с Сервера №1 и убедитесь, что служба больше не может быть запущена.

Установка и настройка ClickHouse

Раздел содержит инструкции по установке, обновлению и удалению СУБД ClickHouse, подключению приложения Infomaximum к СУБД и настройке кластера ClickHouse в Docker с поддержкой репликации.

Установка и удаление БД ClickHouse

Для установки или удаления ClickHouse на Linux-сервере через Docker нужно заранее подготовить окружение с учетом системных требований.

С рекомендациями по системным требованиям вы можете ознакомиться в разделе Технические требования к серверному и аппаратному оборудованию.

Подготовка сертификата и закрытого ключа для сервера с СУБД «ClickHouse»

Прежде, чем приступить к установке и запуску ClickHouse, нужно обеспечить безопасность соединения между клиентскими приложениями и СУБД.

Для установки или удаления ClickHouse на Linux-сервере через Docker нужно заранее подготовить окружение с учетом системных требований.

Чтобы передача данных между приложением Proceset и СУБД ClickHouse происходила по защищенному протоколу HTTPS, подготовьте SSL-сертификат и закрытый ключ для сервера, где будет установлена СУБД. Файл сертификата должен быть с расширением .crt, ключ — файл с расширением .key. Формат PEM.

Для получения сертификата обратитесь к администраторам инфраструктуры открытых ключей (PKI) в вашей организации.

Вы также можете использовать самоподписанный сертификат. Ниже приведен пример команды для его генерации при помощи OpenSSL в ОС семейства Linux:

```
$ openssl req -x509 -nodes -newkey rsa:2048 -days 365 -keyout key.key -out cert.crt \
-subj "/C=RU/ST=./L=./O=./OU=./CN=server-name.domain.com/emailAddress=." \
-addext "subjectAltName = IP:1.1.1.1,DNS:server-name.domain.com"
```

Запуск Docker службы с СУБД «ClickHouse»

Когда SSL-сертификаты готовы, перейдите к установке СУБД ClickHouse в контейнере Docker.

Дистрибутив СУБД ClickHouse предоставляется в виде Docker-образа. Для установки контейнера с СУБД на сервере Linux выполните следующие действия:

1. Установите на сервер ПО Docker, следуя официальной документации https://docs.docker.com/engine/install/.

2. Загрузите переданный архив с Docker-образом на сервер.

3. Распакуйте архив и загрузите образ в локальное хранилище Docker.

```
# gunzip infomaximum_clickhouse-23.3.2.37.tar.gz
```

```
# docker load < infomaximum_clickhouse-23.3.2.37.tar
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

4. Если Docker Swarm еще не был инициализирован на сервере, выполните команду запуска.

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

5. Создайте секреты с учетными данными администратора СУБД ClickHouse.

```
# echo -n "infomaximum_user" | docker secret create infomaximum_app_user –
```

```
# echo -n "b5w4GDSg36" | sha256sum | awk '{print $1}' | docker secret create infomaximum_app_user_password_hash –
```

6. Создайте секреты с учетными данными пользователя СУБД ClickHouse с правами только на чтение.

```
# echo -n "infomaximum_read_user" | docker secret create infomaximum_external_user –
```

```
# echo -n "m26dhdhFdgj" | sha256sum | awk '{print $1}' | docker secret create infomaximum_external_user_password_hash -
```

7. Создайте секреты с SSL-сертификатом и ключом.

```
# docker secret create infomaximum_clickhouse.crt cert.crt
```

```
# docker secret create infomaximum_clickhouse.key key.key
```

```
# openssl dhparam 4096 | docker secret create infomaximum_clickhouse_dhparam.pem -
```

8. Создайте тома Docker (volume) для постоянного хранения данных работы контейнера.

```
# docker volume create infomaximum-clickhouse
```

```
# docker volume create infomaximum-clickhouse-log
```

9. Создайте службу (service).

```
# docker service create --name infomaximum-clickhouse \
--secret infomaximum_app_user \
--secret infomaximum_app_user_password_hash \
--secret infomaximum_external_user \
--secret infomaximum_external_user_password_hash \
--secret infomaximum_clickhouse_dhparam.pem \
--secret infomaximum_clickhouse.crt \
--secret infomaximum_clickhouse.key \
--publish published=8123,target=8123,mode=host \
--mount type=volume,src=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount type=volume,src=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
dockerhub.office.infomaximum.com/infomaximum/infomaximum-clickhouse:23.3.2.37
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

Если служба удачно запущена, результатом выполнения команды должно быть сообщение: *verify: Service converged*.

Чтобы убедиться, что СУБД работает стабильно, откройте в браузере страницу https://<адрес_сервера>:8123/ping или используйте команду curl https://<адрес сервера>:8123/ping. Если настройка выполнена верно, появится сообщение: *OK*.

Если сообщение не появляется:

- Проверьте, что контейнер работает.

```
# docker ps
```

- Обратитесь к логам приложения.

```
# docker service logs -f infomaximum-clickhouse
```

- Убедитесь, что открыты сетевые проходы до сервера.

Проверьте, не была ли допущена ошибка в шагах запуска, и попробуйте повторить создание службы. Для удаления нерабочей службы используйте команду docker service rm infomaximum-clickhouse.

Переменные окружения

При запуске контейнера можно передавать переменные окружения, которые модифицируют конфигурационные файлы приложения. Доступные параметры представлены в таблице.

Параметр	**Значение**	**Описание**
APP_USER_ACCESS_MANAGEMENT	0 или 1 По умолчанию — 0	Запрещает/разрешает от имени администратора СУБД ClickHouse создавать дополнительных пользователей и регламентировать права доступа через SQL-запросы
LIMIT_MEMORY	По умолчанию — 0.9 (90%)	Коэффициент максимального потребления оперативной памяти сервером для обработки всех SQL-запросов
PROMETHEUS	true или false По умолчанию — false	Включает передачу метрик для Prometheus Для корректной работы необходимо открыть порт 9363 в контейнере. Для этого в команду запуска службы добавьте: --publish published=9363,target=9363
SIZE_LOG_FILE	Размер лога в мегабайтах. По умолчанию — 1000 (1 ГБ)	Размер технических логов clickhouse-server.log и clickhouse-server.err.log
COUNT_LOG_FILE	По умолчанию — 10	Количество хранимых технических логов

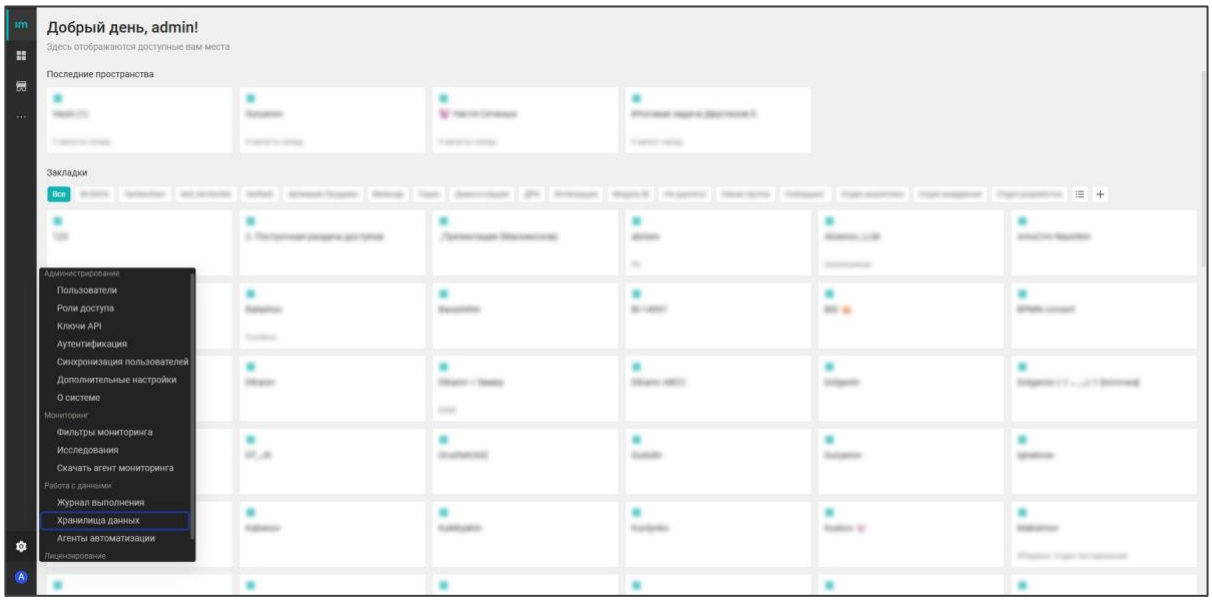
Подробнее о Prometheus в документации ClickHouse.

Подключение сервера Proceset к аналитической СУБД ClickHouse

После установки ClickHouse и приложения Infomaximum необходимо настроить соединение между ними, чтобы Proceset мог использовать базу данных для построения отчетов, выполнения SQL-запросов и работы с аналитикой.

После установки приложения Infomaximum и СУБД ClickHouse необходимо настроить их соединение. Для этого выполните следующие шаги:

1. Нажмите шестеренку в левом нижнем углу и откройте раздел *Хранилища данных*.



2. На странице хранилищ данных выберите нужное подключение и кликните по нему. Откроется страница настройки подключения.

3. Укажите параметры подключения к СУБД ClickHouse в соответствии с настройками вашего сервера:

- **Название** — удобное имя сервера для подключения
- **Хост** — FQDN сервера ClickHouse
- **Порт** — по умолчанию 8123
- **Имя пользователя** — логин администратора СУБД
- **Пароль** — администратора СУБД
- **SSL** (Вкл/Выкл) — необходимо включить, если ClickHouse работает по HTTPS. При включении SSL вы можете загрузить корневой сертификат, которым подписан сертификат сервера ClickHouse (или самоподписанный, в случае использования)
- **Аутентификация** — необходимо выбрать способ проверки подлинности при подключении к ClickHouse. По умолчанию используется стандартная аутентификация, но вам также доступна сквозная аутентификация (логин/пароль) в бета-версии
- **Время сессии (минуты)** — указывается количество минут, отведенное на хранение сессии по отдельному SQL-запросу. Рекомендуется оставить значение по умолчанию
- **Лимит памяти на запрос в скриптах (МБ)** — максимальный объем ОЗУ, который выделяется сервером ClickHouse на выполнение одного SQL-запроса, отправленного из скрипта. Значение по умолчанию — 4112 МБ
- **Лимит памяти на запрос в дашбордах (МБ)** — максимальный объем ОЗУ, который выделяется сервером ClickHouse на выполнение одного SQL-запроса, инициированного из дашборда. Значение по умолчанию — 1024 МБ

- **Одновременные подключения** — допустимое количество открытых сессий сервера Proceset с данным подключением ClickHouse. Рекомендуется оставить значение по умолчанию
- **Размер очереди на выполнение** — лимит количества SQL-запросов, ожидающих выполнения. Рекомендуется оставить значение по умолчанию

4. Нажмите **Проверить подключение** для тестирования корректности настроек.

5. Если тест соединения прошел успешно, нажмите **Сохранить**. Подключение выполнено.

Если тест соединения не проходит, обратитесь к логу приложения Infomaximum:

- На Windows: C:\ProgramData\Infomaximum\logs\main.log
- На Linux: var/lib/docker/volumes/infomaximum-app-log/_date/main.log или команда docker service logs infomaximum-app

Удаление СУБД ClickHouse

Когда ClickHouse больше не нужен или требуется переустановка, нужно корректно удалить все компоненты: саму службу, секреты, тома и настройки кластера Docker.

Для удаления СУБД ClickHouse выполните следующие команды на сервере, где она установлена:

```
# docker service rm infomaximum-clickhouse
```

```
# docker secret rm infomaximum_app_user infomaximum_app_user_password_hash infomaximum_clickhouse.crt infomaximum_clickhouse.key infomaximum_clickhouse_dhparam.pem infomaximum_external_user infomaximum_external_user_password_hash
```

```
# docker volume rm infomaximum-clickhouse infomaximum-clickhouse-log
```

```
# docker swarm leave –force
```

Найти команды для удаления Docker для конкретного дистрибутива Linux можно в официальной документации.

Обновление СУБД ClickHouse

Примечание. Обновление СУБД ClickHouse выполняется отдельно от обновления серверного приложения Infomaximum. ClickHouse работает только на Linux и обновляется с использованием Docker-образа.

Чтобы обновить СУБД ClickHouse:

1. Удалите службу ClickHouse в Docker:

```
# docker service rm infomaximum-clickhouse
```

2. Запустите создание резервной копии данных СУБД. Во время выполнения команды в консоли будут отображаться копируемые каталоги. Убедитесь, что после завершения процедуры нет сообщений об ошибках копирования.

```
# docker run -it --rm \
--mount source=infomaximum-clickhouse,target=/clickhouse \
-v /tmp:/target \
infomaximum/infomaximum-clickhouse:22.8.3.13 \
/bin/bash -c "tar -cvf /target/clickhouse-$(date -u +%d.%m.%Y).tar /clickhouse"
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

3. Запустите скрипт *setpermission.sh* из нового образа ClickHouse. Скрипт подготовит права файлов, чтобы при следующем запуске службы выполнилось обновление.

```
# docker run -it --user root --rm \
--mount source=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount source=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
infomaximum/infomaximum-clickhouse:23.3.2.37 /setpermission.sh
```

Обратите внимание: Версия образа приведена в качестве примера. При использовании команды замените указанную версию на ту, которую вы устанавливаете.

4. Запустите службу ClickHouse.

```
# docker service create --name infomaximum-clickhouse \
--secret infomaximum_app_user \
--secret infomaximum_app_user_password_hash \
--secret infomaximum_external_user \
--secret infomaximum_external_user_password_hash \
--secret infomaximum_clickhouse_dhparam.pem \
--secret infomaximum_clickhouse.crt \
--secret infomaximum_clickhouse.key \
--publish published=8123,target=8123,mode=host \
--mount type=volume,src=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount type=volume,src=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
--no-resolve-image \
infomaximum/infomaximum-clickhouse:23.3.2.37
```

5. Убедитесь, что служба ClickHouse работает — откройте в браузере страницу https://<адрес сервера>:8123/ping или используйте команду curl https://<адрес сервера>:8123/ping.

- Если служба работает, появится сообщение: OK
- Если сообщение не появляется, обратитесь к логам службы ClickHouse, выполнив команду:

```
# docker service logs -f infomaximum-clickhouse
```

Подключение приложения к аналитической СУБД ClickHouse

После установки приложения и СУБД ClickHouse их необходимо связать. Для этого:

В веб-интерфейсе приложения перейдите в настройки (иконка шестеренки в левом нижнем углу страницы). Далее в разделе *Работа с данными* откройте *Хранилища данных*.

Выберите подключение и кликните по нему. Откроется страница настройки подключения. Укажите параметры СУБД ClickHouse, заданные на странице Установка, активация и удаление приложения Infomaximum:

- Название сервера
- Хост — FQDN сервера ClickHouse
- Порт — по умолчанию 8123
- Имя пользователя — логин администратора СУБД
- Пароль пользователя — администратора СУБД
- SSL (Вкл/Выкл) — необходимо включить, если ClickHouse работает по HTTPS. При включении SSL вы можете загрузить корневой сертификат, которым подписан сертификат сервера ClickHouse (или самоподписанный, в случае использования)
- Лимит памяти на запрос в скриптах (МБ) — максимальный объем ОЗУ, который выделяется сервером ClickHouse на выполнение одного SQL-запроса, отправленного из скрипта. Значение по умолчанию — 4112 МБ
- Лимит памяти на запрос в дашбордах (МБ) — максимальный объем ОЗУ, который выделяется сервером ClickHouse на выполнение одного SQL-запроса, инициированного из дашборда. Значение по умолчанию — 1024 МБ
- Одновременные подключения — допустимое количество открытых сессий сервера Proceset с этим подключением ClickHouse. Рекомендуется оставить значение по умолчанию
- Размер очереди на выполнение — лимит количества SQL-запросов, ожидающих выполнения. Рекомендуется оставить значение по умолчанию
- Время сессии (минуты) — указывается количество минут, отведенное на хранение сессии по отдельному SQL-запросу. Рекомендуется оставить значение по умолчанию

Нажмите **Проверить соединение** для тестирования корректности настроек. Если тест соединения прошел успешно, нажмите **Сохранить**. Подключение выполнено.

Если тест соединения не проходит, обратитесь к логу приложения Infomaximum:

- Windows: C:\ProgramData\Infomaximum\logs\main.log
- Linux: var/lib/docker/volumes/infomaximum-app-log/_date/main.log или команда:

```
# docker service logs infomaximum-app
```

Создание учетных записей пользователей в ClickHouse

Создание учетных записей в ClickHouse выполняется через SQL-запрос и доступно, если при запуске контейнера передать переменную окружения:

```
APP_USER_ACCESS_MANAGEMENT='1'
```

Запрос:

```
CREATE USER [IF NOT EXISTS | OR REPLACE] name1 [ON CLUSTER cluster_name1]
    [, name2 [ON CLUSTER cluster_name2] ...]
  [NOT IDENTIFIED | IDENTIFIED {[WITH {no_password | plaintext_password | sha256_password | sha256_hash | double_sha1_password | double_sha1_hash}] BY {'password' | 'hash'}} | {WITH ldap SERVER 'server_name'} | {WITH kerberos [REALM 'realm']}]
  [HOST {LOCAL | NAME 'name' | REGEXP 'name_regexp' | IP 'address' | LIKE 'pattern'} [,...] | ANY | NONE]
  [DEFAULT ROLE role [,...]]
  [DEFAULT DATABASE database | NONE]
  [GRANTEES {user | role | ANY | NONE} [,...] [EXCEPT {user | role} [,...]]]
  [SETTINGS variable [= value] [MIN [=] min_value] [MAX [=] max_value] [READONLY | WRITABLE] | PROFILE 'profile_name'] [,...]
```

Где:

- ON CLUSTER — создание пользователей в кластере

Идентификация пользователей

Для идентификации пользователей существуют следующие способы:

- IDENTIFIED WITH no_password
- IDENTIFIED WITH plaintext_password BY 'qwerty'
- IDENTIFIED WITH sha256_password BY 'qwerty' or IDENTIFIED BY 'password'
- IDENTIFIED WITH sha256_hash BY 'hash' or IDENTIFIED WITH sha256_hash BY 'hash' SALT 'salt'
- IDENTIFIED WITH double_sha1_password BY 'qwerty'
- IDENTIFIED WITH double_sha1_hash BY 'hash'
- IDENTIFIED WITH ldap SERVER 'server_name'
- IDENTIFIED WITH kerberos or IDENTIFIED WITH kerberos REALM 'realm'

Для идентификации с sha256_hash используйте SALT — хэш должен быть вычислен от конкатенации 'password' и 'salt'.

Секция GRANTEES

Указываются пользователи или роли, которым разрешено получать привилегии от создаваемого пользователя при условии, что этому пользователю также предоставлен весь необходимый доступ с использованием GRANT OPTION. Параметры секции GRANTEES:

- user — указывается пользователь, которому разрешено получать привилегии от создаваемого пользователя
- role — указывается роль, которой разрешено получать привилегии от создаваемого пользователя
- ANY — любому пользователю или любой роли разрешено получать привилегии от создаваемого пользователя. Используется по умолчанию

- NONE — никому не разрешено получать привилегии от создаваемого пользователя

Вы можете исключить любого пользователя или роль, используя выражение EXCEPT. Например, CREATE USER user1 GRANTEES ANY EXCEPT user2. Это означает, что, если user1 имеет привилегии, предоставленные с использованием GRANT OPTION, он сможет предоставить их любому, кроме user2.

Выражение GRANT

Присваивает привилегии пользователям или ролям ClickHouse. Назначает роли пользователям или другим ролям.

Отозвать привилегию можно с помощью выражения REVOKE. Чтобы вывести список присвоенных привилегий, воспользуйтесь выражением SHOW GRANTS.

Синтаксис присвоения привилегий:

```
GRANT [ON CLUSTER cluster_name] privilege[(column_name [,...])] [,...] ON {db.table|db.*|*.*|table|*} TO {user | role | CURRENT_USER} [,...] [WITH GRANT OPTION] [WITH REPLACE OPTION]
```

- privilege — тип привилегии
- role — роль пользователя ClickHouse
- user — пользователь ClickHouse

WITH GRANT OPTION разрешает пользователю или роли выполнять запрос GRANT. Пользователь может выдавать только те привилегии, которые есть у него, той же или меньшей области действий. WITH REPLACE OPTION заменяет все старые привилегии новыми привилегиями для user или role, если не указано, добавляет новые привилегии.

Синтаксис назначения ролей:

```
GRANT [ON CLUSTER cluster_name] role [,...] TO {user | another_role | CURRENT_USER} [,...] [WITH ADMIN OPTION] [WITH REPLACE OPTION]
```

- role — роль пользователя ClickHouse
- user — пользователь ClickHouse

WITH ADMIN OPTION присваивает привилегию ADMIN OPTION пользователю или роли. WITH REPLACE OPTION заменяет все старые роли новыми ролями для пользователя user или role, если не указано, добавляет новые роли.

Синтаксис присвоения текущих привилегий:

```
GRANT CURRENT GRANTS{(privilege[(column_name [,...])] [,...] ON {db.table|db.*|*.*|table|*}) | ON {db.table|db.*|*.*|table|*}} TO {user | role | CURRENT_USER} [,...] [WITH GRANT OPTION] [WITH REPLACE OPTION]
```

- privilege — тип привилегии
- role — роль пользователя ClickHouse
- user — пользователь ClickHouse

Использование выражения CURRENT GRANTS позволяет присвоить все указанные и доступные для присвоения привилегии. Если список привилегий не задан, то указанный пользователь или роль получат все доступные привилегии для CURRENT_USER.

Привилегии

Привилегия — это разрешение на выполнение определенного типа запросов.

Привилегии имеют иерархическую структуру. Набор разрешенных запросов зависит от области действия привилегии. С полным списком привилегий ClickHouse ознакомьтесь по ссылке.

Примеры

Создать аккаунт mira, защищенный паролем qwerty:

```
CREATE USER mira IDENTIFIED WITH sha256_password BY 'qwerty';
```

Пользователь mira должен запустить клиентское приложение на хосте, где запущен ClickHouse.

Создать аккаунт john, назначить на него роли, сделать данные роли ролями по умолчанию:

```
CREATE USER john DEFAULT ROLE role1, role2;
```

Создать аккаунт john и установить ролями по умолчанию все его будущие роли:

```
CREATE USER john DEFAULT ROLE ALL;
```

Когда роль будет назначена аккаунту john, она автоматически станет ролью по умолчанию.

Создать аккаунт john и установить ролями по умолчанию все его будущие роли, кроме role1 и role2:

```
CREATE USER john DEFAULT ROLE ALL EXCEPT role1, role2;
```

Создать пользователя с аккаунтом john и разрешить ему предоставить свои привилегии пользователю с аккаунтом jack:

```
CREATE USER john GRANTEES jack;
```

Управление доступом

ClickHouse поддерживает управление доступом на основе ролей.

Объекты системы доступа в ClickHouse:

- аккаунт пользователя
- роль
- политика доступа к строкам
- профиль настроек
- квота

Вы можете настроить объекты системы доступа, используя:

- функцию SQL-ориентированного управления доступом
- конфигурационные файлы сервера: *users.xml* и *config.xml*

Рекомендуется использовать функцию SQL-ориентированного управления доступом. Оба метода конфигурации работают одновременно, поэтому, если для управления доступом вы используете конфигурационные файлы, вы можете плавно перейти на SQL-ориентированное управление доступом.

Примечание. Недопустимо одновременно использовать оба метода для управления одним и тем же объектом системы доступа.

Использование

По умолчанию сервер ClickHouse предоставляет аккаунт пользователя *default*, для которого выключена функция SQL-ориентированного управления доступом, но у него есть все права и разрешения. Аккаунт *default* используется во всех случаях, когда имя пользователя не определено. Например, при входе с клиента или в распределенных запросах. При распределенной обработке запроса *default* используется, если в конфигурации сервера или кластера не указаны свойства user и password.

Если вы начали пользоваться ClickHouse недавно, попробуйте следующий сценарий:

1. Включите SQL-ориентированное управление доступом для пользователя default.

2. Войдите под пользователем default и создайте всех необходимых пользователей. Не забудьте создать аккаунт администратора (GRANT ALL ON *.* TO admin_user_account WITH GRANT OPTION).

3. Ограничьте разрешения для пользователя default и отключите для него SQL-ориентированное управление доступом.

Примечание.

- Вы можете выдавать разрешения на базы данных или таблицы, даже если они не существуют.
- При удалении таблицы все связанные с ней привилегии не отзываются. Если затем создать новую таблицу с таким же именем, все привилегии останутся действительными. Чтобы отозвать привилегии, связанные с удаленной таблицей, необходимо выполнить, например, запрос REVOKE ALL PRIVILEGES ON db.table FROM ALL.
- У привилегий нет настроек времени жизни.

Аккаунт пользователя

Аккаунт пользователя — это объект системы доступа, позволяющий авторизовать кого-либо в ClickHouse. Аккаунт содержит:

- идентификационную информацию
- привилегии, определяющие область действия запросов, которые могут быть выполнены пользователем
- хосты, которые могут подключаться к серверу ClickHouse
- назначенные роли и роли по умолчанию
- настройки и их ограничения, которые применяются по умолчанию при входе пользователя
- присвоенные профили настроек

Привилегии присваиваются аккаунту пользователя с помощью запроса GRANT или через назначение ролей. Отозвать привилегию можно с помощью запроса REVOKE. Чтобы вывести список присвоенных привилегий, используется выражение SHOW GRANTS.

Запросы управления:

- CREATE USER
- ALTER USER
- DROP USER
- SHOW CREATE USER

Применение настроек

Настройки могут быть заданы разными способами: для аккаунта пользователя, для назначенных ему ролей или в профилях настроек. При входе пользователя, если настройка задана для разных объектов системы доступа, значение настройки и ее ограничения применяются в следующем порядке (от высшего приоритета к низшему):

1. Настройки аккаунта.

2. Настройки ролей по умолчанию для аккаунта. Если настройка задана для нескольких ролей, порядок применения не определен.

3. Настройки из профилей настроек, присвоенных пользователю или его ролям по умолчанию. Если настройка задана в нескольких профилях, порядок применения не определен.

4. Настройки, которые по умолчанию применяются ко всему серверу, или настройки из профиля по умолчанию.

Роль

Роль — это контейнер объектов системы доступа, которые можно присвоить аккаунту пользователя.

Роль содержит:

- привилегии
- настройки и ограничения
- список назначенных ролей

Запросы управления:

- CREATE ROLE
- ALTER ROLE
- DROP ROLE
- SET ROLE
- SET DEFAULT ROLE
- SHOW CREATE ROLE

Привилегии можно присвоить роли с помощью запроса GRANT. Для отзыва привилегий у роли ClickHouse предоставляет запрос REVOKE.

За полной документацией обращайтесь на официальный сайт ClickHouse.

Настройка кластера СУБД ClickHouse в Docker с поддержкой репликации

Настройка кластера ClickHouse с поддержкой репликации решает задачи отказоустойчивости, масштабируемости и целостности данных в распределенной аналитической системе. Репликация позволяет дублировать данные между несколькими нодами, обеспечивая их доступность даже при выходе из строя отдельных серверов. Кроме того, кластерный режим дает возможность распределять нагрузку между нодами, повышая общую производительность запросов и устойчивость системы к пиковым нагрузкам.

Запуск Docker служб кластера СУБД ClickHouse

Важно. Для обеспечения согласованности данных в кластере ClickHouse при использовании реплицированных таблиц необходимо нечетное количество нод в кластере.

Перед запуском Docker служб на каждой ноде кластера выполните те же настройки, что и при запуске ClickHouse в некластерном режиме.

Для конфигурации кластера ClickHouse при старте Docker служб на каждой ноде необходимо дополнительно указать следующие переменные окружения:

- CLUSTER_NODE_ID — индивидуальный порядковый номер ноды
- CLUSTER_NODES — список всех нод кластера (переменная должна быть одинаковой на всех нодах), в виде массива в формате:

```
[{"id":1,"host":"node1.example.com","replica":"01-
01"},{"id":2,"host":"node2.example.com","replica":"01-
02"},{"id":3,"host":"node3.example.com","replica":"01-03"}]
```

Где:

- id — порядковый номер ноды кластера
- host — адрес хоста ноды кластера
- replica — макрос, указывающий на номер реплики, в формате 01-<номер_реплики>

Допустимо использовать в качестве хранилища данных только две ноды кластера. Третья нода не хранит данные, но обязательно требуется для обеспечения кворума — она выступает в роли арбитра, позволяя кластеру сохранять согласованность и избегать «расщепления» (split-brain) при временной недоступности одной из нод.

Поскольку репликация синхронизирует данные между всеми нодами, указание `replica` на всех трех нодах приведет к избыточному хранению и замедлению записи — данные будут дублироваться на три узла вместо двух. Поэтому базовая и рекомендуемая конфигурация — две ноды с данными и одна нода только для кворума.

В этом случае в переменной `CLUSTER_NODES` значение `replica` указывается только для нод, участвующих в хранении данных, а для арбитра — опускается. Например:

```
[{"id":1,"host":"node1.example.com","replica":"01-01"},
{"id":2,"host":"node2.example.com","replica":"01-02"}, {"id":3,"host":"node3.example.com"}]
```

Важно. Третью ноду можно развернуть на любом доступном сервере — она потребляет минимальные ресурсы, так как не хранит пользовательские данные, а лишь участвует в

координации кластера, в том числе отслеживает изменения метаданных и помогает восстановить согласованность после временных сбоев. Эту ноду не следует указывать в настройках подключения к ClickHouse в приложениях — запросы направляются только на ноды с данными.

Важно. Для корректной работы кластера необходимо обеспечить сетевую доступность между нодами по портам 9000, 8123, 2181, 2180.

Защищенное взаимодействие между нодами кластера

Для обеспечения защищенного взаимодействия между нодами кластера и предотвращения несанкционированного доступа рекомендуется задать специальный ключ — секрет.

Секрет позволяет нодам кластера подтверждать подлинность друг друга. Без заданного ключа любой сервер может попытаться подключиться к кластеру.

Только серверы, знающие секрет, могут участвовать в кластере, поэтому на всех нодах кластера должен быть указан одинаковый ключ.

Чтобы задать секрет кластера ClickHouse, создайте Docker секрет командой:

```
echo -n "Secure_secret_123@" | docker secret create cluster_secret -
```

Где Secure_secret_123@ — установленный секрет.

Примечание. Рекомендуется использовать сложное значение секрета.

Пример команды для запуска Docker служб на одной из нод кластера:

```
docker service create --name infomaximum-clickhouse-node1 \
--secret infomaximum_app_user \
--secret infomaximum_app_user_password_hash \
--secret infomaximum_external_user \
--secret infomaximum_external_user_password_hash \
--secret infomaximum_clickhouse_dhparam.pem \
--secret infomaximum_clickhouse.crt \
--secret infomaximum_clickhouse.key \
--secret cluster_secret \
--publish published=8123,target=8123,mode=host \
--publish published=9000,target=9000,mode=host \
--publish published=2181,target=2181,mode=host \
--publish published=2180,target=2180,mode=host \
--mount type=volume,src=infomaximum-clickhouse,target=/var/lib/clickhouse/ \
--mount type=volume,src=infomaximum-clickhouse-log,target=/var/log/clickhouse-server \
-e CLUSTER_NODE_ID=1 \
-e CLUSTER_NODES='[{"id":1,"host":"node1.example.com","replica":"01-01"},{"id":2,"host":"node2.example.com","replica":"01-02"},{"id":3,"host":"node3.example.com","replica":"01-03"}]' \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
infomaximum/infomaximum-clickhouse:25.3.2.39
```

После запуска Docker служб на всех нодах кластера настройте кластерное подключение в Proceset. В качестве имени кластера укажите *default*.

Важно. При настройке подключения к кластерному ClickHouse на вкладке Хранилища данных в свойствах хранилища необходимо включить кластерный режим и указать все ноды в рамках кластера. Proceset будет распределять запросы к этим нодам самостоятельно. Если подключение к кластерному ClickHouse осуществляется через балансировщиков нагрузки (например, Nginx), то подключение к ClickHouse настраивается в кластерном режиме, а в качестве хоста указывается один адрес балансировщика.

Управление агентами

Раздел содержит инструкции по установке, обновлению, запуску и удалению агентов системы Proceset. Включает в себя материалы по агенту мониторинга, агенту автоматизации, AI-агенту и агенту Webhook.

Управление агентом мониторинга

Для работы мониторинга необходимо установить и настроить агент.

В этом разделе собраны инструкции по основным операциям с агентом мониторинга:

- Получение дистрибутива
- Установка, обновление и удаление агента мониторинга на Windows
    - Конфигуратор агента мониторинга
- Установка, обновление и удаление агента мониторинга на Linux
- Запуск и остановка агента мониторинга

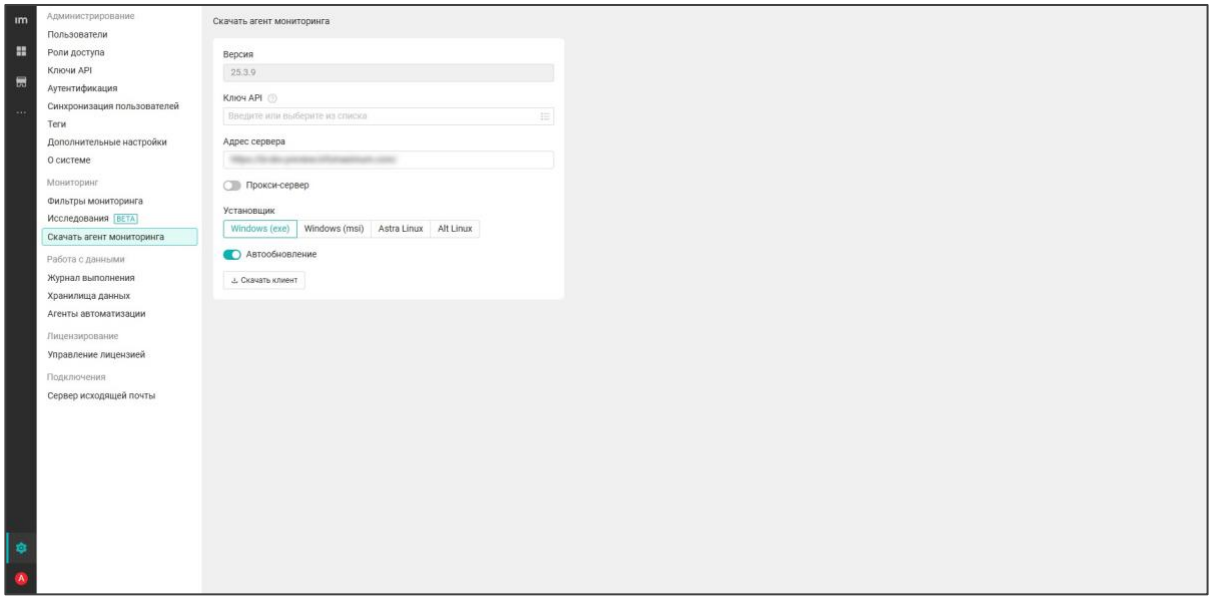
Получение дистрибутива

Чтобы установить агент мониторинга на рабочие станции сотрудников, сначала необходимо скачать его дистрибутив через веб-интерфейс системы. После установки агент автоматически подключается к серверу и начинает передавать данные в соответствии с заданной конфигурацией.

Чтобы скачать дистрибутив агента мониторинга, в веб-интерфейсе приложения перейдите в раздел *Настройки → Скачать агент мониторинга*.

Задайте параметры дистрибутива:

- **Ключ API** — с его помощью агент авторизуется на сервере приложения. По умолчанию в системе уже присутствует ключ API *Агент мониторинга*, который можно использовать для работы. При необходимости в системе можно создавать дополнительные ключи API для агентов
- **Адрес сервера** — URL, по которому агент мониторинга будет соединяться с сервером. Поле заполняется автоматически. При необходимости его можно исправить, чтобы адрес представлял собой полный FQDN, указанный в SSL-сертификате для сервера приложения
- **Прокси-сервер** — при необходимости можно указать отличные от системных настройки проксисервера, которые будут использоваться агентом (редко используется на практике)
- **Установщик** — если выбрать **Windows (exe)**, дистрибутив будет представлять собой исполняемый exeфайл. Если выбрать **Windows (msi)**, дистрибутив агента будет представлять собой архив с пакетом *.msi* и файлом готовых ответов *.mst*, которые можно использовать для установки агента при помощи групповой политики (GPO). Если выбрать **Astra Linux**, агент загрузится в формате *.deb*, если выбрать **Alt Linux** — в формате *.rpm*
- **Автообновление** — переключатель регулирует, будет ли включена функция автоматического обновления агента (агент самостоятельно загружает и устанавливает обновление с сервера Proceset, если оно доступно)



После указания всех настроек нажмите **Скачать клиент**, чтобы начать загрузку файла дистрибутива.

Скачанный файл можно использовать для установки агента мониторинга на рабочих местах сотрудников.

Установка, обновление и удаление агента мониторинга на Windows

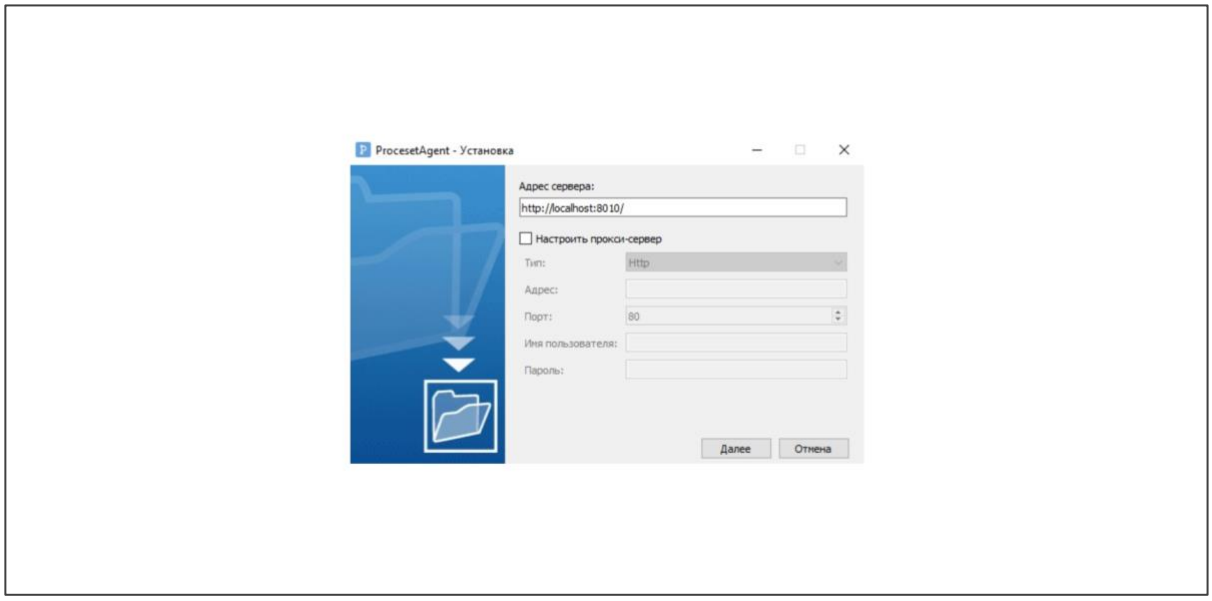
Чтобы агент мониторинга начал собирать данные на рабочих станциях Windows, его необходимо установить и настроить. На этой странице приведены инструкции по установке, обновлению и удалению агента.

Установка агента мониторинга

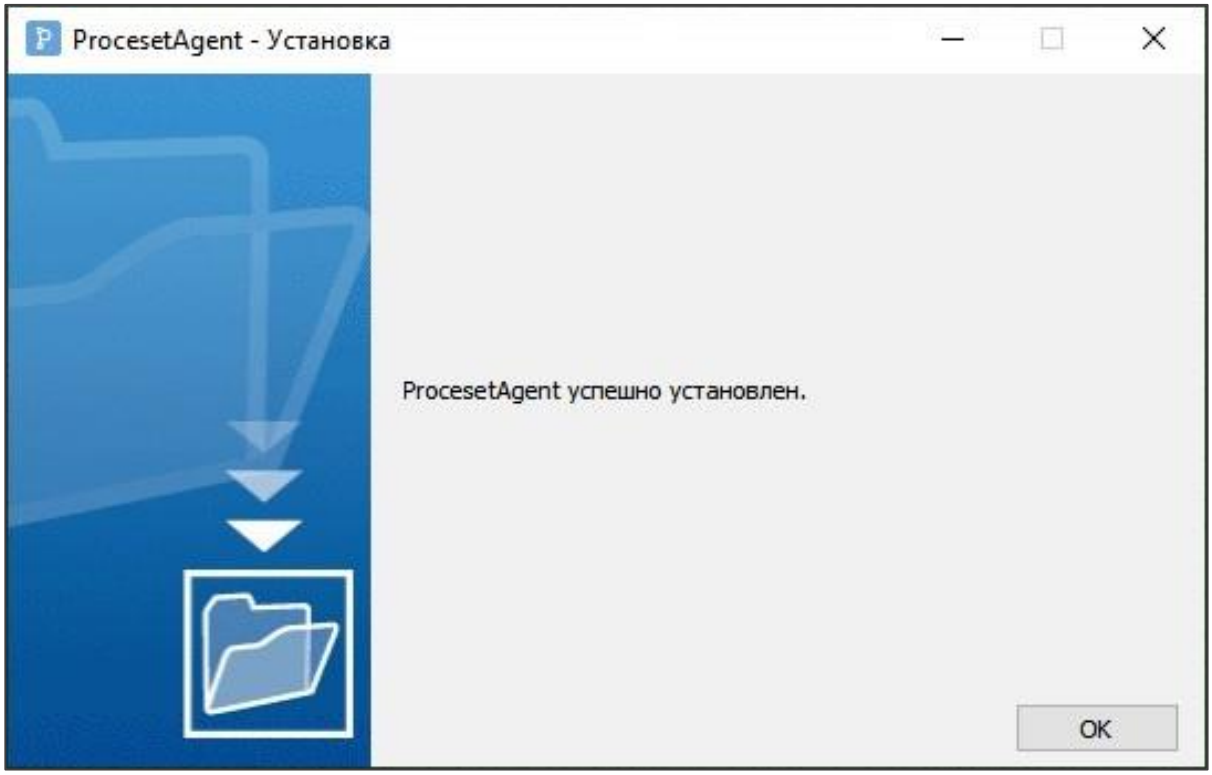
Установить агент мониторинга в Windows можно несколькими способами: через exe-дистрибутив, командную строку или msi-пакет.

Установка при помощи exe-дистрибутива, если он получен через интерфейс Proceset

Запустите файл *agent_setup.exe*, скачанный по кнопке **Windows (exe)**, на компьютере, где нужно установить агент мониторинга. Во время установки будут использоваться параметры, заданные при скачивании клиента.



Чтобы начать установку, нажмите **Далее**. По завершению установки нажмите **Ок**.

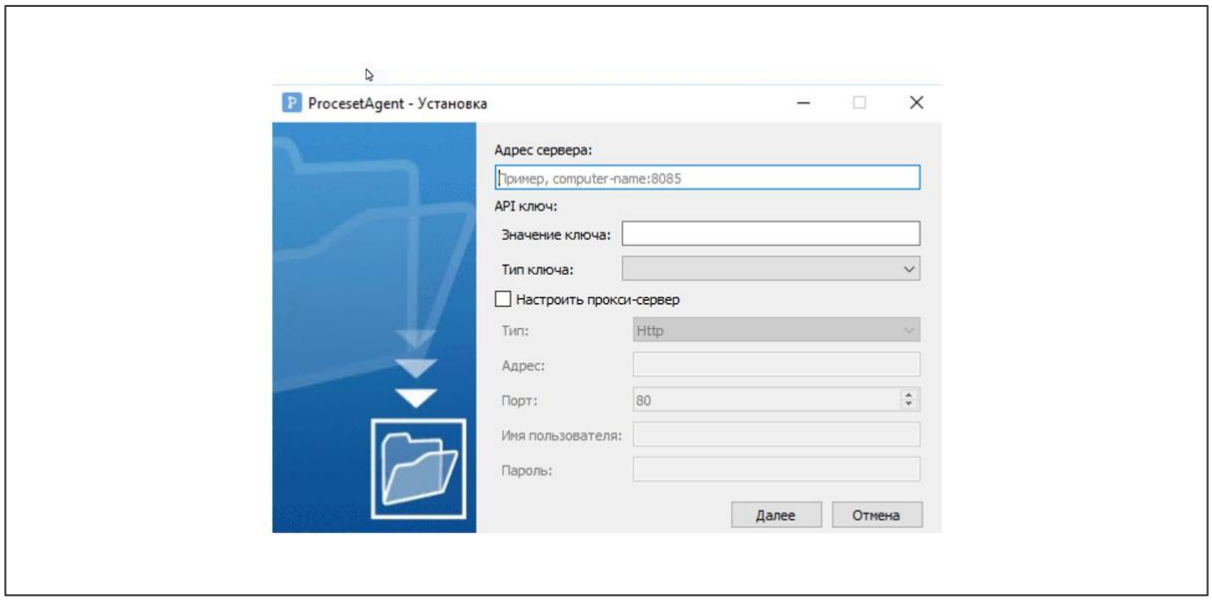


Если компьютер не имеет доступа к серверу приложения, установка завершится с ошибкой.

Установка при помощи exe-дистрибутива, если он получен не через интерфейс Proceset

Если *agent_setup.exe* был получен другим способом, то при установке поля настроек будут пустыми.

Чтобы выполнить установку, заполните поля **Адрес сервера** и **Значение ключа**, а также выберите тип ключа.



Установка через командную строку

Вы можете установить агент мониторинга с помощью файла *agent_setup.exe*, запустив его из командной строки.

Основные параметры

- silent — установка в фоновом режиме, без всплывающих окон и запросов
- action=install — выполняемая операция. Возможные значения: install — установка, update— обновление, remove — удаление

Дополнительные параметры

Если вы хотите настроить агент сразу при установке, можно добавить дополнительные параметры:

- autoupdate_enabled=true или false — включить или отключить автоматическое обновление агента. По умолчанию — зависит от настроек файла
- web_server.host=http://ваш-сервер:порт — адрес сервера Proceset, к которому должен подключаться агент. Например: http://server.contoso.com:8010
- auth.value= ваш_ключ_ API — ключ API для доступа к системе, который можно получить в вебинтерфейсе Proceset
- auth.type=none / ad / certificate — тип авторизации. Чаще всего используется none — простая авторизация по ключу

Важно. Если параметры не указаны, используются значения, встроенные в *agent_setup.exe*. Если же файл *agent_setup.exe* пустой и параметры не переданы, агент установится, но потребует ручной настройки.

Примеры команд

- «Тихая» установка с готовыми настройками:

```
agent_setup.exe silent action=install
```

- «Тихая» установка с указанием параметров вручную:

```
agent_setup.exe silent action=install autoupdate_enabled=true
web_server.host=http://server.contoso.com:8010
auth.value=01235ed47546459780f1001a36e41de2 auth.type=none
```

Установка при помощи msi-пакета

Msi-пакет содержит mst-файл с готовыми настройками и подходит для систем автоматизированного развертывания ПО, например, GPO, SCCM и других. Установка в этом случае происходит согласно рекомендациям компании разработчика соответствующей системы.

В архиве находятся 2 файла:

- *agent_setup.msi* — файл для установки
- *setting.mst* — файл конфигурации

При необходимости этот дистрибутив агента мониторинга можно установить с помощью стандартного инструмента Windows — msiexec.

Пример команды msiexec для «тихой» установки агента мониторинга:

```
msiexec.exe /i "путь_до_файла_agent_setup.msi" /q
TRANSFORMS="путь_до_файла_settings.mst"
```

Проверка и настройка агента мониторинга после установки

После установки агент мониторинга можно проверить и при необходимости перенастроить с помощью конфигуратора агента.

Обновление агента мониторинга

Агент мониторинга обновляется автоматически через сервер приложения. Исключение — если агент установлен с помощью msi-пакета или в настройках выключен параметр **Автообновление**.

Обновление агентов также возможно произвести вручную.

Восстановление старой версии агента возможно только через полную переустановку.

Автообновление

Раз в час служба агента проверяет наличие на сервере наличие новой версии агента. Если обновление доступно, агент скачивает и устанавливает его.

Перед установкой новой версии агента мониторинга создается резервная копия текущей версии.

Если обновление проходит успешно, резервные копии удаляются. Если при обновлении произошли ошибки, агент восстанавливается из резервной копии.

Поэтапное автообновление

Администратор может ограничить список компьютеров, для которых включено автообновление. Для этого:

1. В конфигурационный файл com.infomaximum.subsystem.monitoring.json, расположенный на сервере по пути C:\ProgramData\Infomaximum\, добавьте блок настройки phased_agent_upgrade.

2. Укажите в блоке:

- stable_version — стабильная версия агента (должна быть указана в формате XXX.XXX.XXX)
- host_mask — массив FQDN-имен компьютеров в формате регулярных выражений (не может быть пустым)

3. Сохраните изменения и перезагрузите службу Infomaximum.

Пример конфигурационного файла:

```
{
   "partition_life_circle_time": 3,
   "agent_activity_queue_dir": "agent_activity_queue",
   "rdb_ch_synchronization": {
      "delay": "10s",
      "interval": "30s"
   },
   "parsing_activity_enabled": true,
   "phased_agent_upgrade": {
      "stable_version": "2.6.41",
```

```
    "host_masks": [
      ".+\\.testinfomaximum\\.com"
    ]
  }
}
```

Примечание.

- Если версия агента меньше или равна stable_version, автообновление предоставляется только компьютеры, чьи FQDN соответствуют значениям, указанным в массиве host_mask.
- Если версия установленного агента выше stable_version, автообновление предоставляется вне зависимости от FQDN компьютера.

Ручное обновление

Ручное обновление агента мониторинга может потребоваться для обновления агента на конкретном компьютере. Его можно осуществить с помощью:

- Конфигуратора агента
- Дистрибутива агента новой версии

Для обновления с помощью конфигуратора:

1. Запустите конфигуратор агента, расположенный по пути C:\Program Files\ProcesetAgent\agent_configurator.exe.

2. В открывшемся окне нажмите **Обновить агент**.

Для обновления с помощью дистрибутива новой версии:

1. Запустите установку нового дистрибутива *agent_setup.exe*.

2. Подтвердите обновление.

Для обновления через командную строку, выполните команду:

```
agent_setup.exe silent action=update
```

Примечание. Запускайте команду из папки с файлом *agent_setup.exe* или укажите полный путь к файлу.

Особенности обновления агента мониторинга описаны на странице Известные ограничения.

Обновление с помощью msi-пакета

Если вы используете сторонние программы или скрипты для обновления агента мониторинга, рекомендуется делать это через msi-пакет.

Для управления установкой используется встроенная в Windows утилита msiexec.

Чтобы обновить агент с помощью msi-пакета:

1. Удалите старую версию агента с помощью команды:

```
agent_setup.exe silent action=update
```

Где:

- /x — команда на удаление, в которой нужно указать полный путь к вашему MSI-файлу
- /q — тихий режим

2. Установите новую версию с помощью команды:

```
msiexec /i "путь_до_файла_agent_setup.msi" /q TRANSFORMS="путь_до_файла_settings.mst"
```

Где:

- /i — команда на установку
- /q — тихий режим
- TRANSFORMS="..." — путь к файлу настроек, в котором сохранены параметры подключения (адрес сервера, ключ и т. д.)

Удаление агента мониторинга

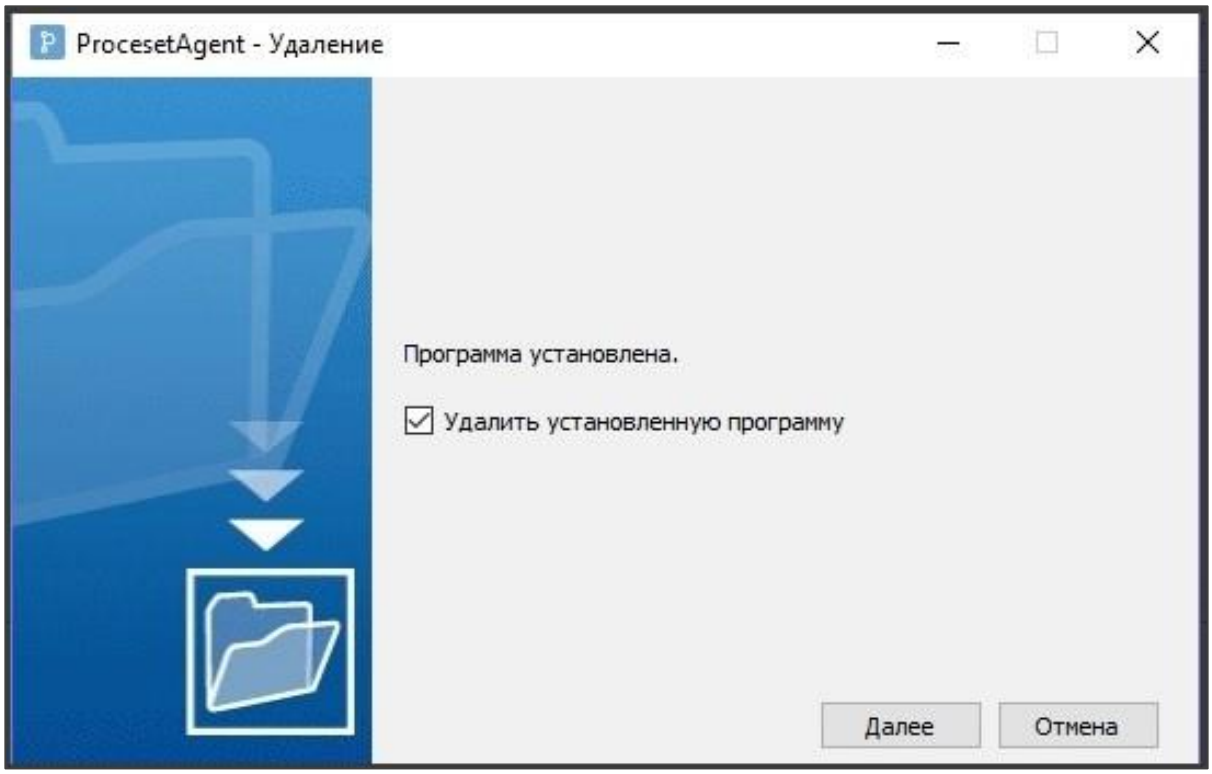
Удалить агент мониторинга с Windows можно несколькими способами: через exe-файл, командную строку или msi-пакет (если установка производилась через GPO или msiexec).

Удаление при помощи exe-файла

Для удаления агента мониторинга, установленного при помощи exe-файла:

1. Запустите файл C:\Program Files\ProcesetAgent\agent_setup.exe.

2. В открывшемся окне выберите удаление программы.



Удаление через командную строку

Выполните команду:

```
agent_setup.exe silent action=remove
```

Примечание. Запускайте команду из папки с файлом *agent_setup.exe* или укажите полный путь к файлу.

Удаление с помощью msi-пакета

Если агент был установлен с помощью msiexec (например, через GPO), используйте команду:

```
msiexec.exe /x "путь_до_файла_agent_setup.msi" /q
```

Важно. Не рекомендуется использовать файл C:\Program Files\ProcesetAgent\agent_setup.exe для удаления агента мониторинга, установленного через GPO.

Агент мониторинга, установленный через GPO, можно также удалить через **Панель управления** Windows в разделе **Программы и компоненты**.

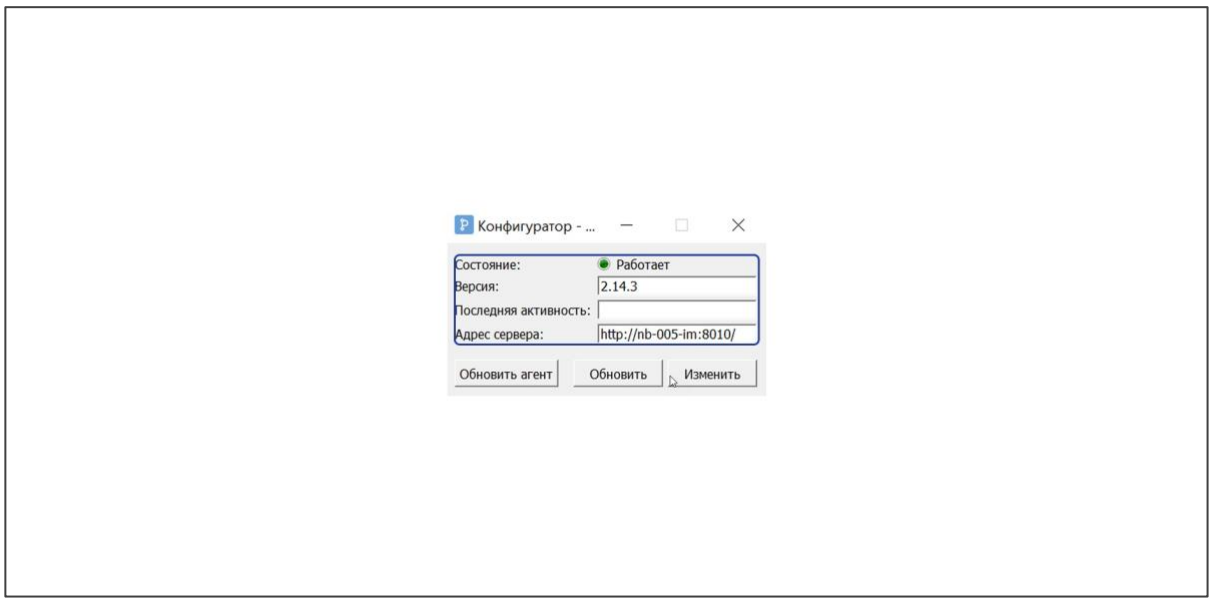
Конфигуратор агента мониторинга

В состав инсталлятора агента мониторинга входит конфигуратор agent_configurator, который можно использовать для проверки состояния и изменения настроек установленного агента мониторинга.

Исполняемый файл конфигуратора находится по пути C:\Program Files\ProcesetAgent\agent_configurator.exe. Запустить конфигуратор можно только под учетной записью с правами локального администратора.

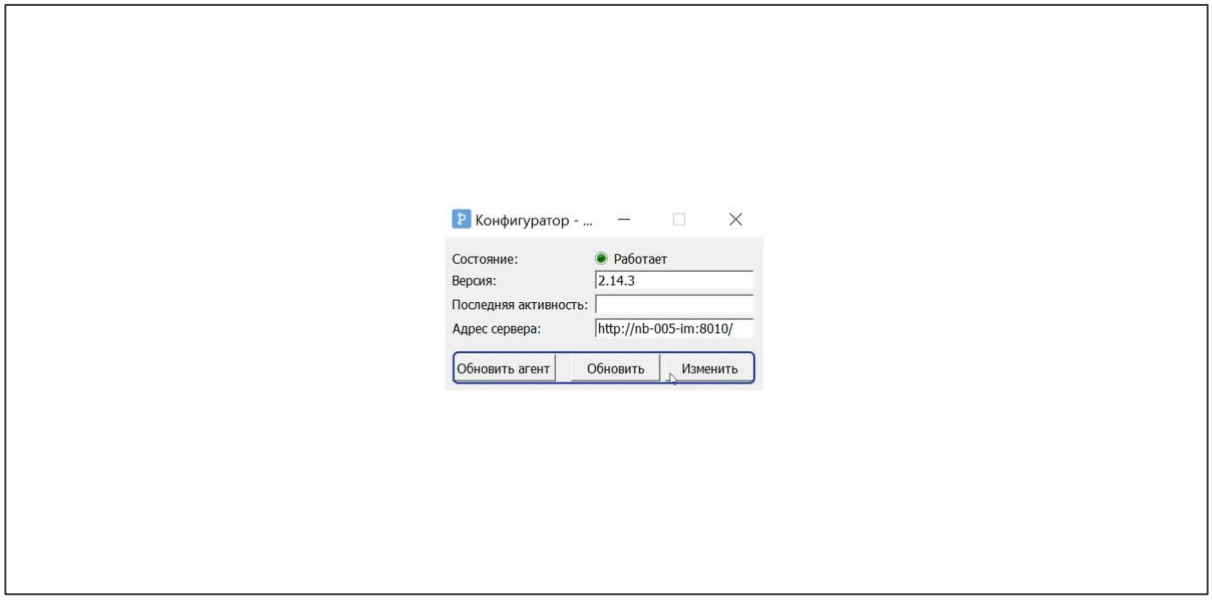
После запуска откроется окно конфигуратора со следующими полями:

- **Состояние** – отражает статус работы агента
  - *Работает*
  - *Не авторизован на сервере*
  - *Нет связи с сервером*
- **Версия** – текущая версия агента
- **Последняя активность** – время последней успешной передачи данных на сервер
- **Адрес сервера** – URL сервера приложения, указанный в настройках агента



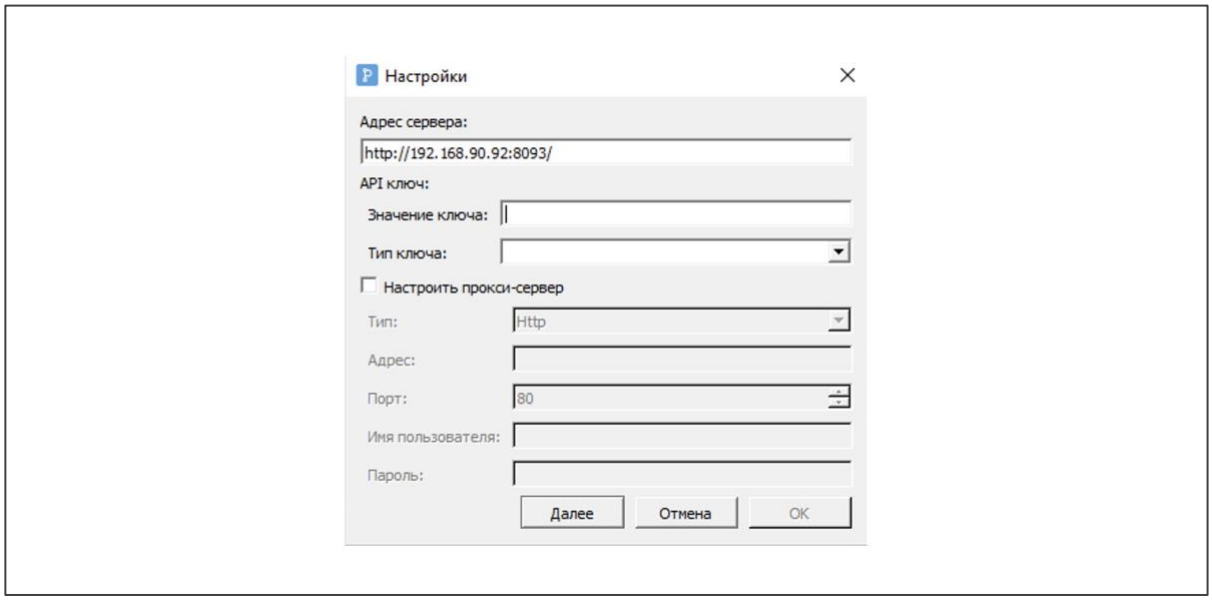
В окне конфигуратора доступны три действия:

- **Обновить агент** – выполняется запрос к серверу приложения для обновления версии агента. Если обновление есть, агент будет обновлен
- **Обновить** – обновить информацию в окне
- **Изменить** – открывает окно **Настройки**, позволяющее изменить адрес сервера и настройки прокси для этого агента



Чтобы изменить адрес сервера в окне **Настройки**:

- Впишите нужный URL адрес сервера приложения
- Заполните значение ключа API
- Выберите тип ключа:
    - Стандартный
    - SSL-сертификат
    - Active Directory



При необходимости в этом окне вы также можете изменить настройки прокси-сервера для агента мониторинга.

Установка, обновление и удаление агента мониторинга на Linux

Этот раздел описывает установку и настройку агента мониторинга на рабочих станциях под управлением Linux.

Агент мониторинга поддерживается на следующих версиях Linux:

- Astra Linux
- ALT Linux
- AlterOS
- РЕД ОС

Установка агента мониторинга

Для корректной работы агента мониторинга, когда работа на Astra Linux осуществляется через удаленный доступ (RDP), необходимо в параметрах xorg включить расширение record. Для этого необходимо:

1. В файле /etc/X11/fly-dm/fly-dmrc в переменной ServerArgsLocal= добавить +extension RECORD:

```
ServerArgsLocal=... +extension RECORD
```

2. В файл /etc/X11/trusted добавить /usr/libexec/ProcesetAgent/agent_inspector и /usr/libexec/ProcesetAgent/agent_service.

3. В файл /etc/X11/xrdp/xorg.conf добавить секцию:

```
Section "Extensions"
   Option "RECORD" "Enable"
EndSection
```

4. Перезагрузить компьютер.

Примечание. При установке инсталлятор не должен находиться в каталоге пользователя. Например, можно создать папку */opt/repo*, поместить в нее инсталлятор и запускать его из нее.

Для установки на Astra Linux введите команду:

```
# apt install /opt/repo/agent.deb
```

Для установки на Alt Linux введите команду:

```
# apt-get install /opt/repo/agent.rpm
```

Для установки на AlterOS или РЕД ОС введите команду:

```
# rpm --nosignature --nodigest -iv /tmp/agent_setup.rpm
```

Примечание. Путь к agent_setup.rpm указан для примера и может меняться в зависимости от ваших настроек.

Обновление агента мониторинга

Примечание. При обновлении агента мониторинга на Alt Linux и Astra Linux инсталлятор не должен находиться в каталоге пользователя. Например, можно создать папку */opt/repo*, поместить в нее инсталлятор и запускать его из нее.

Важно. Агент не обновится, если его текущая версия новее или совпадает с версией обновления.

Для обновления агента на *Astra Linux* воспользуйтесь командой:

```
# apt install /opt/repo/agent.deb
```

Для обновления агента на Alt Linux воспользуйтесь командой:

```
# apt-get install /opt/repo/agent.rpm
```

Для обновления агента на AlterOS или РЕД ОС воспользуйтесь командой:

```
# rpm --nosignature --nodigest -iv /tmp/agent_setup.rpm
```

Удаление агента мониторинга

Примечание. Для удаления используется стандартный пакетный менеджер соответствующей операционной системы. Параметр proceset-agent во всех командах — имя пакета.

Чтобы удалить агент на Astra Linux, введите команду:

```
# apt remove proceset-agent
```

Чтобы удалить агент на Alt Linux, введите команду:

```
# apt-get remove proceset-agent
```

Чтобы удалить агент на AlterOS или РЕД ОС, введите команду:

```
# rpm --nosignature --nodigest -e proceset-agent
```

Запуск и остановка агента мониторинга

После установки агент мониторинга запускается автоматически и начинает собирать данные о работе пользователей. При каждой загрузке операционной системы он стартует без участия пользователя, что обеспечивает непрерывный сбор информации.

Работа агента зависит от лицензии. Если для пользователя не назначена лицензия на мониторинг, агент не будет передавать данные на сервер, даже если он установлен и запущен.

Остановить агент можно при необходимости, например, для обслуживания системы или устранения ошибок. При остановке сбор данных полностью прекращается.

Запуск агента мониторинга

Агент мониторинга запускается автоматически — сразу после установки и при старте операционной системы. Ручных действий от пользователя не требуется.

Агент мониторинга состоит из нескольких компонентов:

- Служба *agent_service.exe* — запускается при загрузке Windows. Управляет инспекторами: запускает их в каждой пользовательской сессии, выгружает из неактивных сессий и перезапускает в случае сбоя. Процесс службы всегда один
- Инспектор *agent_inspector.exe* — приложение агента мониторинга. Запускается службой *agent_service.exe* в каждой пользовательской сессии. В диспетчере задач может отображаться несколько процессов *agent_inspector.exe* — по одному на каждую активную сессию
- Инспектор *agent_inspector_custom.exe* — используется для фиксации активности в Java-приложениях. Работает аналогично обычному инспектору *agent_inspector.exe*
- Приложение трекинга *agent_tracking.exe* — (может включаться в состав агента) дополняет функциональность агента и позволяет пользователям отмечать время, затраченное на выполнение задач

Важно. Для сбора активности сотрудников им необходимо назначить лицензию в веб-интерфейсе системы, в разделе *Настройки → Пользователи*. По умолчанию всем новым пользователям лицензия не назначена. Если лицензия не назначена сотруднику, агент не собирает активность и не передает ее на сервер.

Доступные лицензии:

- Мониторинг базовый
- Мониторинг расширенный

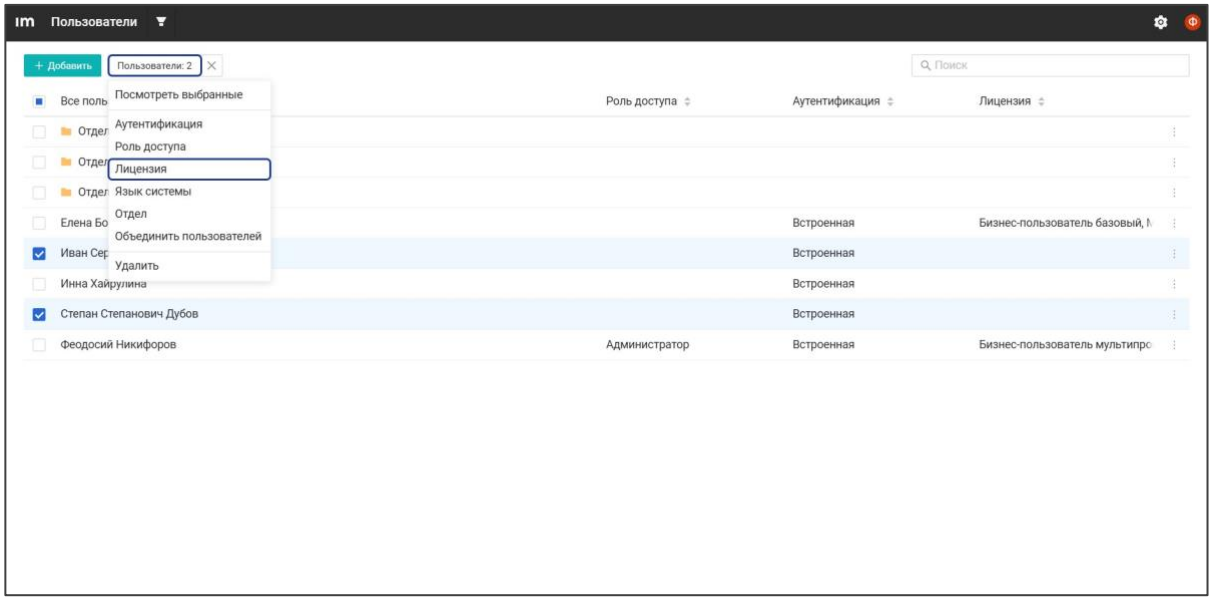
Лицензию можно назначить двумя способами:

- Через массовое назначение:

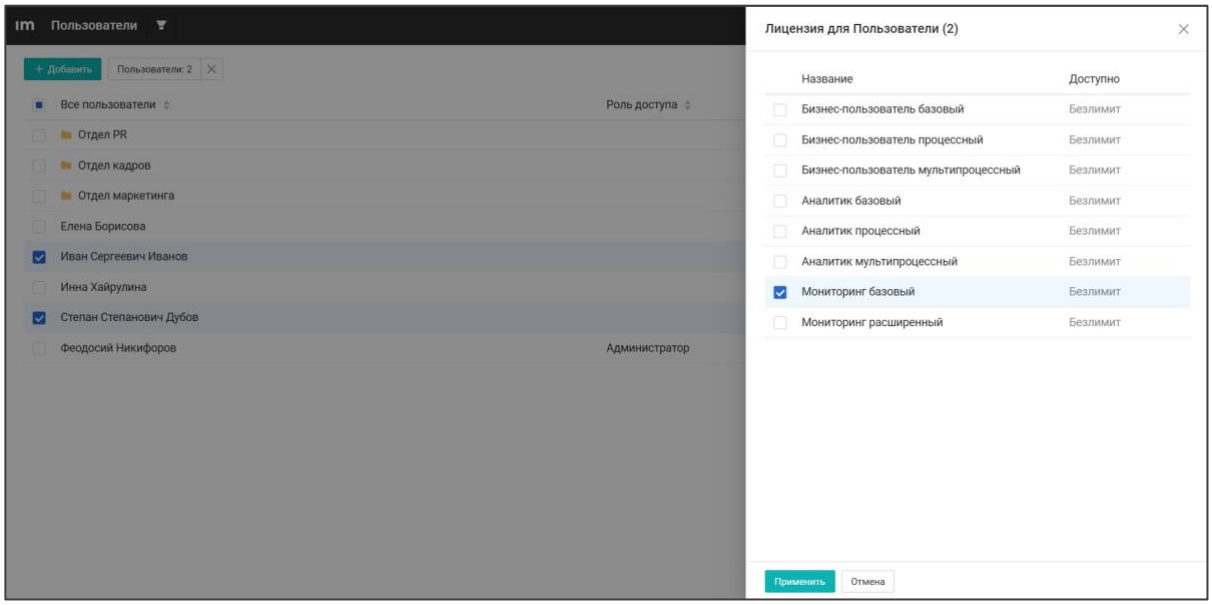
1. Перейдите в раздел *Пользователи*.

2. Выберите пользователей или отделы, по которым необходимо собирать активность.

3. Нажмите кнопку с выделенными пользователями/отделами рядом с кнопкой добавления и в выпадающем меню выберите **Лицензия**.



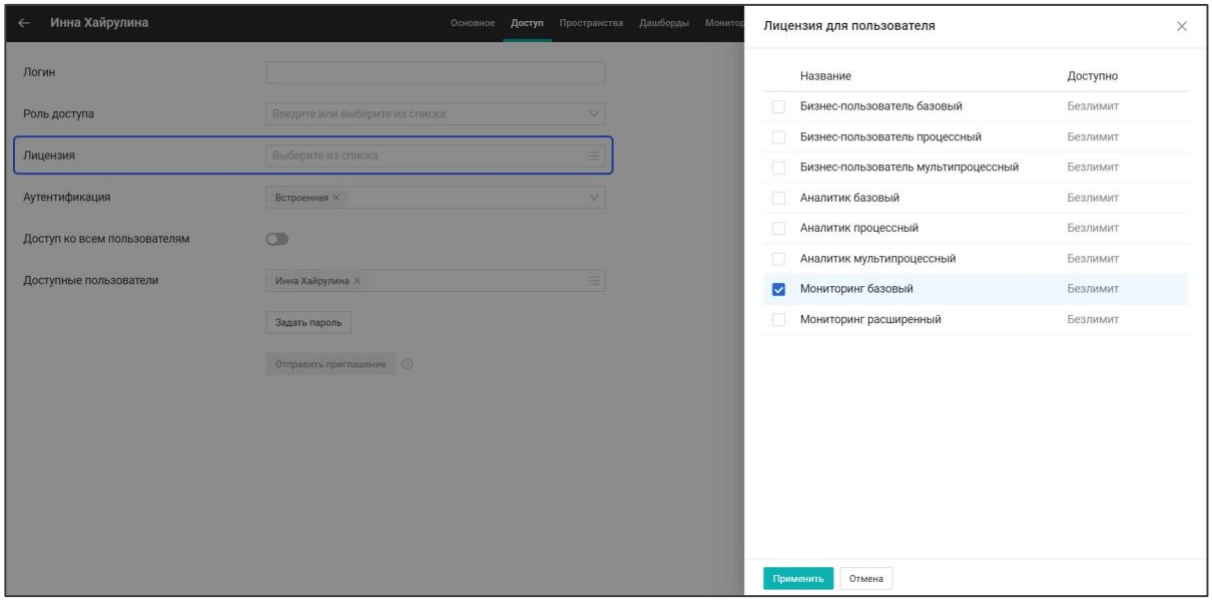
4. Выберите тип лицензии: **Мониторинг базовый** или **Мониторинг расширенный** и нажмите **Применить**.



- Через профиль конкретного пользователя:

  1. Перейдите в профиль пользователя, которому необходимо включить мониторинг.

  2. Перейдите во вкладку *Доступы*.



3. В поле **Лицензия** выберите тип: **Мониторинг базовый** или **Мониторинг расширенный**.

4. Нажмите **Применить**.

Изменение настроек мониторинга для пользователей вступает в силу не сразу. Агент получает новые настройки от сервера приложения раз в 4 часа.

Остановка агента мониторинга

Остановить работу агента можно через оснастку **Службы** в Windows. Для этого необходимы права локального администратора.

При остановке службы **Proceset Agent** мониторинг прекращается для всех пользователей компьютера.

Службу можно также перезапустить. Перезапуск службы можно использовать, если нужно быстрее получить обновленные настройки от сервера приложения.

Установка агента автоматизации

Агент автоматизации по умолчанию входит в базовый образ Proceset и разворачивается автоматически при установке, однако, в ряде случаев, может потребоваться разместить агент на отдельном сервере. Ниже описано, как развернуть агент автоматизации вручную и настроить его подключение.

В зависимости от типа подключения выберите подходящий способ запуска контейнера.

Запуск контейнера (работа по HTTP)

Если вы запускаете агент автоматизации вне кластерной среды и не планируете использовать защищенное соединение (HTTPS), запустите его в виде Docker-контейнера.

```
# docker run --name=infomaximum-automation-agent \
 -e CL_NAME='2' -e CL_PORT='7000' \
 -e CL_REMOTE_NODES='192.168.1.1:7000' \
 --mount type=volume,src=infomaximum-automation-agent-log,target=/var/log/infomaximum/ \
 -p 0.0.0.0:7000:7000 -d --restart=always \
 dockerhub.office.infomaximum.com/infomaximum/infomaximum_automation-agent:da220803
```

Примечание. Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

Запуск контейнера в Docker Swarm (HTTP)

Если вы уже используете Docker Swarm или планируете масштабировать систему, то агент автоматизации можно развернуть как отдельный сервис в кластере.

1. Инициализируйте Docker Swarm:

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

2. Создайте сервис:

```
# docker service create --name infomaximum-automation-agent \
 -e CL_NAME='2' -e CL_PORT='7000' \
 -e CL_REMOTE_NODES='192.168.1.1:7000' \
 --publish published=7000,target=7000,mode=host \
 --mount type=volume,src=infomaximum-automation-agent-log,target=/var/log/infomaximum/ \
 --limit-memory 30G \
 --limit-cpu 2 \
 --restart-max-attempts 5 \
 --restart-condition "on-failure" \
 dockerhub.office.infomaximum.com/infomaximum/infomaximum_automation-agent:da220803
```

Обратите внимание: Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

Запуск контейнера в Docker Swarm (HTTPS)

Чтобы запустить контейнер по HTTPS:

1. Инициализируйте Docker Swarm:

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

2. Создайте секреты:

- Сконфигурируйте SSL текущей ноды:

```
# docker secret create infomaximum_app_automation_agent.crt ${PATH_FILE}
```

```
# docker secret create infomaximum_app_automation_agent.key ${PATH_FILE}
```

- Сконфигурируйте SSL-подключение к main ноде:

```
# docker secret create infomaximum_app_remote_main.crt ${PATH_FILE}
```

- Обратите внимание, что при использовании секретов в кластерном режиме необходимо использовать метод с указанием источника и цели со строгим названием цели: cluster_current.crt — основной сертификат ноды, cluster_current.key — основной ключ ноды, cluster_remote_node_*.crt — сертификаты удаленных нод, где `*` заменяется на имя удаленной ноды.

3. Создайте сервис:

```
# docker service create --name infomaximum-automation-agent \
-e CL_NAME='2' -e CL_PORT='7000' \
-e CL_REMOTE_NODES='192.168.1.1:7000' \
--secret source=infomaximum_app_automation_agent.crt,target=cluster_current.crt \
--secret source=infomaximum_app_automation_agent.key,target=cluster_current.key \
--secret source=infomaximum_app_remote_main.crt,target=cluster_remote_node_main.crt \
--publish published=7000,target=7000,mode=host \
--limit-memory 30G \
--limit-cpu 2 \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
dockerhub.office.infomaximum.com/infomaximum/infomaximum_automation-agent:da220803
```

Обратите внимание: Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

Если сертификаты выписаны на DNS-имена, замените:

```
-e CL_REMOTE_NODES='192.168.1.1:7000'
```

На следующее:

```
-e CL_REMOTE_NODES='n1.local:7000'
```

При необходимости также добавьте параметр:

```
--host n1.local:192.168.1.1
```

Если:

- n1.local — основной сервер
- n2.local — внешний агент

То будет следующее соответствие:

- n2.crt → infomaximum_app_automation_agent.crt
- n2.key → infomaximum_app_automation_agent.key
- n1.crt → infomaximum_app_remote_main.crt

Установка и запуск AI агента

AI агент — это дополнительный модуль системы Proceset, предназначенный для работы с нейросетями и выполнения задач, связанных с LLM. AI агент — это опциональный компонент, который устанавливается только при необходимости использования нейросетей в системе Proceset.

AI агент обеспечивает выполнение следующих функций:

- Аналитический ассистент
- Распознавание текста (OCR)
- Распознавание речи (ASR)

Для каждой функции используется отдельный дистрибутив AI агента. В системе Proceset может одновременно работать несколько AI агентов.

Proceset и AI агенты взаимодействуют через протокол gRPC, который обеспечивает быстрый и безопасный обмен данными.

Все дистрибутивы AI агента представляют собой Docker-образы.

Технические требования представлены в разделе Технические требования к серверному и аппаратному оборудованию.

Запуск AI агента

Подготовка хостов и параметров конфигурации

Перед запуском AI агента необходимо определить сетевые адреса и параметры конфигурации.

Для примера рассмотрим следующие условные имена хостов, которые можно заменить на актуальные в вашей системе:

- proceset.domain.local — адрес хоста, на котором располагается система Proceset
- agent.domain.local — адрес хоста, на котором располагается AI агент

Во время запуска AI агента с помощью `docker run` необходимо задать переменные окружения, определяющие его параметры работы.

Переменная окружения	Описание
CL_NAME	Уникальное имя текущей ноды в рамках системы Proceset
CL_PORT	Порт ноды внутри сети Docker, используемый для входящих соединений gRPC
CL_REMOTE_NODES	Массив, в котором указываются полные адреса (включая порт) всех нод, с которыми происходит взаимодействие по gRPC
AI_LLM_MODEL	Название языковой модели Необходимо указывать только при использовании Аналитического ассистента (qwen-2.5-instruct-14B-Q4), чтобы в рабочих пространствах появилась вкладка ProcesetAI

Примечание. Если значение переменной `AI_LLM_MODEL` указано неверно, сервер приложения не запустится.

Таблица ниже содержит справочную информацию о соответствии названия модели ее возможностям.

Функционал	Имя языковой модели
Аналитический ассистент	qwen-2.5-instruct-14B-Q4
OCR	qwen-2.5-vl-instruct-7B-Q4

Подготовка образа и сертификатов

Загрузите образ с дистрибутивом AI агента на сервер с поддержкой GPU. После этого добавьте образ в локальный реестр Docker:

```
$ gunzip infomaximum_app_llm-agent:dlagdlag241201-llama3.1-instruct-8B-Q4.tar.gz
```

```
# docker load < infomaximum_app_llm-agent:dlag241201-llama3.1-instruct-8B-Q4.tar
```

Примечание. Версия в имени файла приведена для примера. Укажите файл, соответствующий устанавливаемой версии.

Подготовьте сертификаты X.509 для работы HTTPS на этой ноде в формате *PEM*. Сертификат (*.crt*) содержит открытый ключ, а файл ключа (*.key*) содержит закрытый ключ, который используется для шифрования соединения.

Разместите сертификаты в удобном каталоге на хосте, например, в /opt/agent/. В этот же каталог добавьте сертификат сервера Proceset (.crt).

Запуск контейнера

Выполните запуск контейнера с AI агентом с помощью следующей команды (внесите изменения в соответствии с вашей инфраструктурой):

```
 # docker run --name infomaximum-llama \
-d \
--gpus all \
--runtime=nvidia \
-e CL_NAME='agent' \
-e CL_PORT='7000' \
-e CL_REMOTE_NODES='proceset.domain.local:7000' \
-v /opt/agent/agent_cert.crt:/run/secrets/cluster_current.crt:ro \
-v /opt/agent/agent_key.key:/run/secrets/cluster_current.key:ro \
-v /opt/agent/proceset_cert.crt:/run/secrets/cluster_remote_node_agent.crt:ro \
-p 7000:7000 \
--restart=on-failure \
infomaximum/infomaximum_app_llm-agent:dlag241201-llama3.1-instruct-8B-Q4
```

Где:

- v /opt/agent/agent_cert.crt:/run/secrets/cluster_current.crt:ro — передача файла сертификата текущей ноды в контейнер
- v /opt/agent/agent_key.key:/run/secrets/cluster_current.key:ro — передача файла закрытого ключа текущей ноды в контейнер
- v /opt/agent/proceset_cert.crt:/run/secrets/cluster_remote_node_agent.crt:ro — передача файла сертификата сервера Proceset в контейнер
- infomaximum/infomaximum_app_llm-agent:dlag241201-llama3.1-instruct-8B-Q4 — имя образа (может отличаться в зависимости от версии)

Если контейнер AI агента успешно запущен, перезапустите сервер Proceset с новыми параметрами.

Настройка взаимодействия AI агента с сервером Proceset на ОС семейства Linux

Для работы AI агента с Proceset подготовьте сертификат X.509 для HTTPS на этой ноде в формате *PEM* (если у вас есть PFX-файл, его можно конвертировать в *PEM*), а также сертификат ноды AI агента.

Разместите сертификаты в каталоге /opt/agent/, после чего создайте секреты Docker на их основе:

```
# docker secret create cluster_current.crt /opt/agent/proceset.crt
# docker secret create cluster_current.key /opt/agent/proceset.key
# docker secret create cluster_remote_node_agent.crt /opt/agent/agent.crt
```

Перед запуском с новыми параметрами остановите службу Proceset:

```
# docker service rm infomaximum-app
```

После этого выполните запуск службы с новыми параметрами:

```
docker service create --name infomaximum-app \
-e AI_LLM_MODEL='llama-3.1-instruct-8B-Q4' \
-e CL_NAME='proceset' \
-e CL_PORT='7000' \
-e CL_REMOTE_NODES='agent.domain.local:7000' \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--secret cluster_current.crt \
--secret cluster_current.key \
--secret cluster_remote_node_agent.crt \
--mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
--publish published=443,target=8010,mode=host \
--publish published=7000,target=7000,mode=host \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
-e JVM_MAX_MEMORY='4G' \
-e FE_URL="https://proceset.domain.local" \
infomaximum/infomaximum_app:d241202
```

Примечание. Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

При запуске с новыми параметрами были добавлены:

- Переменные окружения (AI_LLM_MODEL, CL_NAME, CL_PORT, CL_REMOTE_NODES)
- Секреты (cluster_current.crt, cluster_current.key, cluster_remote_node_agent.crt)
- Публикация на хосте дополнительного порта для gRPC (publish published=7000,target=7000,mode=host)

Остальные параметры должны соответствовать вашей предыдущей команде запуска Proceset.

Проверка успешного запуска

Если запуск прошел успешно, перейдите в веб-интерфейс и убедитесь, что новые возможности корректно работают.

Дополнительная информация о функционале представлена в следующих разделах:

- ProcesetAI
- Работа с нейросетью

Установка и запуск агента Webhook

Агент Webhook — это дополнительный модуль системы Proceset, предназначенный для обработки входящих HTTP-запросов Webhook и запросов JS-трекера. Этот модуль может быть развернут на отдельном сервере, что позволяет разграничить доступ к функционалу системы в разных сетевых контурах, что особенно важно с точки зрения информационной безопасности.

Сервер Proceset и агент Webhook взаимодействуют по протоколу gRPC.

Все дистрибутивы агента Webhook предоставляются в виде Docker-образов.

Дополнительная информация о функционале в следующих разделах:

- Вебхук
- JS-трекер

Запуск агента Webhook

Подготовка хостов и параметров конфигурации

Перед запуском агента Webhook необходимо определить сетевые адреса и параметры конфигурации.

Для примера используются следующие имена хостов:

- proceset.domain.local — адрес хоста с установленной системой Proceset
- agent.domain.local — адрес хоста, на котором будет работать агент Webhook

При запуске агента Webhook необходимо задать переменные окружения, определяющие его параметры работы.

Переменная окружения	Описание
CL_NAME	Уникальное имя ноды в системе Proceset
CL_PORT	Порт ноды внутри сети Docker, используемый для входящих соединений gRPC
CL_REMOTE_NODES	Список адресов (включая порт) всех нод для взаимодействия по gRPC
FE_GRAPHQL_DISABLE	Отключение GraphQL на веб-сервере агента Webhook (true)
FE_CORS_POLICY	Доверенные узлы для CORS (список доменов через запятую, от которых на агент будут поступать HTTP-запросы)
FE_URL	URL-адрес ноды агента Webhook

Подготовка образа и сертификатов

Сначала загрузите дистрибутив агента Webhook на сервер, где он будет запущен. Затем добавьте образ в локальный реестр Docker:

```
$ gunzip infomaximum_docker_webhook_241201.tar.gz
```

```
# docker load < infomaximum_docker_webhook_241201.tar
```

Примечание. Версия в имени файла приведена для примера. Укажите файл, соответствующий устанавливаемой версии.

Подготовьте сертификат X.509 для работы HTTPS. Сертификат (*.crt*) содержит открытый ключ в формате *PEM*, а файл ключа (*.key*) содержит закрытый ключ для шифрования соединения в формате *PKCS8*.

Разместите сертификаты в удобном каталоге на хосте, например, в /opt/agent/. В этот же каталог добавьте сертификат сервера Proceset.

Если Docker Swarm ранее не был запущен на сервере, выполните команду инициализации:

```
# docker swarm init --advertise-addr 127.0.0.1:2377 --listen-addr 127.0.0.1:2377
```

- Секреты для SSL-сертификата и ключа сервера агента Webhook для работы gRPC:

```
# docker secret create cluster_current.crt ${PATH_FILE}
# docker secret create cluster_current.key ${PATH_FILE}
```

Где ${PATH_FILE} — до файла открытого (.crt) и закрытого (.key) ключа для агента Webhook

- Секрет для открытого ключа, который необходим для доверия серверу Proceset. Можно использовать сертификат CA, подписавший сертификат сервера Proceset, или сертификат самой ноды Proceset. Формат PEM или DER (.crt):

```
# docker secret create cluster_remote_node_main.crt ${PATH_FILE}
```

Где ${PATH_FILE} — путь к этому файлу

- Секреты для HTTPS-сертификата агента Webhook (PFX-файл, содержащий открытый и закрытый ключ X.509, а также пароль к нему)

```
# docker secret create infomaximum_app_https_certificate ${PATH_FILE}
# echo -n "pfx_password" | docker secret create infomaximum_app_https_certificate_password -
```

Где ${PATH_FILE} — путь к PFX-файлу

Создайте том Docker для хранения логов агента Webhook. Пример команды:

```
# docker volume create infomaximum-agent-log
```

Запуск контейнера агента Webhook

Запустите сервис агента Webhook с помощью следующей команды:

```
# docker service create --name infomaximum-automation-agent \
-e CL_NAME='agent' -e CL_PORT='7000' \
-e CL_REMOTE_NODES='proceset.domain.local:7000' \
-e FE_GRAPHQL_DISABLE='true' \
-e FE_CORS_POLICY='site1.example.ru,site2.example.ru' \
-e FE_URL='https://agent.domain.local' \
--secret cluster_current.crt \
--secret cluster_current.key \
--secret cluster_remote_node_main.crt \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--mount type=volume,src=infomaximum-agent-log,target=/var/log/infomaximum/ \
--publish published=7000,target=7000,mode=host \
--publish published=443,target=8010,mode=host \
```

```
--restart-max-attempts 5 \
--restart-condition "on-failure" \
infomaximum/infomaximum_webhook:241201
```

Примечание. Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

Если контейнер агента Webhook успешно запущен, перезапустите сервер Proceset с новыми параметрами.

В этом разделе описано, как настроить взаимодействие агента Webhook с сервером Proceset, установленным на ОС семейства Linux.

Для работы Proceset с агентом Webhook необходимо подготовить сертификат X.509 для gRPC в формате PEM (если у вас есть PFX-файл, его можно конвертировать в PEM), а также сертификат ноды агента Webhook.

Разместите сертификаты в каталоге /opt/agent/, затем создайте Docker-секреты на их основе:

```
# docker secret create cluster_current.crt ${PATH_FILE}
# docker secret create cluster_current.key ${PATH_FILE}
# docker secret create cluster_remote_node_agent.crt ${PATH_FILE}
```

Где ${PATH_FILE} — путь к соответствующему файлу.

Перед запуском с новыми параметрами остановите службу Proceset:

```
# docker service rm infomaximum-app
```

После этого выполните запуск службы с обновленными параметрами:

```
docker service create --name infomaximum-app \
-e CL_NAME='proceset' \
-e CL_PORT='7000' \
-e CL_REMOTE_NODES='agent.domain.local:7000' \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--secret cluster_current.crt \
--secret cluster_current.key \
--secret cluster_remote_node_agent.crt \
--mount type=volume,src=infomaximum-app-data,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log,target=/var/log/infomaximum/ \
--publish published=443,target=8010,mode=host \
--publish published=7000,target=7000,mode=host \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
-e JVM_MAX_MEMORY='4G' \
-e FE_URL="https://proceset.domain.local" \
dockerhub.office.infomaximum.com/infomaximum/infomaximum_app:d241201
```

Примечание. Имя образа с версией указано для примера. Замените версию на ту, которую вы устанавливаете.

В этом разделе описано, как настроить взаимодействие агента Webhook с сервером Proceset, установленным на ОС Windows.

Для работы агента Webhook с Proceset на сервере Windows требуется сертификат X.509 для gRPC в формате PEM (если у вас есть PFX-файл, его можно конвертировать в PEM), а также сертификат ноды агента Webhook.

На сервере Windows, где установлена служба Infomaximum (Proceset), необходимо внести изменения в файл конфигурации cluster.json, который располагается по пути C:\ProgramData\Infomaximum\config\cluster.json. Если файла cluster.json нет, создайте его.

Внесите в файл следующие изменения:

- В параметре target укажите адрес ноды агента Webhook (с портом gRPC)
- В блоке ssl укажите пути к сертификатам (можно указывать пути относительно каталога %ProgramData%\Infomaximum):
    - cert_chain_path — путь к сертификату текущей ноды Proceset
    - private_key_path — путь к закрытому ключу текущей ноды Proceset
    - trust_certs — массив, в который требуется добавить путь к сертификату ноды агента Webhook

Пример конфигурационного файла cluster.json:

```
{
   "network": {
      "current": {
         "name": proceset,
         "port": 7000,
         "ssl":{
            "cert_chain_path":"ssl/n1.crt",
            "private_key_path":"ssl/n1.key",
            "trust_certs":[
              "ssl/n2.crt"
            ]
         }
      },
      "nodes": [
         {
            "target": "agent.domain.local:7000"
         }
      ]
   }
}
```

После сохранения изменений конфигурационного файла перезапустите службу Infomaximum.

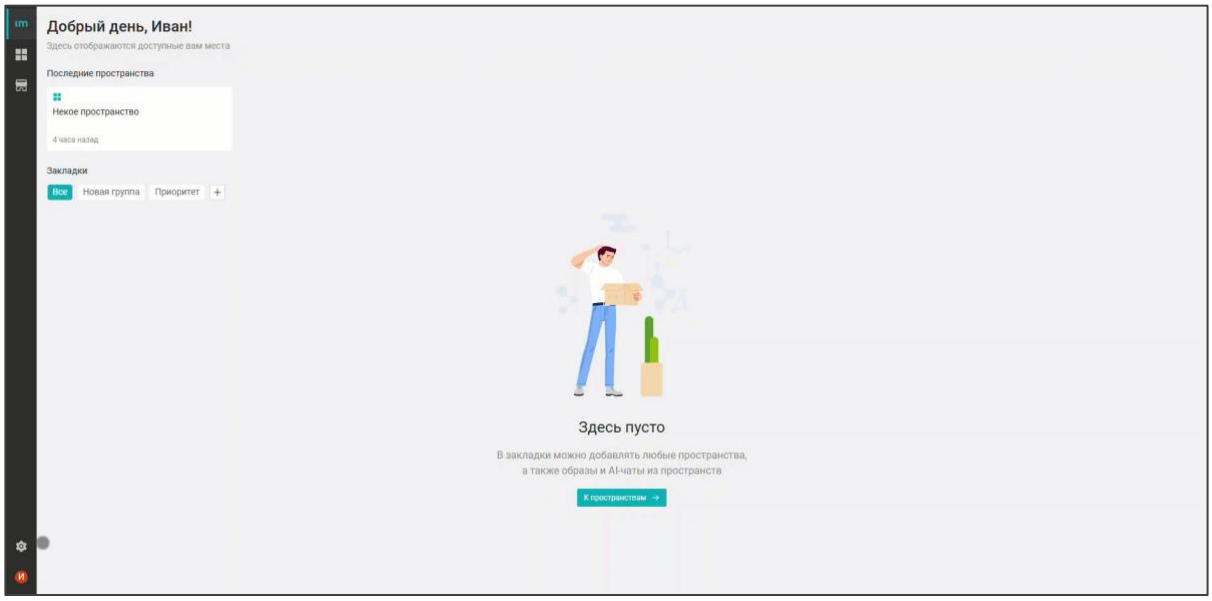
Проверка успешного запуска

Если контейнер или служба успешно запустились, перейдите в веб-интерфейс и убедитесь, что агент Webhook добавлен.

Нажмите шестеренку в нижней части боковой панели и выберите раздел О системе. На странице должна появиться новая строка с агентом Webhook.

Настройка системы

Для корректной работы и управления сотрудниками в системе существует ряд настроек. Чтобы перейти к настройкам, нажмите на шестеренку в нижней части боковой панели.



Все настройки содержат следующие разделы:

- Администрирование
- Таймшит (при наличии модуля «Таймшит»)
- Проекты (при наличии модуля «Таймшит»)
- График рабочих мест (при наличии модуля «Таймшит»)
- Мониторинг (при наличии модуля «Мониторинг»)
- Работа с данными
- Лицензирование
- Подключения

Пользователи

Добавление новых пользователей доступно пользователю, имеющему привилегию *Пользователи и отделы* с операциями доступа **R**, **W** и **C**. Также пользователю должен быть доступен хотя бы один отдел (настраивается в профиле пользователя). Добавить пользователя можно только в те отделы, к которым у него есть доступ. Для добавления пользователя в корневой отдел необходим доступ ко всем пользователям.

Вы можете добавить нового пользователя в настройках:

1. Кликните по иконке шестеренки.

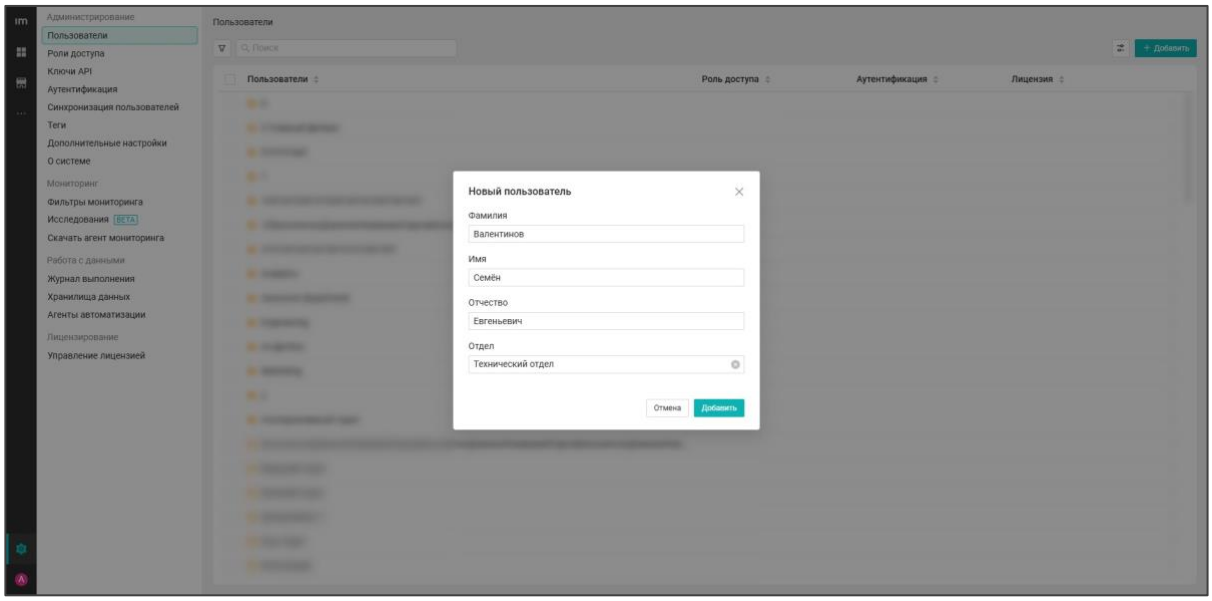
2. Выберите пункт **Пользователи**.

3. На открывшейся странице со списком всех пользователей и отделов нажмите кнопку **+ Добавить**.

4. Из раскрывающегося списка выберите пункт **Пользователя**.

5. В появившемся окне добавления нового сотрудника введите следующие данные:

- Фамилия
- Имя
- Отчество
- Отдел. Если конкретный отдел не указан, новый пользователь будет сохранен в текущем открытом отделе. Если текущий отдел недоступен, пользователь будет добавлен в первый доступный отдел в иерархии



6. Чтобы добавить нового пользователя (поля должны быть заполнены), нажмите **Добавить**.

7. Чтобы отменить добавление нового пользователя, нажмите **Отмена**.

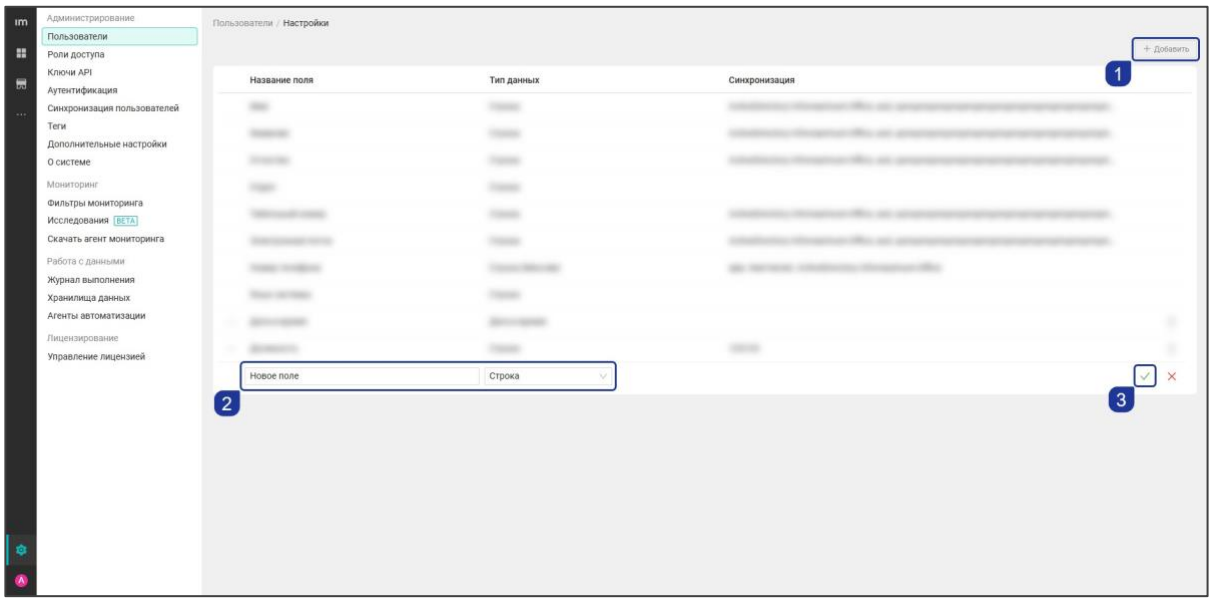
При добавлении пользователя вы автоматически перейдете к настройке его профиля.

Настройка полей пользователя

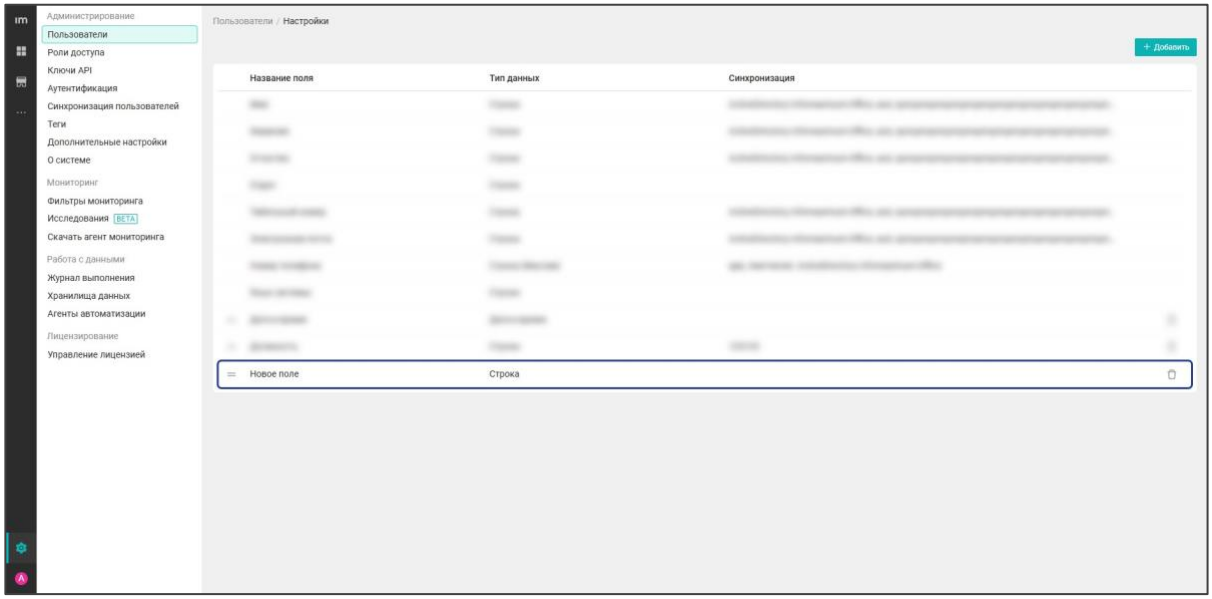
Чтобы настроить поля пользователя, кликните по кнопке **Настройки**.

Добавление поля пользователя

Чтобы добавить поле с данными в профили пользователей, нажмите + Добавить. Задайте название нового поля и тип данных для него, затем кликните по галочке.



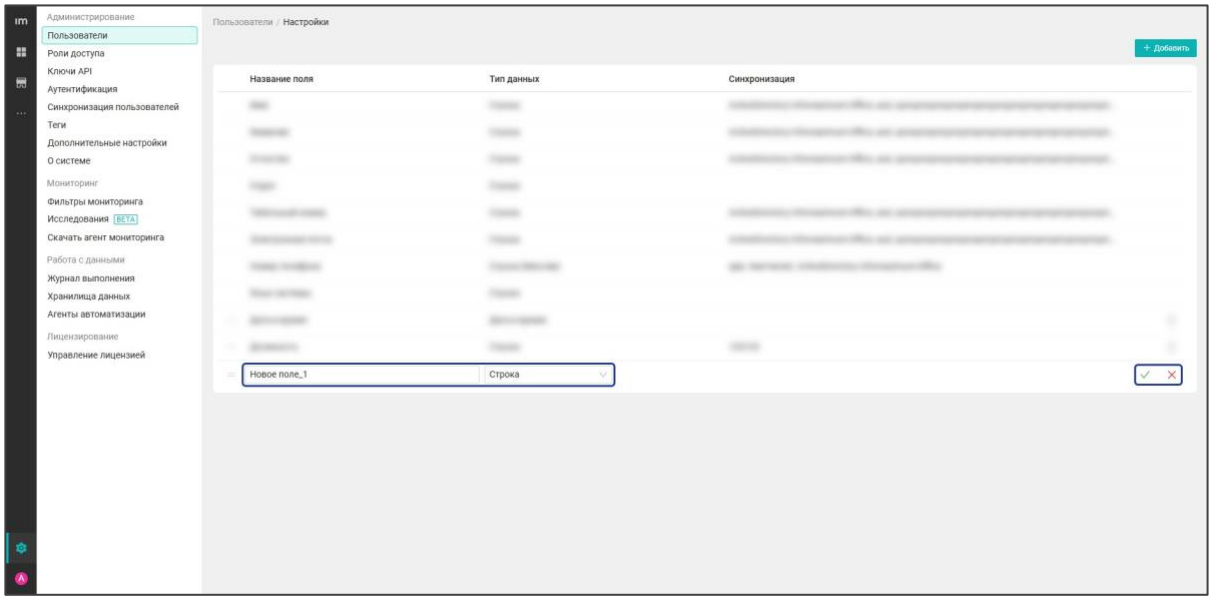
Добавленное поле появится в профилях пользователей с указанным типом данных.



Тип данных	Описание
Строка	Последовательный набор символов
Строка (Массив)	Набор строковых элементов
Число	Целочисленное значение
Дата	Дата без времени
Список	Списки системных пользователей и отделов
Дата и время	Дата и время
Логический	Принимает значение true или false

Редактирование поля пользователя

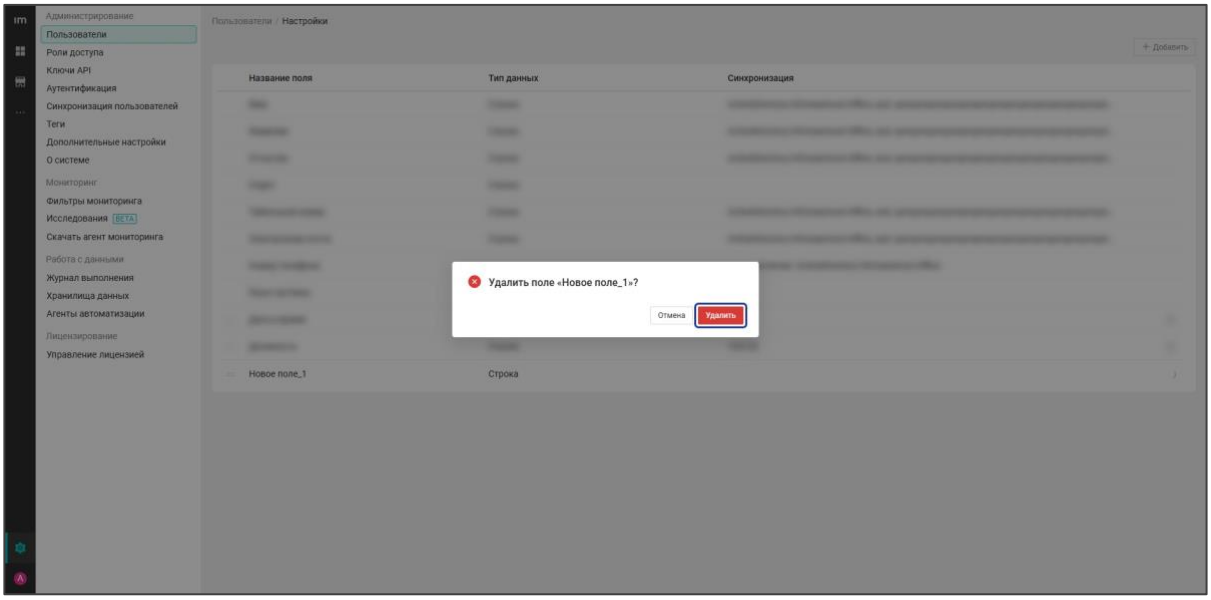
Чтобы отредактировать поле, кликните по его названию или типу данных. Внесите изменения и подтвердите их.



Удаление поля пользователя

Чтобы удалить поле, нажмите иконку корзины.

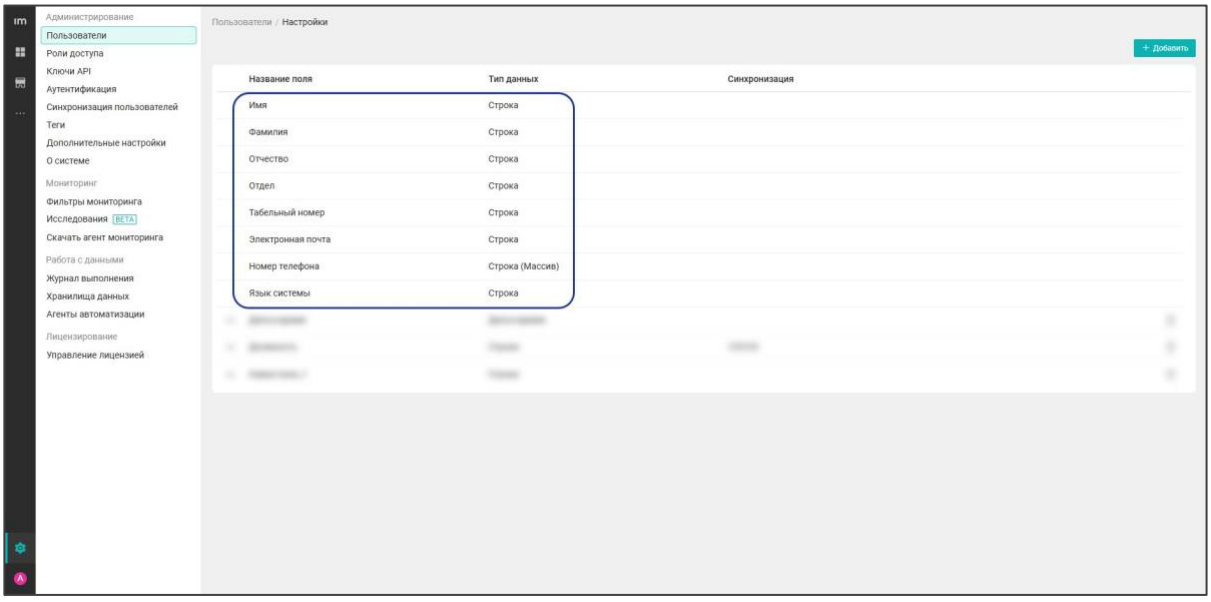
Подтвердите удаление в открывшемся модальном окне.



Предустановленные поля пользователя

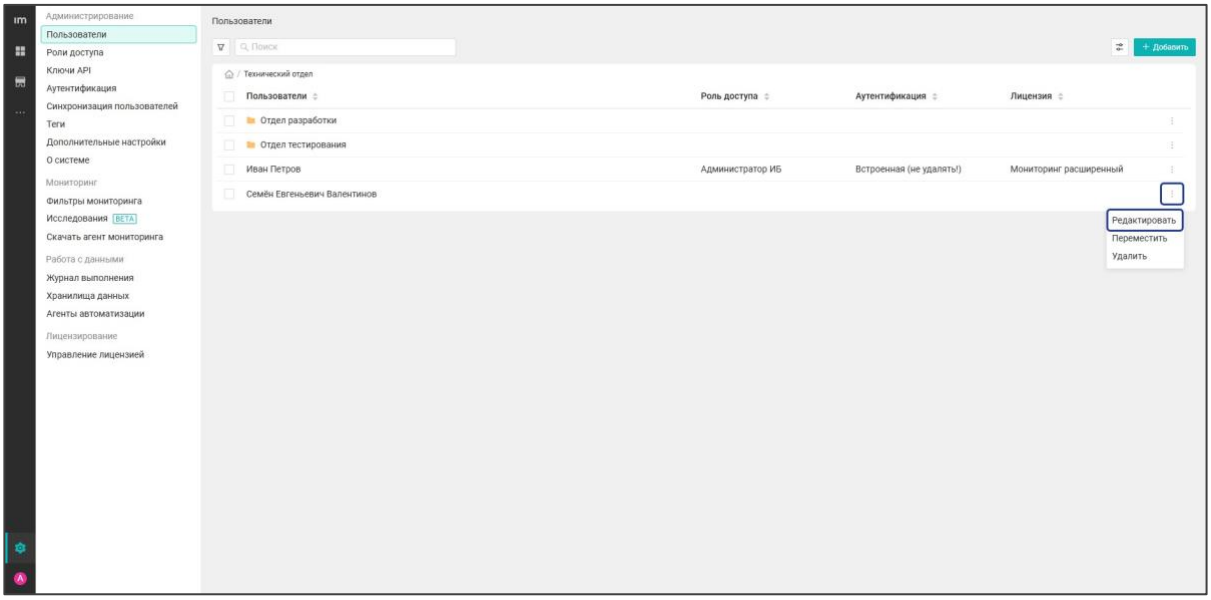
Невозможно отредактировать или удалить предустановленные поля:

- Имя
- Фамилия
- Отчество
- Отдел
- Табельный номер
- Электронная почта
- Номер телефона
- Язык системы



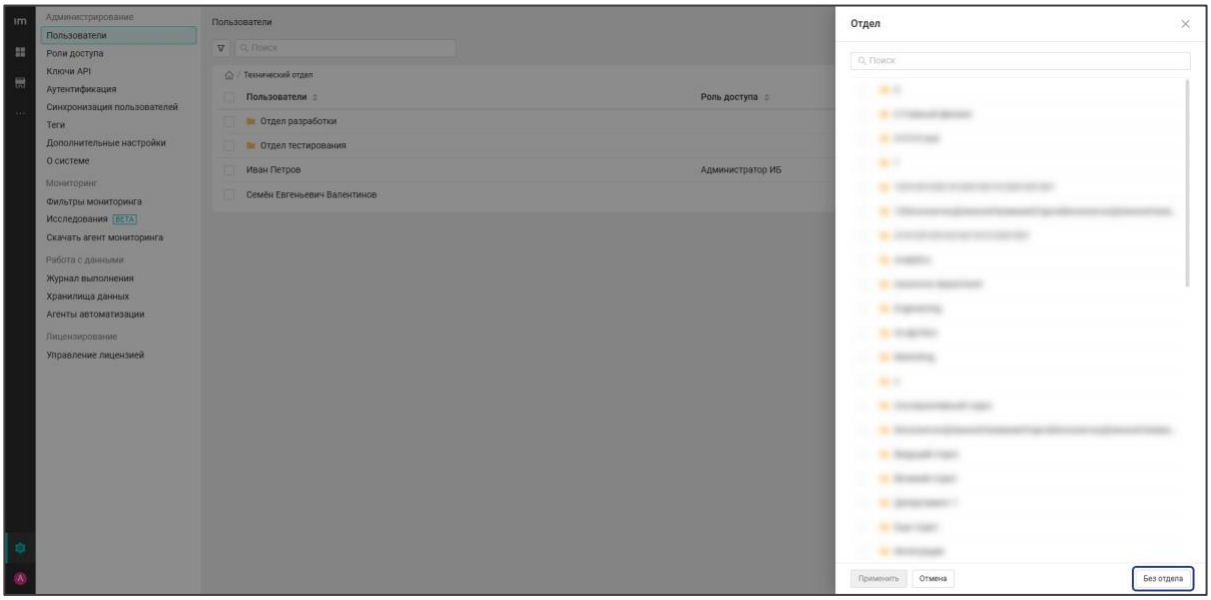
Профиль пользователя

Для того, чтобы перейти в профиль пользователя, вы также можете кликнуть по имени пользователя или нажать значок контекстного меню напротив имени и выбрать **Редактировать** в появившемся списке.



Также в контекстном меню доступны еще 2 действия: переместить и удалить.

Чтобы переместить пользователя в другой отдел, нажмите кнопку **Переместить**. В открывшемся списке выберите отдел для перемещения выбранного пользователя. Если помещать пользователя в какой-либо отдел не требуется, нажмите **Без отдела** в правом нижнем углу боковой панели.



Чтобы сохранить изменения, нажмите **Применить**. Для отмены действия нажмите кнопку **Отмена**.

Чтобы удалить пользователя, нажмите кнопку **Удалить** и подтвердите удаление.

В профиле для редактирования доступно 5 вкладок:

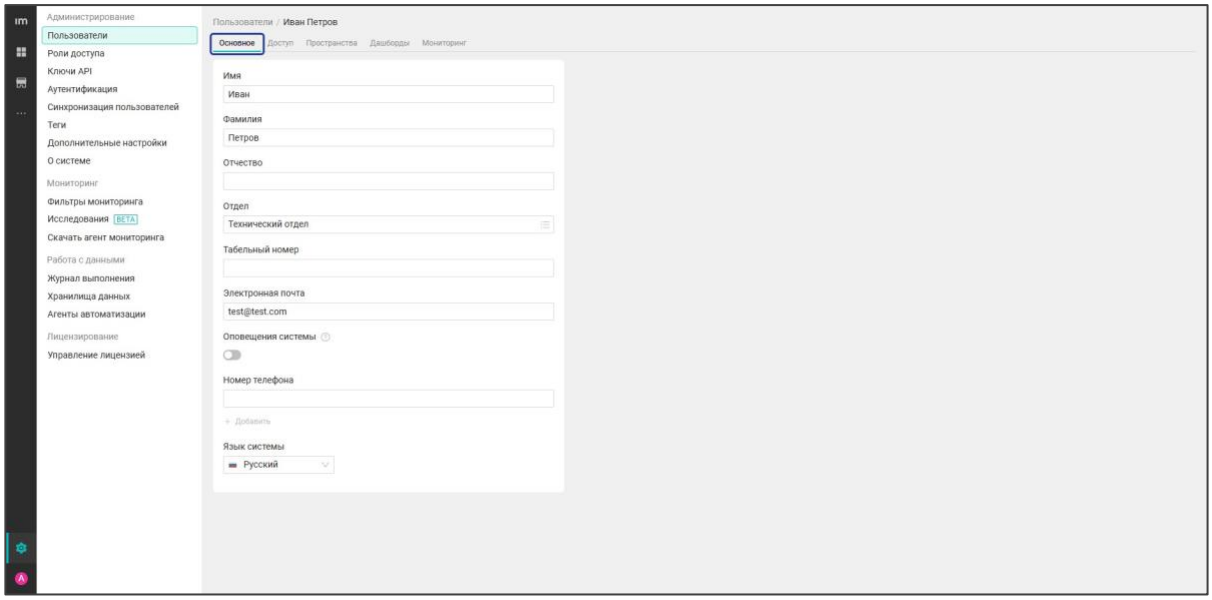
- *Основное*
- *Доступ*
- *Пространства*
- *Дашборды*
- *Мониторинг*

Основное

Во вкладке представлены общие настройки профиля.

В этой вкладке вы можете:

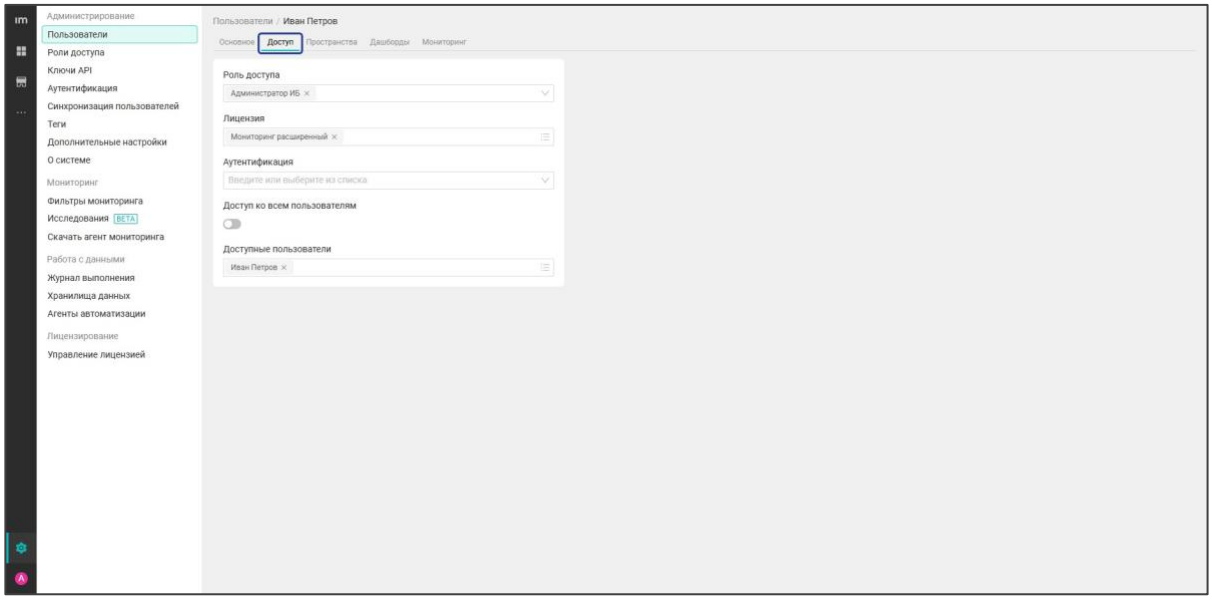
- Изменить имя сотрудника
- Изменить фамилию
- Изменить отчество
- Назначить отдел
- Добавить табельный номер
- Добавить электронную почту. На нее после отправки приглашения будут высланы данные для входа в систему
- Добавить номер(а) телефона(ов)
- Изменить язык системы пользователю (русский/английский)



Нажмите кнопку **Сохранить** в нижней части страницы, чтобы сохранить внесенные изменения.

Доступ

Вкладка отвечает за настройки доступа пользователя в систему.



В этой вкладке вы можете:

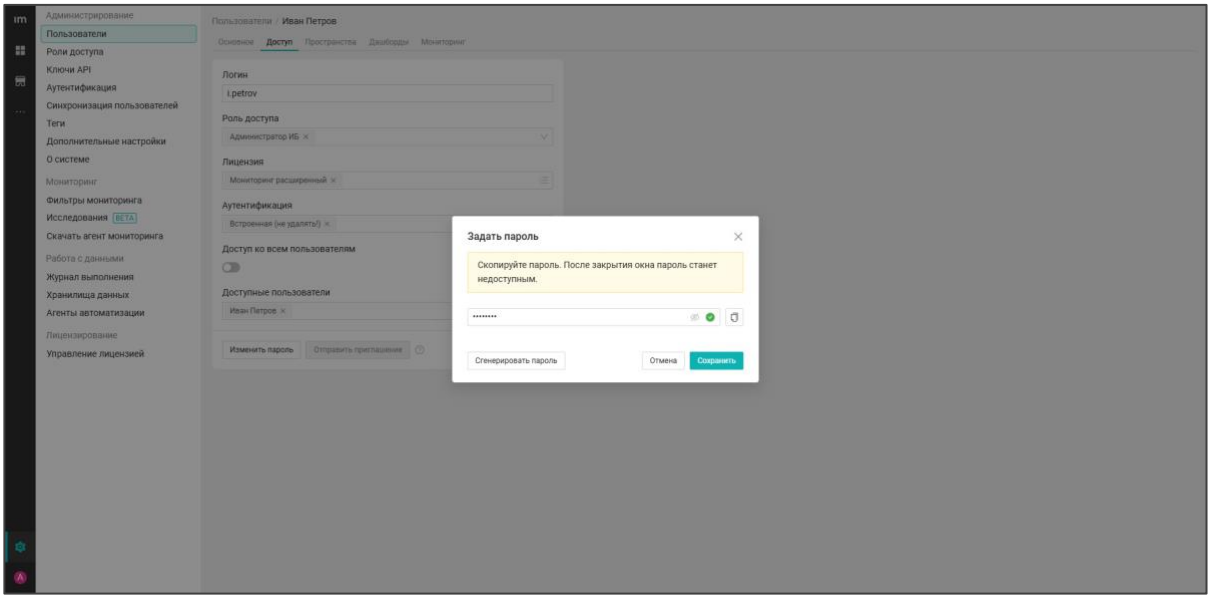
- Назначить роли доступа
- Назначить тип лицензии
- Добавить тип аутентификации в системе
- Включить или отключить доступ ко всем пользователям
- Назначить пользователей (отделы), к которым нужен доступ
- Задать пароль (доступно в случае, если пользователю назначен встроенный тип аутентификации)
- Отправить приглашение (доступно в случае, если пользователю назначен встроенный тип аутентификации)

Задать пароль

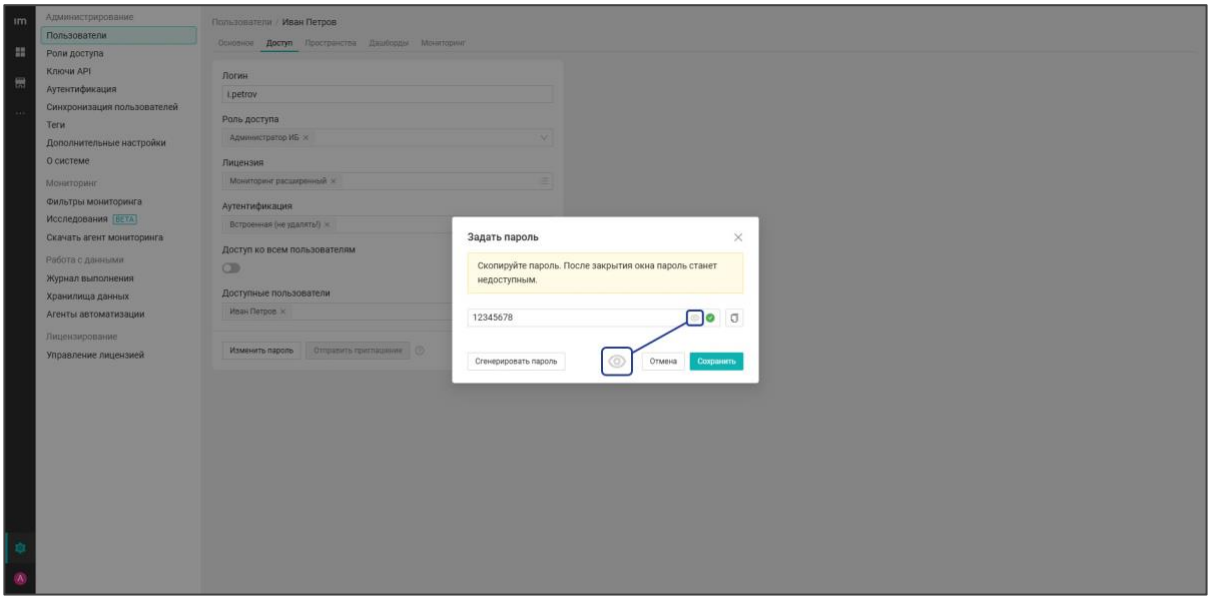
Пароль задается для локального входа в систему (аутентификация с типом *Встроенная*).

1. Пароль можно придумать самостоятельно либо использовать функцию **Сгенерировать пароль**.

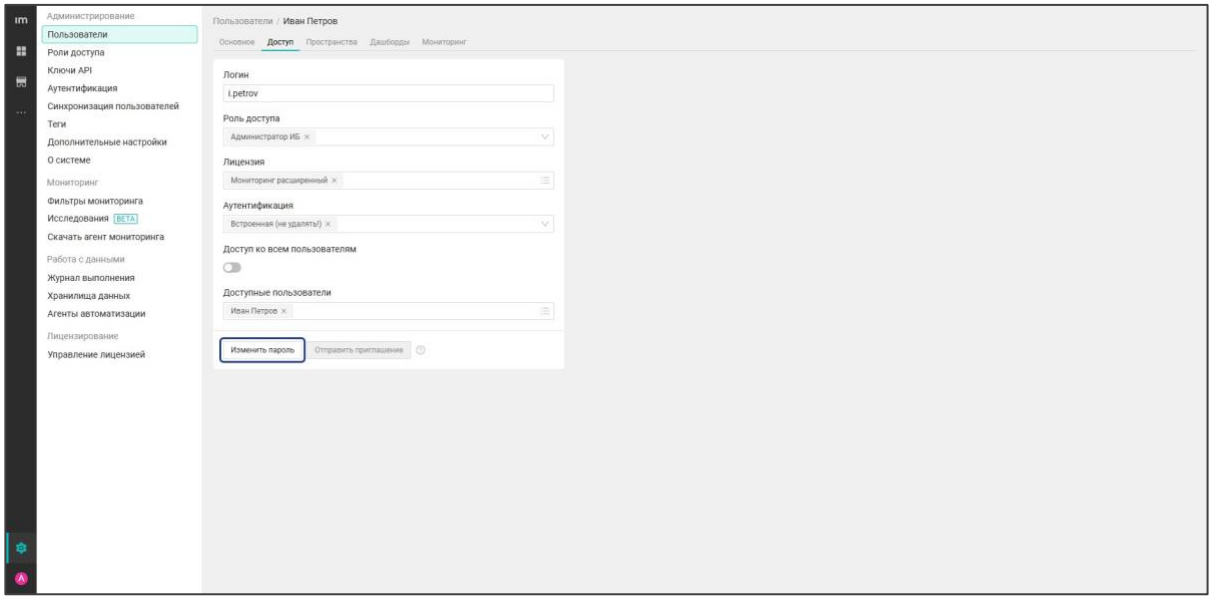
2. После того, как пароль задан, его можно скопировать, нажав соответствующую кнопку. Это единственное место, откуда можно скопировать значение. После закрытия окна пароль больше нельзя посмотреть, но можно задать его заново.



3. По умолчанию пароль скрыт. Чтобы его посмотреть, нажмите кнопку **Показать пароль**. Если пользователь повторно использует генерацию пароля, то новый пароль не скрывается.



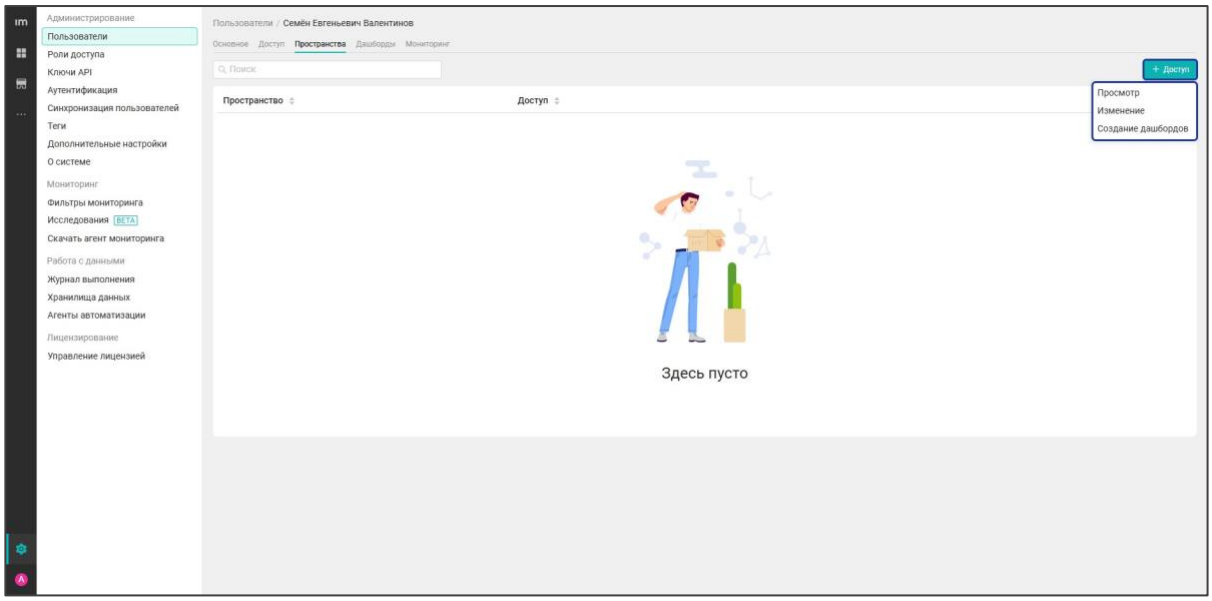
4. Если пароль уже задан, то вместо **Задать пароль** появляется кнопка **Изменить пароль**.



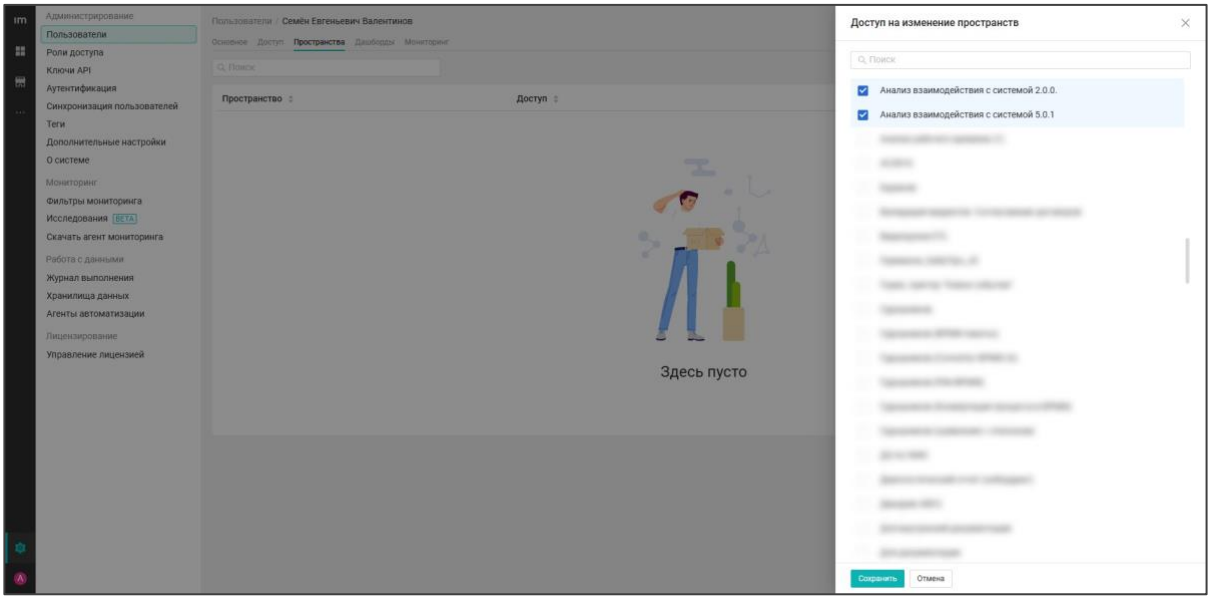
Когда пользователь переходит в свой профиль во вкладку *Доступ*, поле изменения пароля скрывается. Самостоятельно изменить пароль пользователь может только в своем профиле.

Пространства

Вкладка *Пространства* является расширением базового профиля пользователя при подключенном модуле «Аналитика». Во вкладке можно настроить доступ пользователя к пространствам. Чтобы добавить доступ к пространству, нажмите кнопку **+ Добавить** и выберите: **Просмотр**, **Изменение** или **Создание дашбордов**.

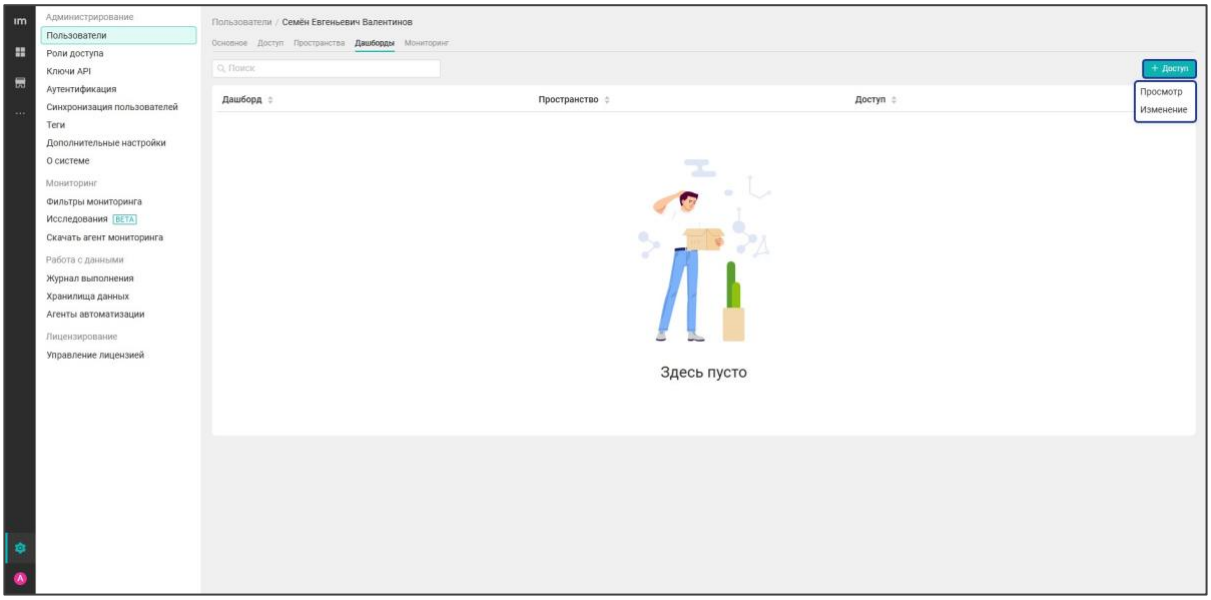


Чтобы подтвердить изменения, нажмите **Сохранить**. Кнопка активна только после внесения изменений.

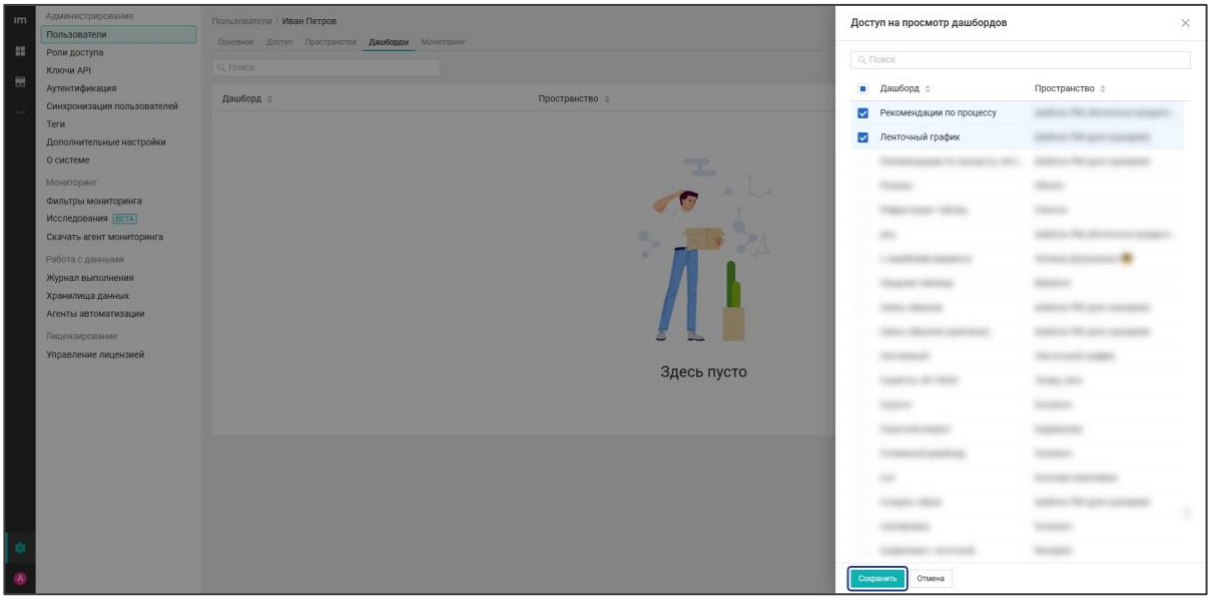


Дашборды

Во вкладке *Дашборды* можно настроить доступ пользователя к дашбордам. Нажмите + **Доступ** и выберите: **Изменение** или **Просмотр**.

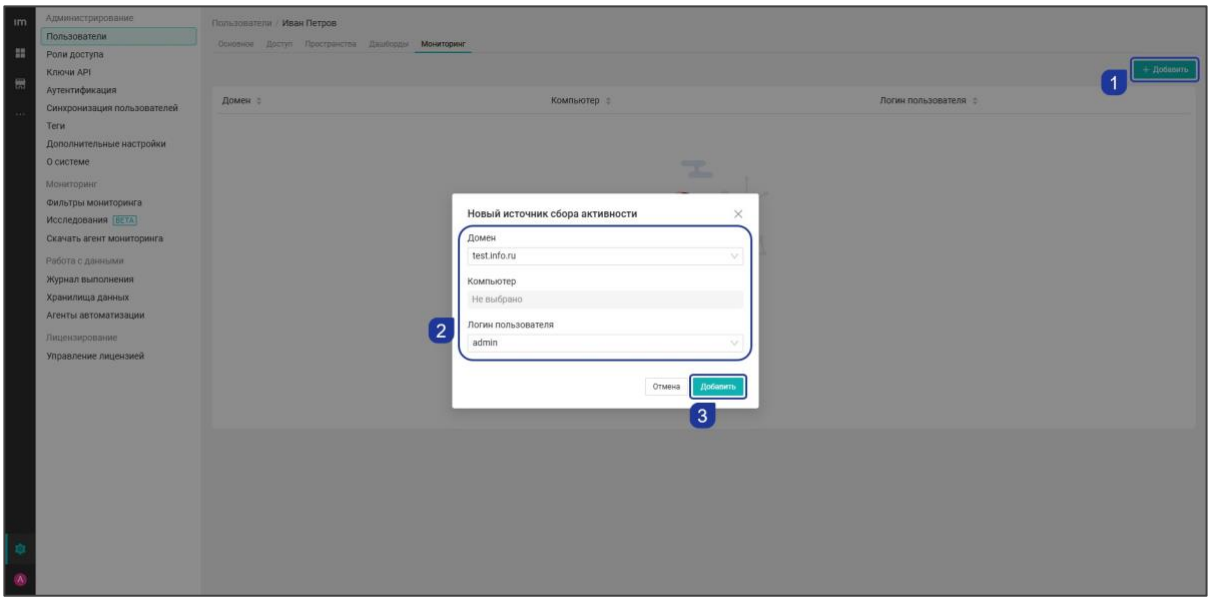


Выберите дашборды и нажмите **Сохранить**.



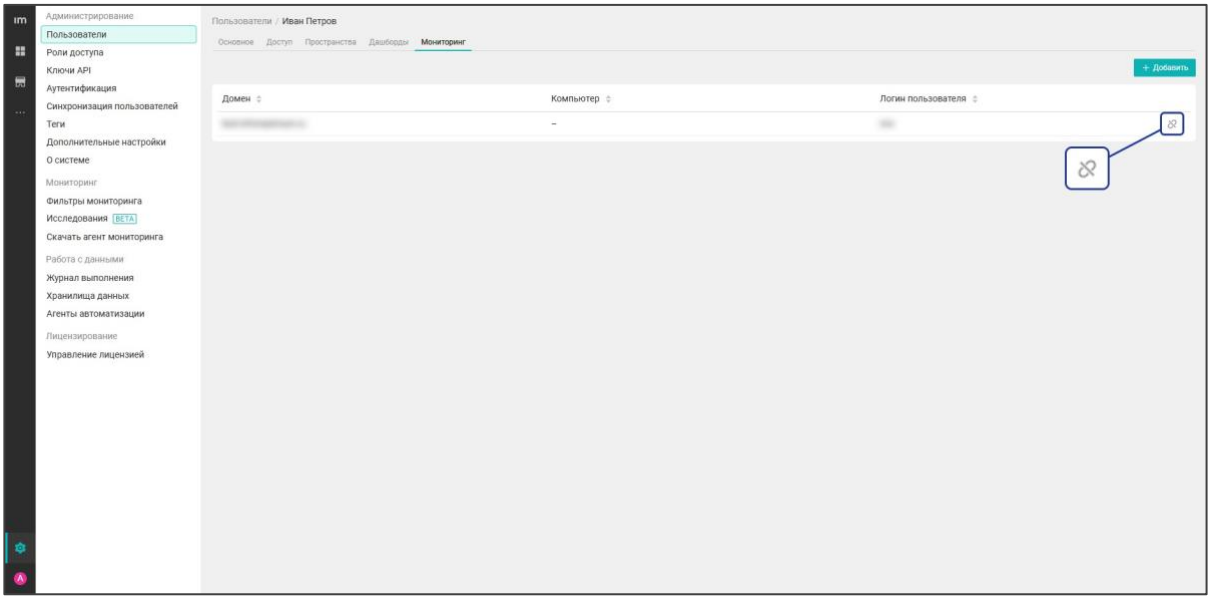
Мониторинг

Во вкладке *Мониторинг* можно добавить новый источник сбора активности для пользователя. Для этого нажмите кнопку **+ Добавить** и выберите домен или компьютер, затем можно указать логин пользователя. Кликните **Добавить**.

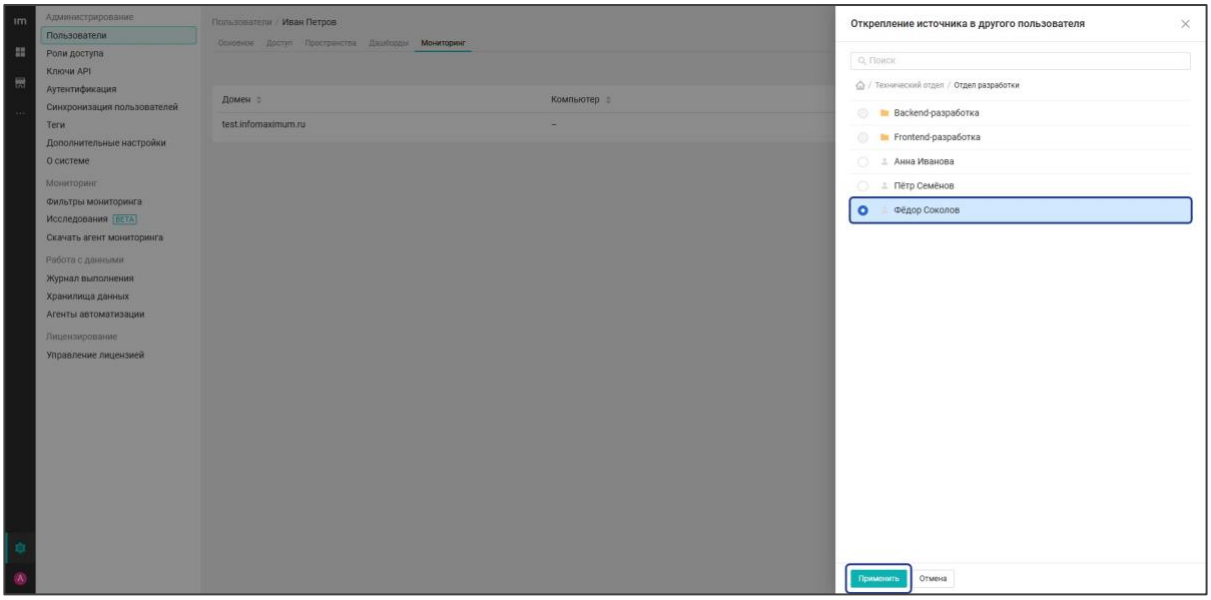


Вы можете открепить источник активности в другого пользователя. Для этого:

1. Нажмите на иконку открепления напротив источника активности.



2. Выберите пользователя, в профиль которого вы хотите открепить источник активности, и нажмите **Применить**.



Чтобы не назначать параметры отдельно для каждого пользователя, к пользователям и отделам можно применять массовые действия.

Отделы

Добавление новых отделов доступно пользователям, имеющим привилегию *Пользователи и отделы* с операциями доступа **R**, **W** и **C**. Также пользователю должен быть доступен хотя бы один отдел (настраивается в профиле пользователя). Добавить отдел можно только в те отделы, к которым у него есть доступ. Для добавления отдела в корневой отдел необходим доступ ко всем отделам.

Вы можете добавить новый отдел в настройках системы. Для этого:

1. Перейдите во вкладку *Настройки*.

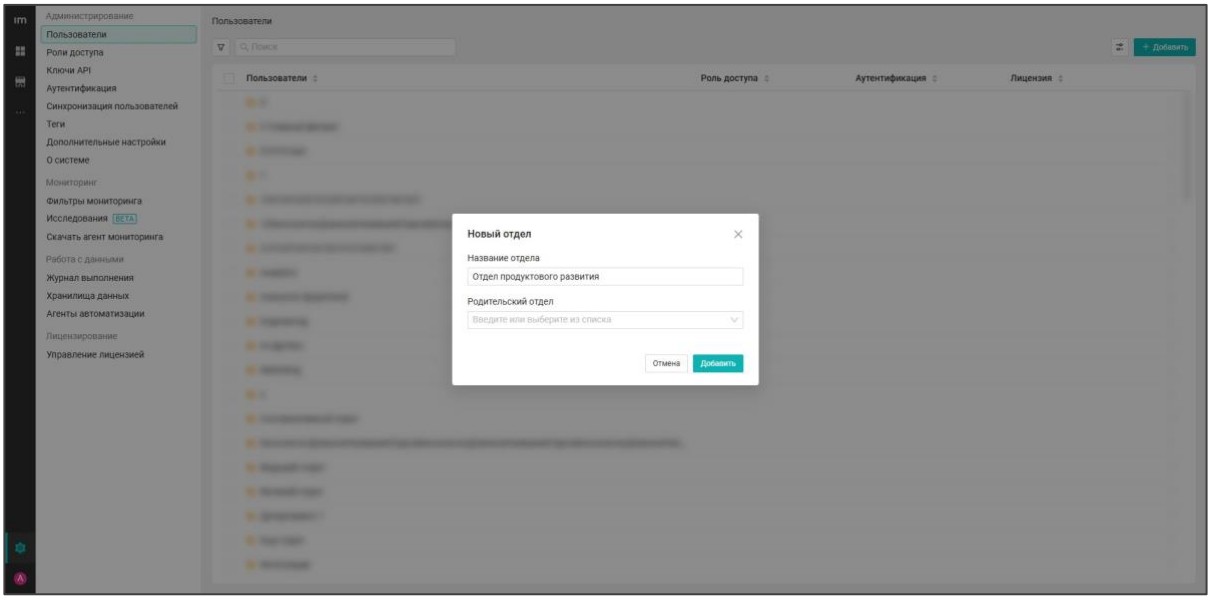
2. Выберите пункт **Пользователи**.

3. На открывшейся странице со списком пользователей и отделов нажмите кнопку + **Добавить**.

4. Выберите пункт **Отдел**.

5. В появившемся окне введите:

- Название отдела
- Родительский отдел: если конкретный отдел не указан, новый отдел будет сохранен в текущем открытом отделе. Если текущий отдел недоступен, новый отдел будет добавлен в первый доступный отдел в иерархии



6. Чтобы добавить новый отдел, нажмите **Добавить**.

7. Чтобы отменить добавление отдела, нажмите **Отмена**.

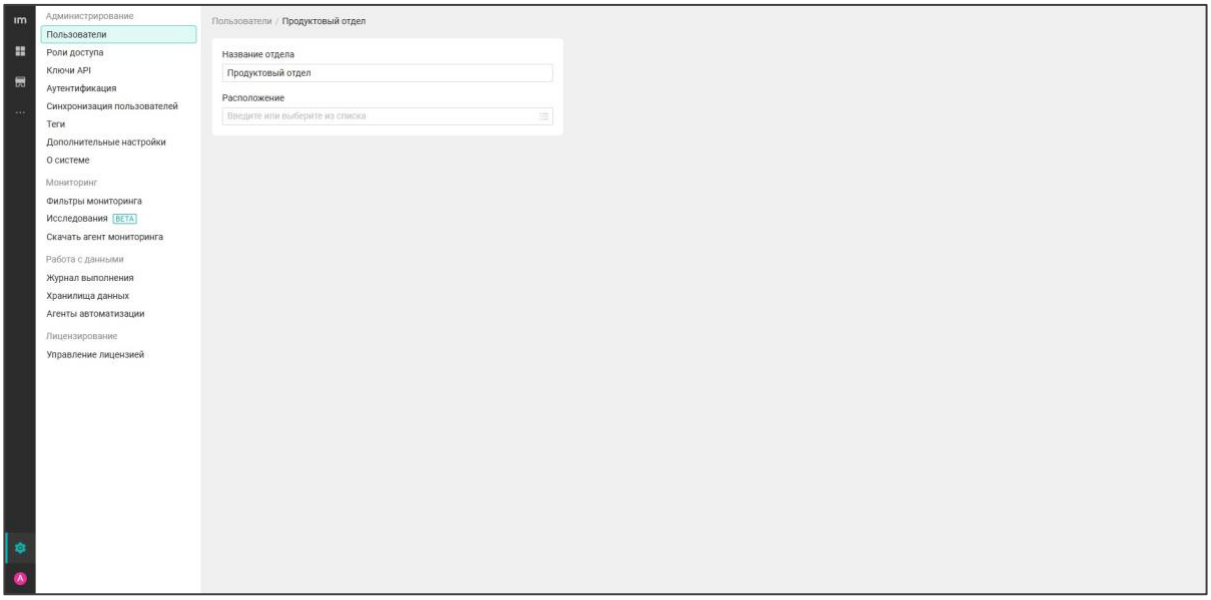
Система сообщит, что отдел создан. После создания нового отдела вы можете перейти в его профиль для редактирования.

Профиль отдела

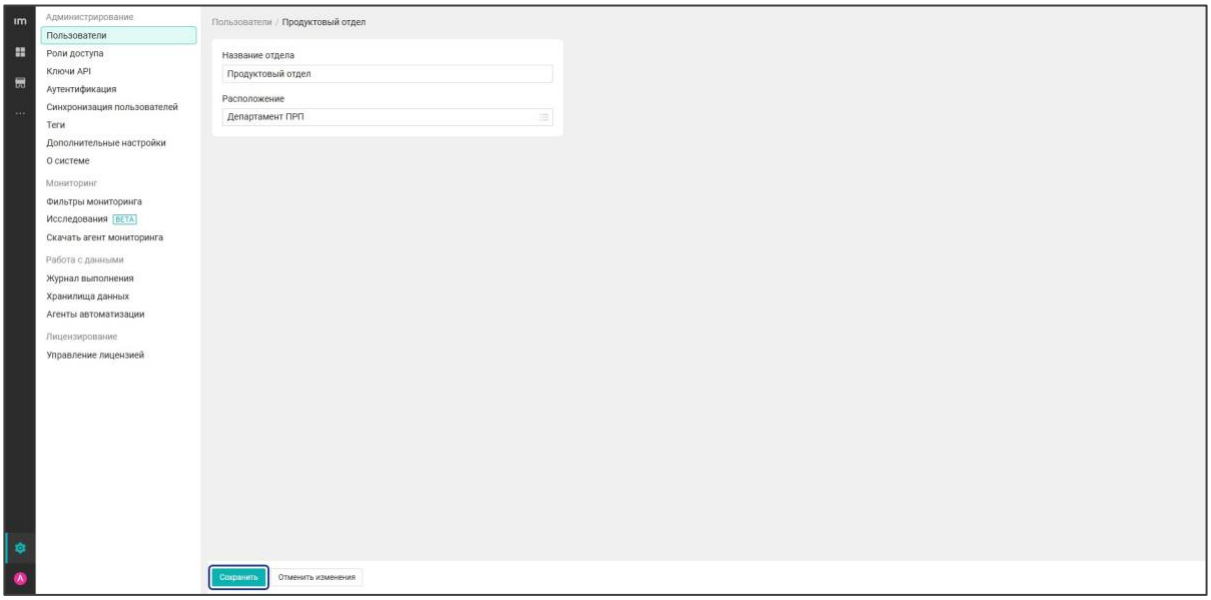
В профиль отдела также можно перейти через контекстное меню, расположенное напротив интересующего отдела. В открывшемся меню выберите **Редактировать**.

1. В профиле отдела вы можете изменить:

- Название отдела
- Расположение отдела

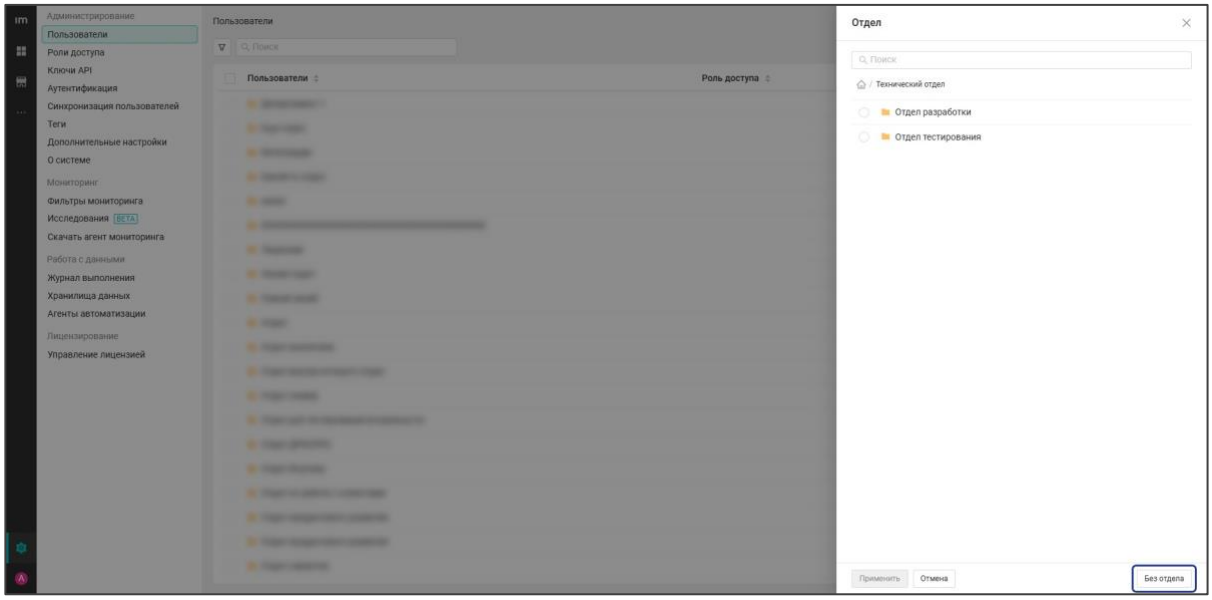


2. Чтобы применить изменения, нажмите кнопку **Сохранить**.



Также в контекстном меню доступны еще 2 действия: переместить и удалить.

Чтобы переместить отдел, нажмите кнопку **Переместить**. В открывшемся списке выберите отдел для перемещения. Чтобы переместить выбранный отдел в корневой отдел, нажмите **Без отдела** в правом нижнем углу боковой панели.



Чтобы сохранить изменения, нажмите **Применить**. Для отмены действия нажмите кнопку **Отмена**.

Чтобы удалить отдел, нажмите кнопку **Удалить** и подтвердите удаление.

Доступ к организационной структуре

Параметр open_org_structure в конфигурационном файле com.infomaximum.subsystem.core.json определяет режим работы организационной структуры и позволяет администратору выбрать один из двух вариантов:

- Открытый режим
- Закрытый режим

Этот параметр помогает администратору определить, будет ли структура компании полностью видимой для удобства взаимодействия или скрытой для защиты конфиденциальных данных.

Режим работы организационной структуры задается на уровне системных настроек. Чтобы включить открытый или закрытый режим, необходимо изменить параметр open_org_structure в конфигурационном файле com.infomaximum.subsystem.core.json.

По умолчанию параметр open_org_structure принимает значение true. Если вам необходимо настроить закрытый режим организационной структуры, замените значение параметра на false.

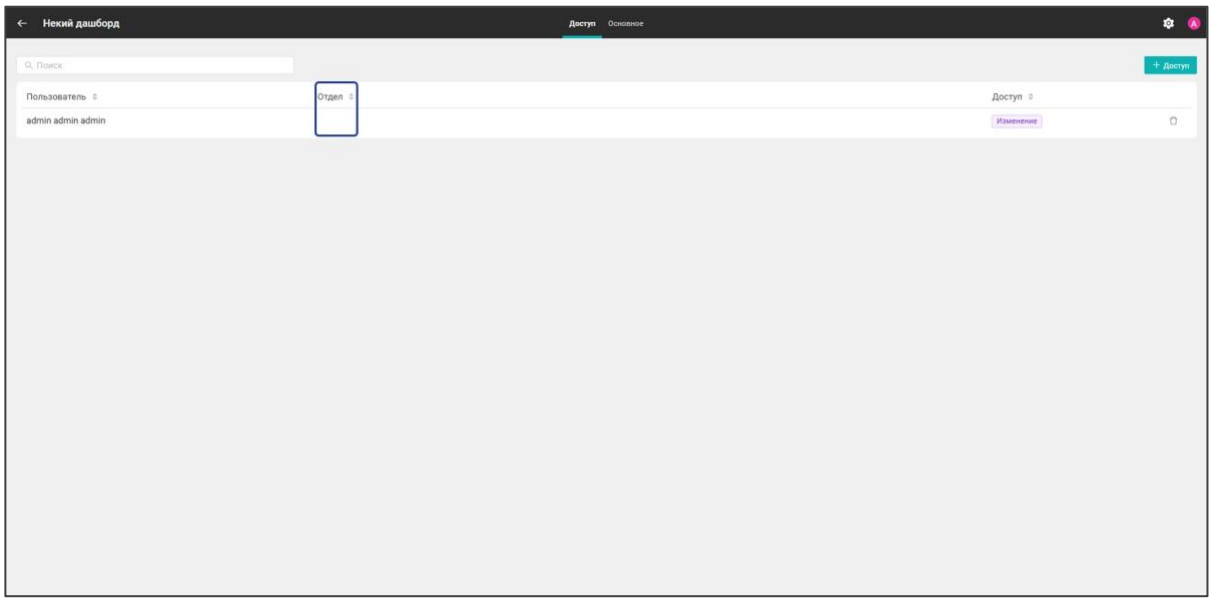
Открытый режим

Если у параметра open_org_structure установлено значение true, организационная структура доступна в открытом режиме.

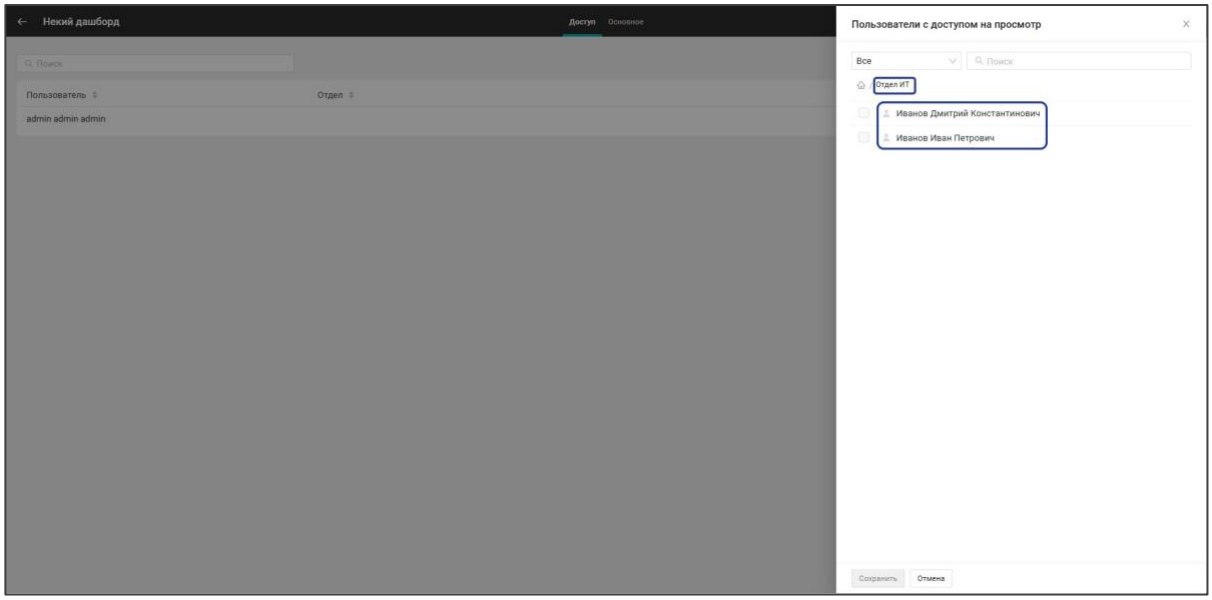
В этом режиме:

- Доступна информация по отделам:
  - При добавлении пользователей в исследование
  - В настройках доступа к пространствам

- В настройках доступа к дашбордам



- В боковом меню доступен поиск по дереву сотрудников и отделов:
  - При добавлении пользователей в исследование
  - При назначении доступа к пространствам
  - При назначении доступа к дашбордам

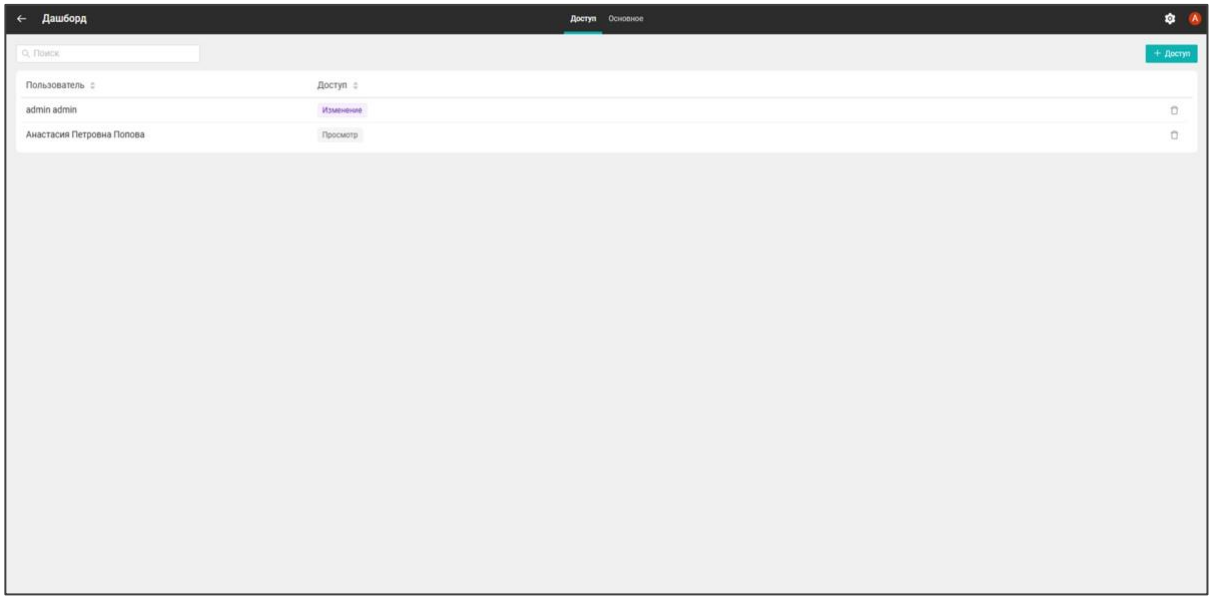


Закрытый режим

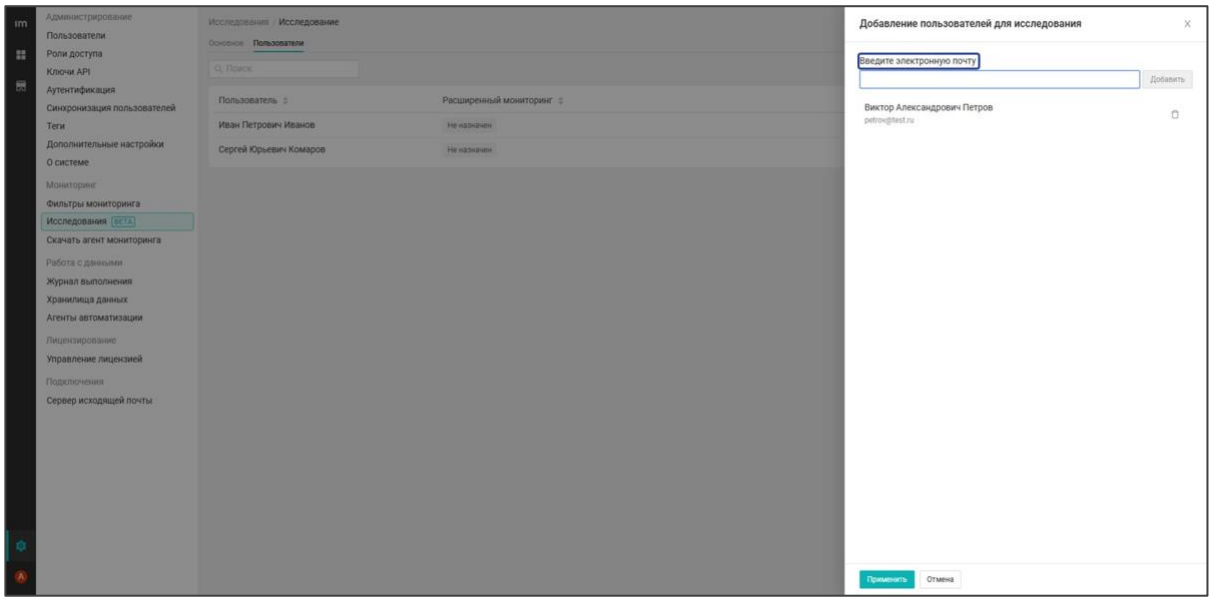
Если у параметра open_org_structure установлено значение false, организационная структура доступна только в пределах прав доступа пользователя.

В этом режиме:

- Недоступна информация по отделам:
  - При добавлении пользователей в исследование
  - В настройках доступа к пространствам
  - В настройках доступа к дашбордам



- В боковом меню доступен поиск только по электронной почте пользователя:
    - При назначении доступа к пространствам
    - При назначении доступа к дашбордам
    - При добавлении пользователей в исследование



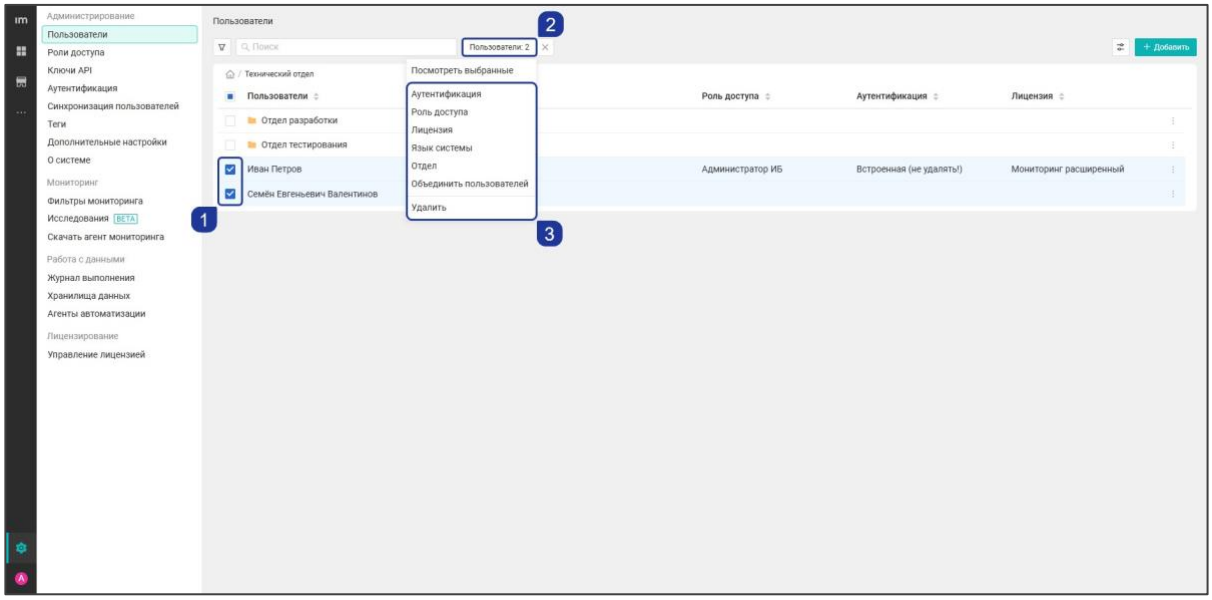
Массовые действия

В разделе *Администрирование* на странице *Пользователи* можно применять массовые действия к пользователям и отделам. Массовые действия позволяют назначать параметры пользователям без перехода в профиль каждого из них.

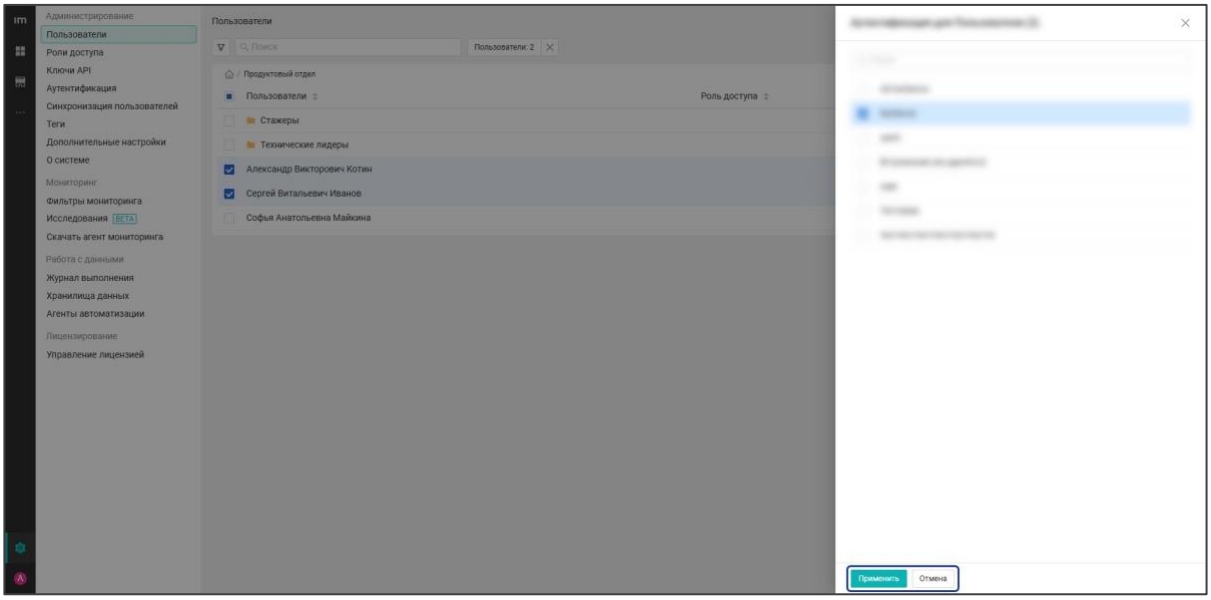
Доступно применение следующих массовых действий:

- Аутентификация
- Роль доступа
- Лицензия
- Язык системы
- Отдел
- Объединение пользователей
- Удаление пользователей и отделов

Чтобы применить действие, выберите пользователей и нажмите кнопку с выделенными пользователями/отделами. В раскрывающемся списке кликните по интересующему действию.



В появившемся боковом окне задайте необходимые настройки. Например, аутентификацию, роль доступа или другое. Нажмите кнопку **Применить**. Для отмены действия нажмите кнопку **Отмена**.



Примечание. Применить действие можно, если выбран минимум 1 пользователь или отдел.

Для переназначения массового действия:

1. Выберите пользователей или отделы, для которых необходимо переназначить действие.

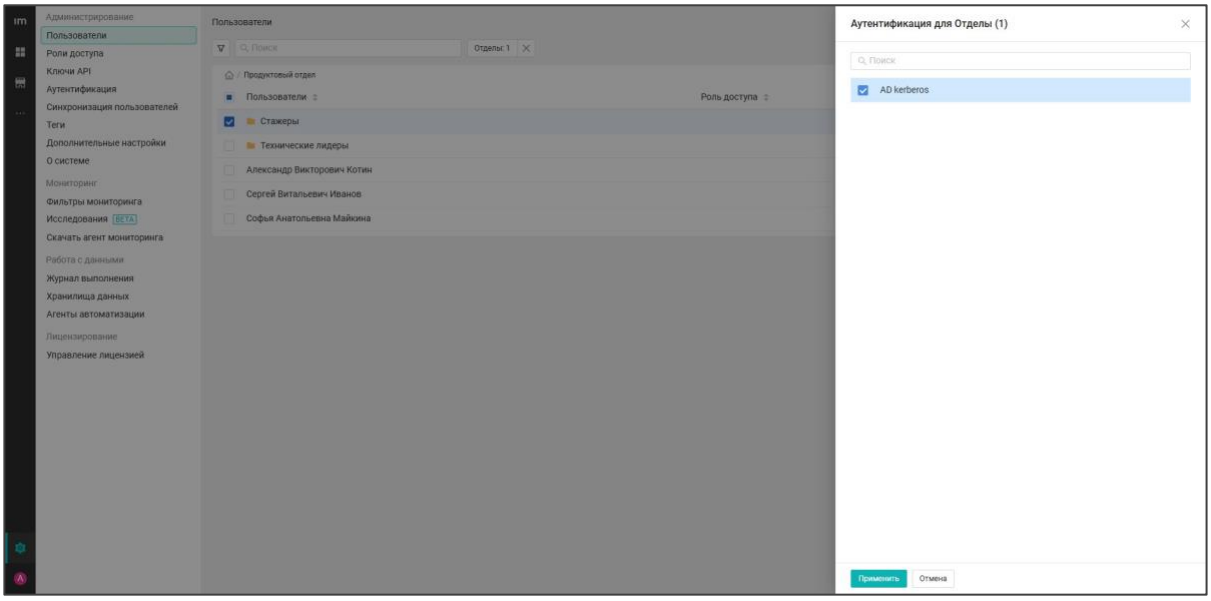
2. Нажмите кнопку с выделенными пользователями/отделами, затем выберите необходимое действие.

3. В появившемся боковом окне уберите галочку с выбранного ранее варианта и отметьте интересующий вариант.

4. Нажмите кнопку **Применить**.

Аутентификация

Для назначения доступны аутентификации, созданные в системе. Аутентификации можно назначать одновременно.

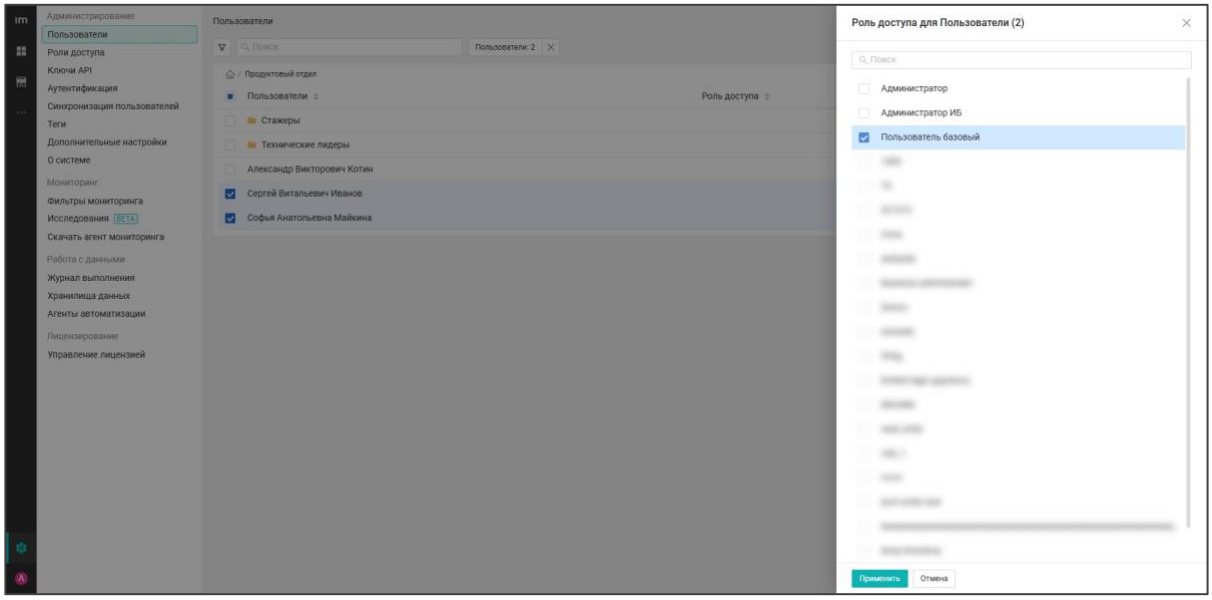


Назначенная пользователям или отделам аутентификация в боковом окне отмечена галочкой. При выборе еще одной аутентификации она добавится к назначенной. Чтобы снять назначенную аутентификацию, уберите с нее галочку и нажмите **Применить**.

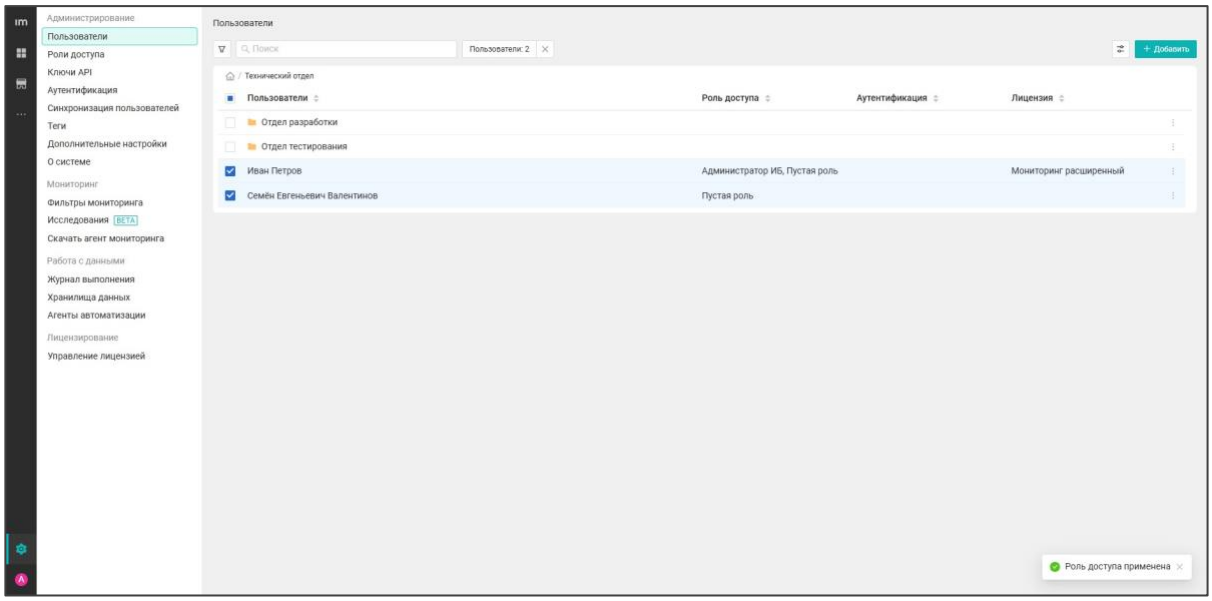
Важно. Если у пользователя отсутствует аутентификация, он не сможет войти в систему.

Роль доступа

Для назначения доступны роли доступа, созданные в системе, а также предустановленные роли доступа: **Администратор** и **Администратор ИБ**. Назначать роли доступа можно одновременно.



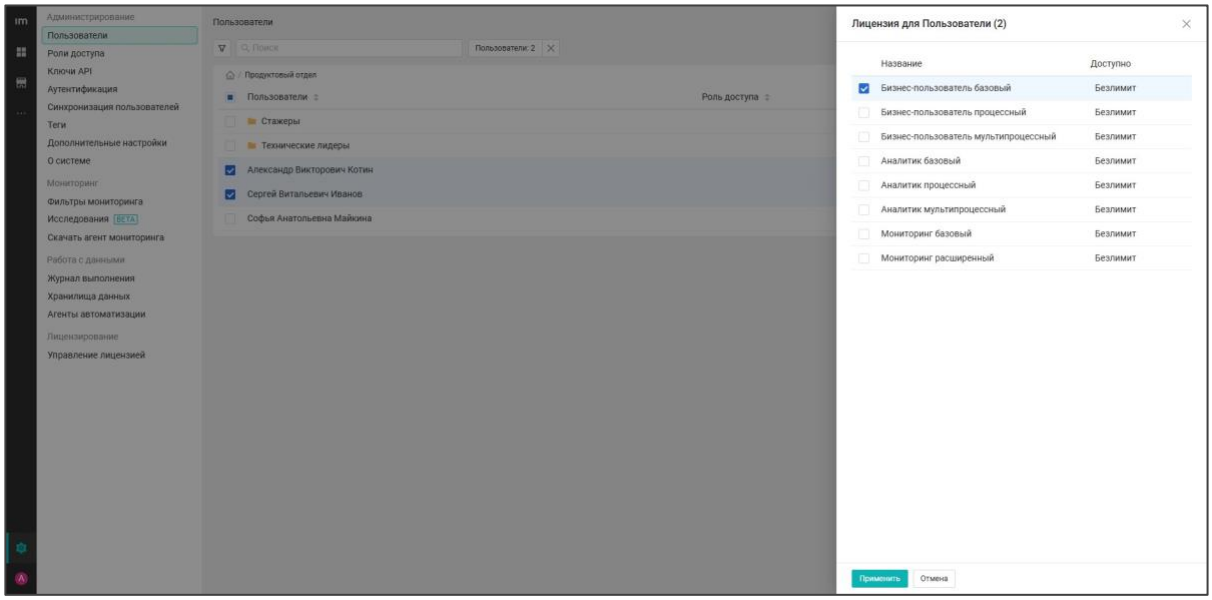
Назначенная пользователям или отделам роль доступа в боковом окне отмечена галочкой. При выборе еще одной роли доступа она добавится к назначенной. Чтобы снять назначенную роль доступа, уберите с нее галочку и нажмите **Применить**.



Лицензия

Для назначения доступны следующие типы лицензии:

- Бизнес-пользователь базовый
- Бизнес-пользователь процессный
- Бизнес-пользователь мультипроцессный
- Аналитик базовый
- Аналитик процессный
- Аналитик мультипроцессный
- Мониторинг базовый
- Мониторинг расширенный



В колонке **Доступно** указано количество лицензий каждого типа, доступное для назначения пользователям и отделам.

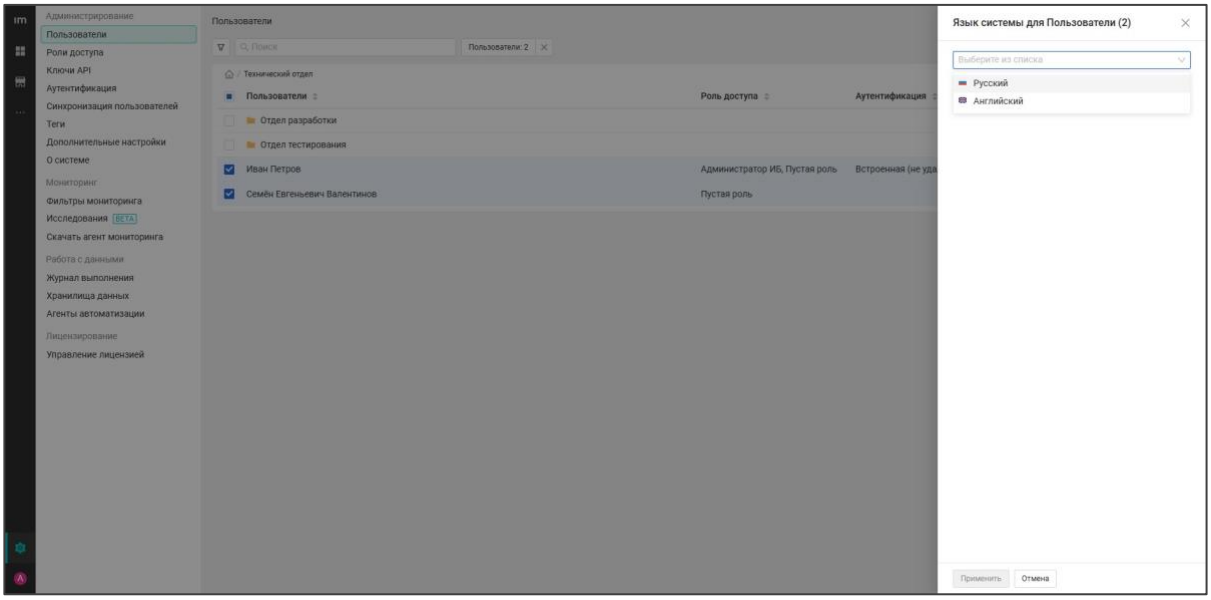
Назначенная пользователям или отделам лицензия в боковом окне отмечена галочкой. Чтобы снять назначенную лицензию, уберите с нее галочку и нажмите **Применить**. Чтобы переназначить лицензию, сначала снимите галочку с назначенной, затем отметьте другую лицензию и нажмите **Применить**.

Примечание.

- Пользователям и отделам можно назначать лицензию одновременно.
- Если при назначении лицензии двум пользователям у одного из них уже есть лицензия, спишется только 1 лицензия.

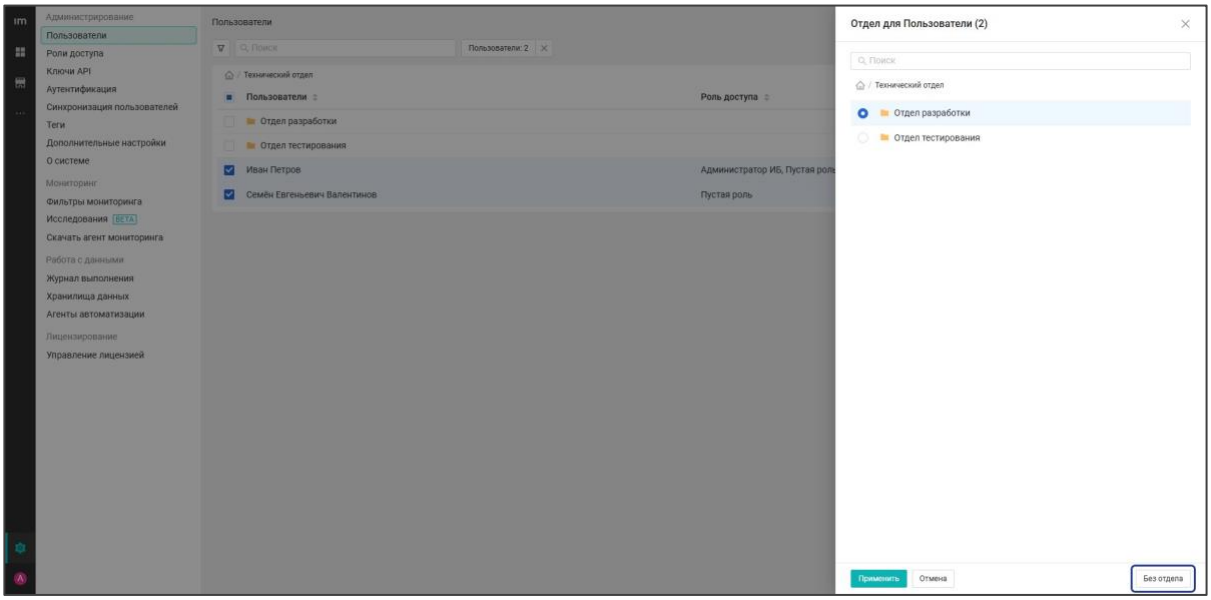
Язык системы

Для назначения доступны русский и английский языки.



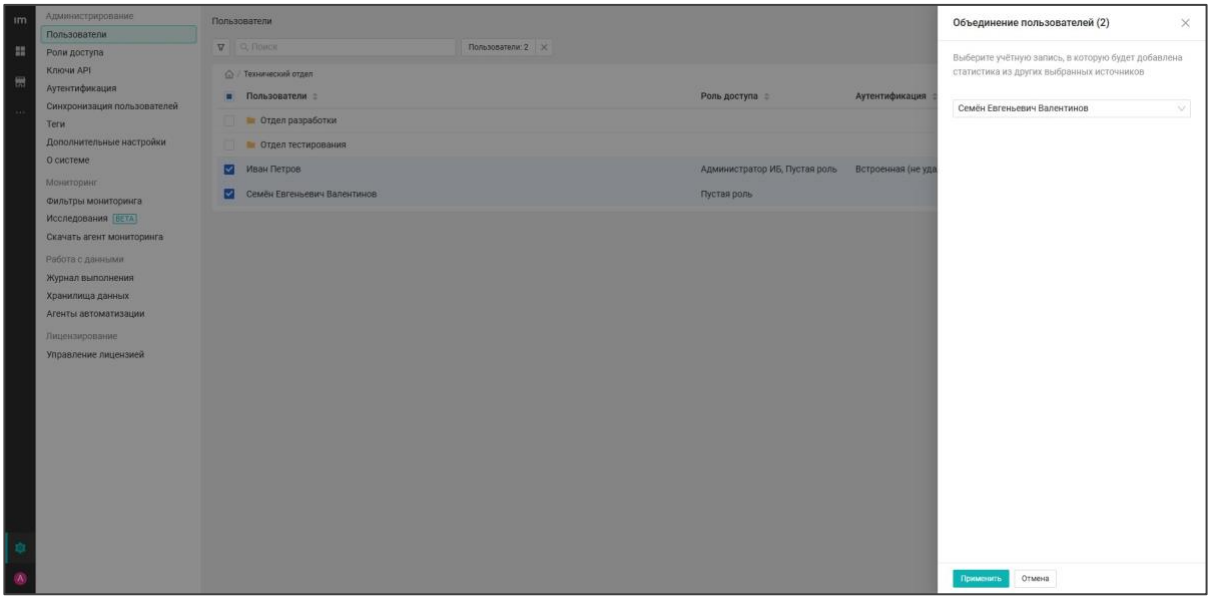
Отдел

Пользователей или отделы можно переместить в другой отдел или выбрать вариант **Без отдела**, нажав кнопку в правом нижнем углу.



Объединение пользователей

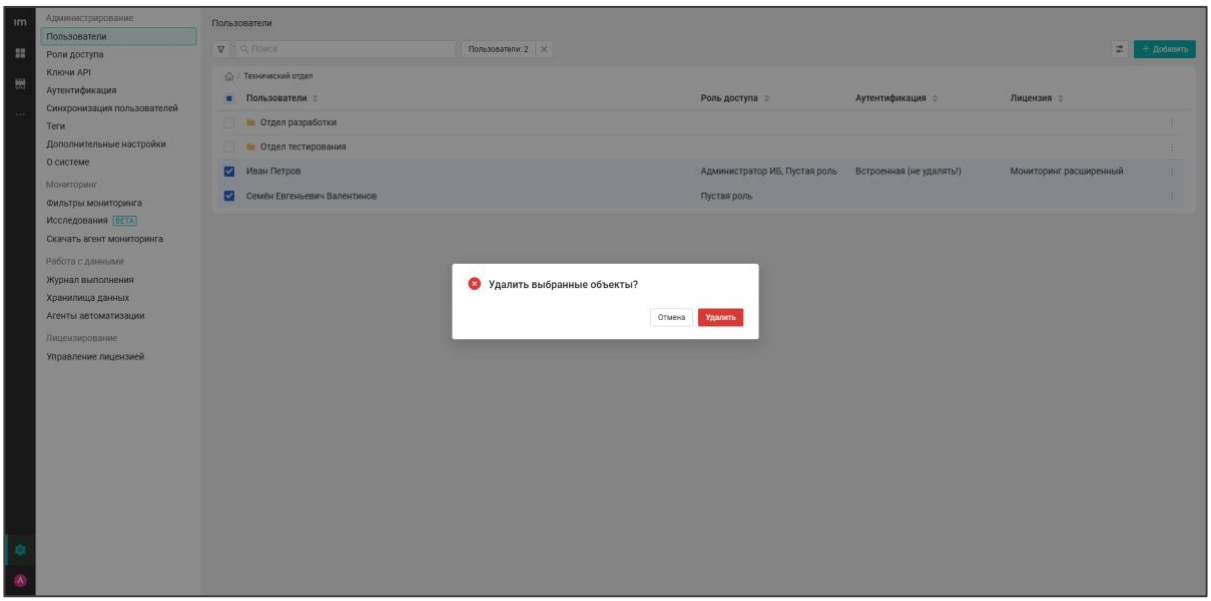
Чтобы объединить пользователей, выберите тех, которых необходимо объединить. В открывшемся окне из выпадающего списка выберите ту учетную запись, в которую нужно объединить другие.



Примечание. Функционал разъединения источников (пользователей) в настоящий момент отсутствует.

Удаление пользователей и отделов

При удалении отдела в нем не должны находиться пользователи. Чтобы удалить отдел, в котором есть пользователи или отделы, сначала переместите или удалите их.



Примечание. Удаленных из системы пользователей невозможно восстановить.

Фильтрация пользователей

В системе можно применять фильтрацию на странице Пользователи Механизм работы фильтров:

1. Примененные фильтры отображаются под кнопкой **Фильтр**.

2. Нажмите крестик в добавленном фильтре, чтобы сбросить его.

3. Независимо от модулей один и тот же фильтр можно добавить только один раз.

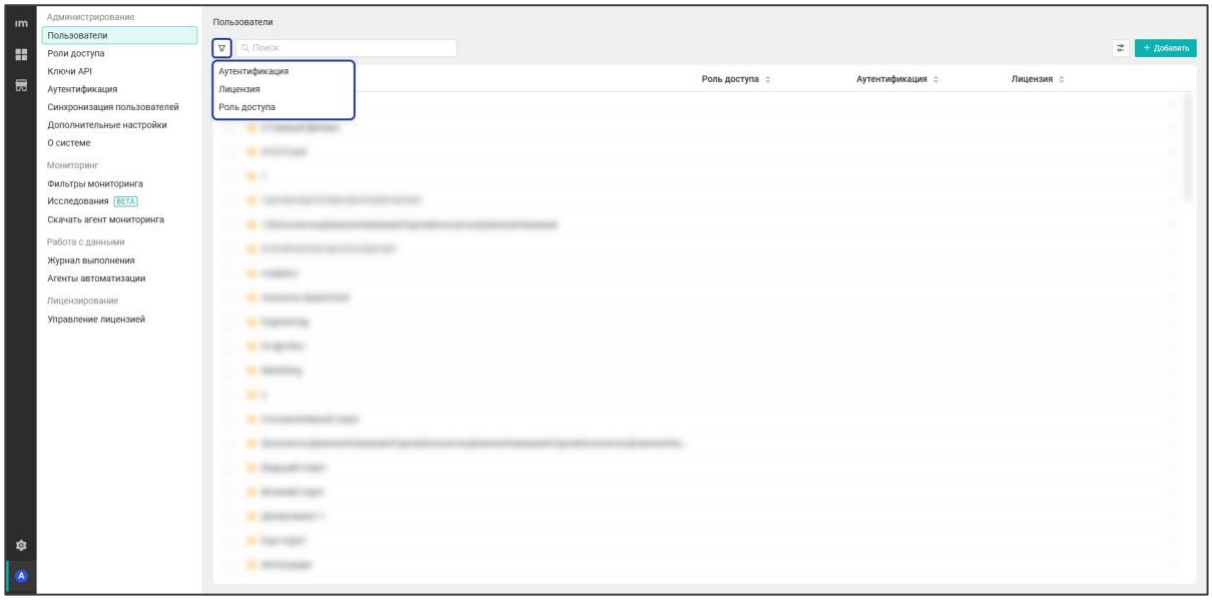
4. Если добавляется фильтр, который уже есть в списке, то он перезаписывается новым.

5. Значения фильтров сортируются по алфавиту. Сначала отображаются значения, начинающиеся на 1-9, далее – Aa-Zz, потом – Аа-Яя.

6. Изменить фильтр по конкретному параметру можно двумя способами:

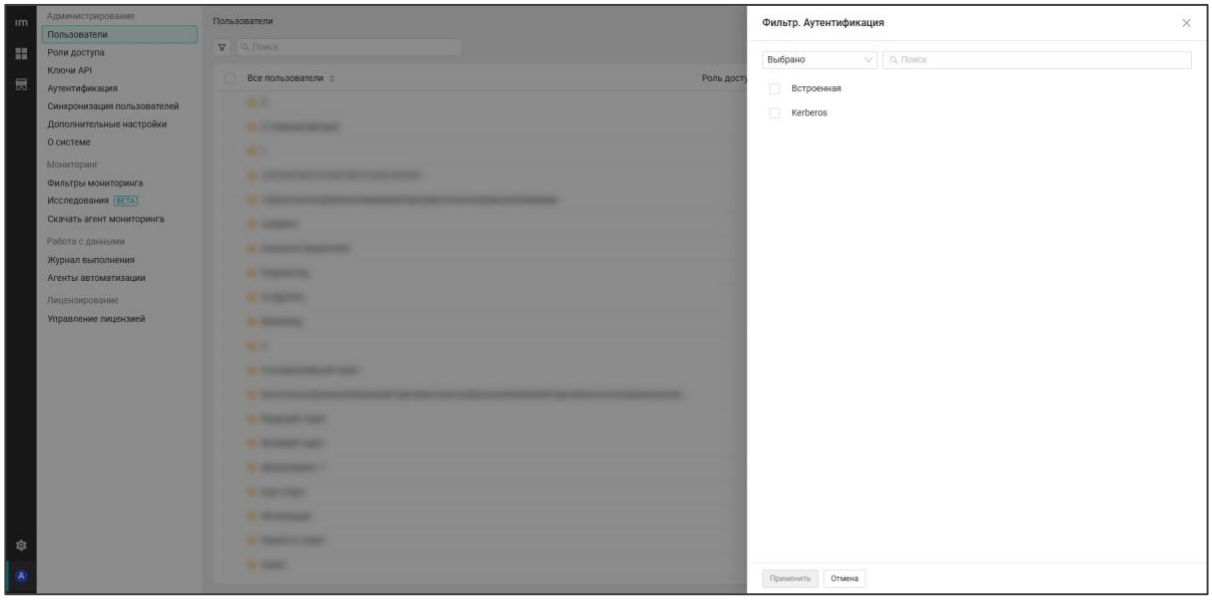
- Перейдите в фильтр с этим параметром и отредактируйте его.
- Удалите текущий фильтр и создайте новый по необходимому параметру.

Чтобы применить фильтрацию, нажмите кнопку **Фильтр**, расположенную слева от поля поиска.

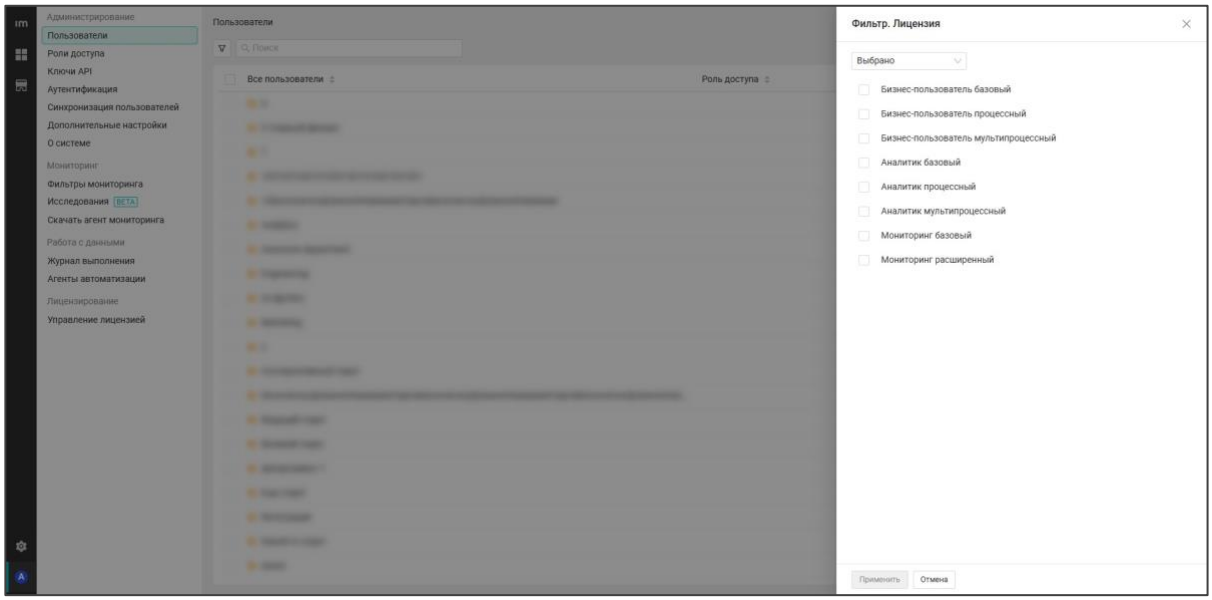


Вы можете отфильтровать список пользователей по следующим параметрам:

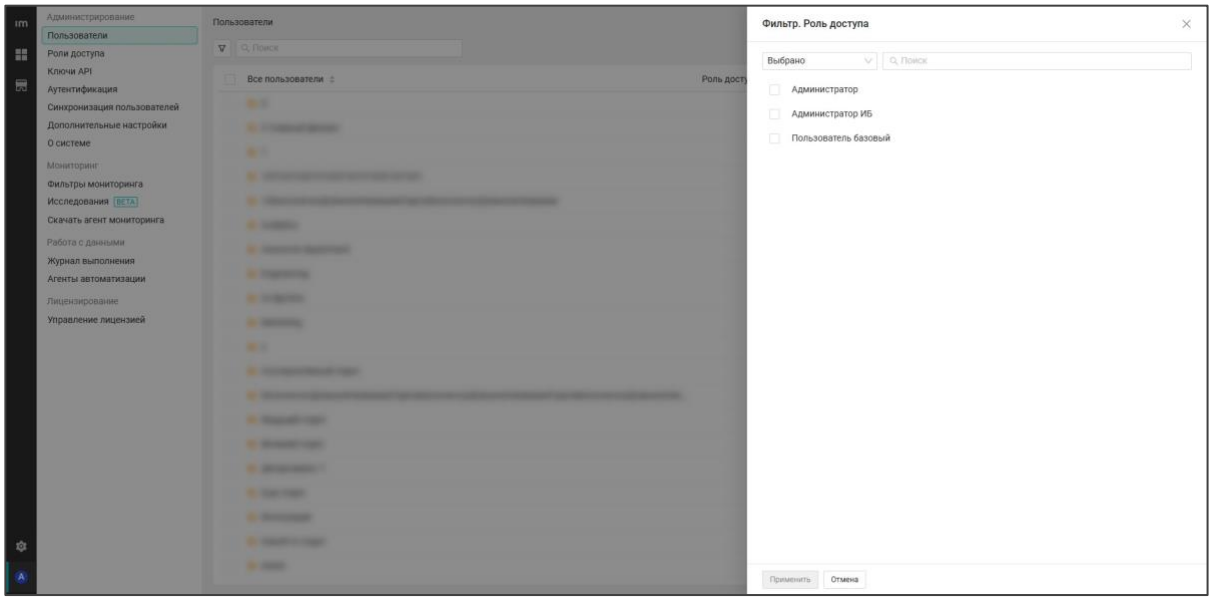
- Аутентификация — фильтрация по типу аутентификации



- Лицензия — фильтрация по типу лицензии



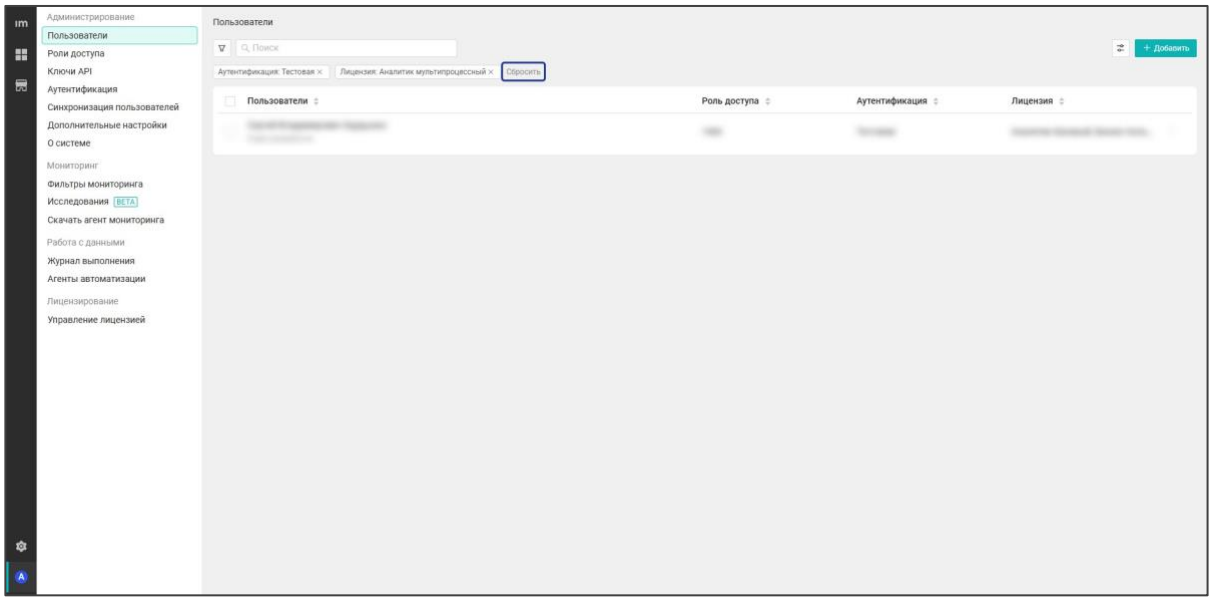
- Роль доступа — фильтрация по существующим ролям доступа



Примечание. Если у пользователя нет доступа к фильтрам из-за настроек привилегий, то иконка фильтрации для него не отображается.

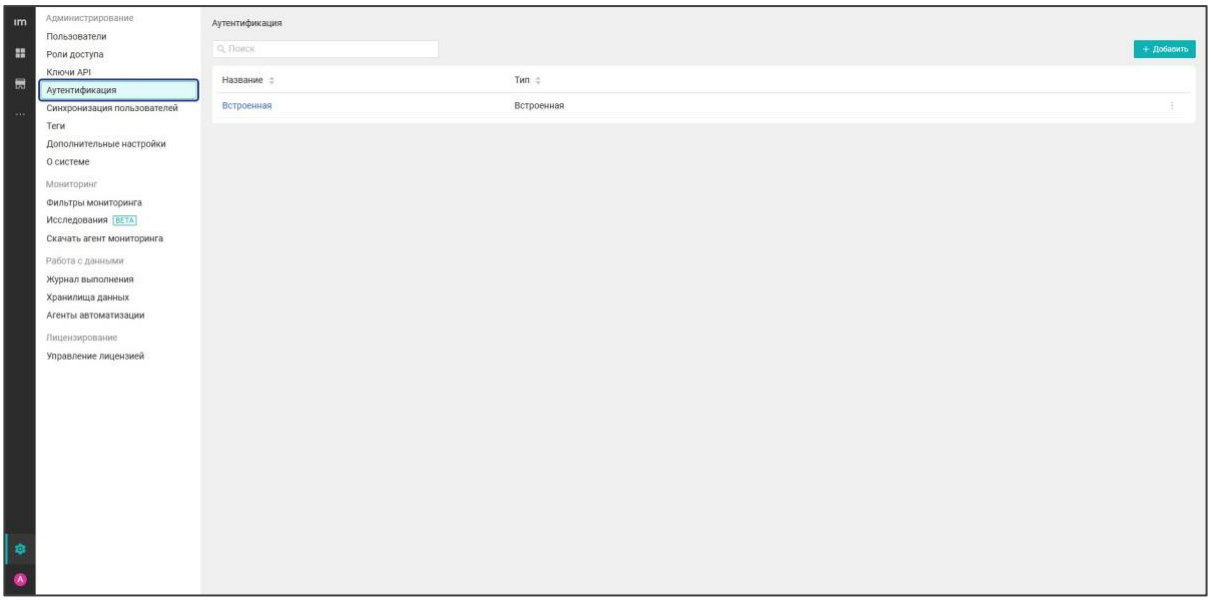
Сброс фильтрации

Чтобы очистить все фильтры, кликните **Сбросить**.



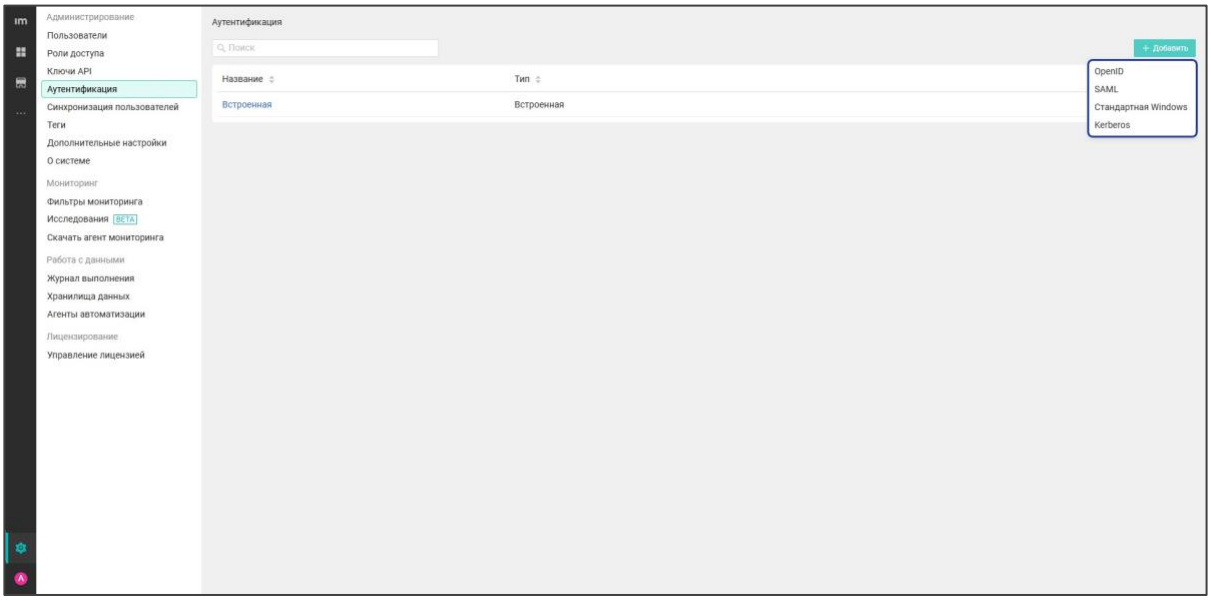
Добавление аутентификации

На странице *Аутентификация* можно настроить требования к аутентификации в системе.



По умолчанию в системе используется встроенная аутентификация. Чтобы добавить новую аутентификацию, нажмите **Добавить** и выберите ее тип:

- OpenID
- SAML
- Стандартная Windows
- Kerberos



Также в системе предусмотрен бесшовный вход. Если настроена авторизация через Active Directory и отсутствует встроенная аутентификация, происходит автоматическая авторизация пользователя даже в случае, когда сессия прервалась.

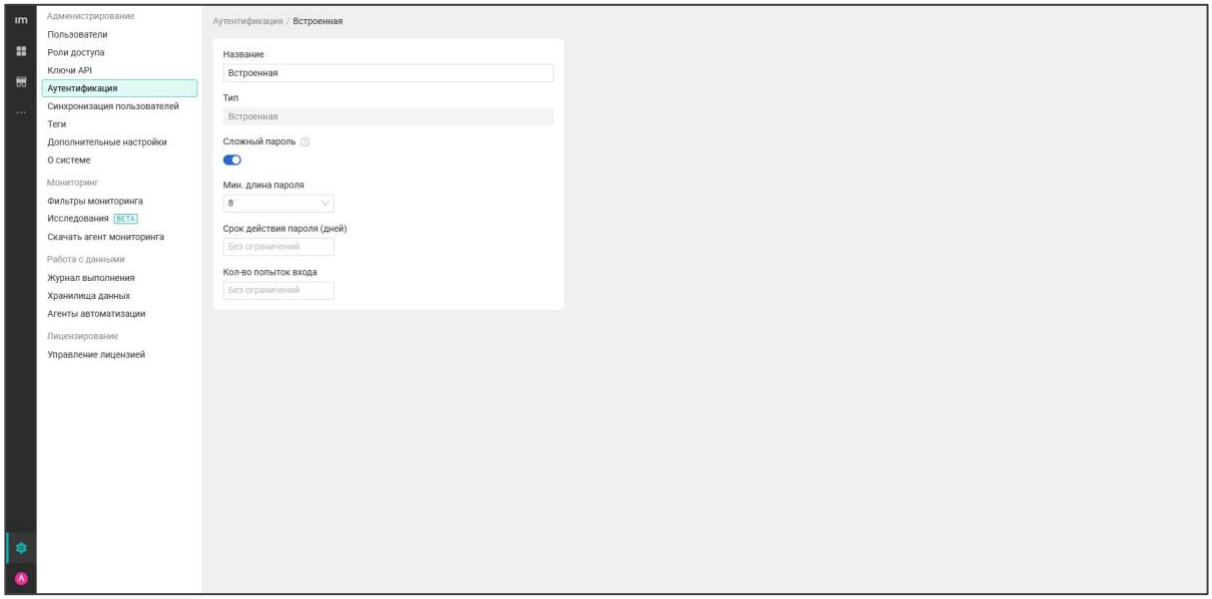
Пользователи из Active Directory, учетные записи которых синхронизируются в систему, могут получить возможность аутентификации на основе своих учетных данных в Active Directory.

Встроенная аутентификация

В интерфейсе отображаются пункты:

- Сложный пароль (вкл/выкл)
- Срок действия пароля (дней)
- Количество попыток входа

При включенном сложном пароле необходимо также указать минимальную длину пароля (от 8 до 15 знаков).



Примечание.

- Если аутентификация с типом *Встроенная* была удалена из таблицы, ее можно создать заново.
- Аутентификация с типом *Встроенная* может быть только одна в системе.

OpenID аутентификация

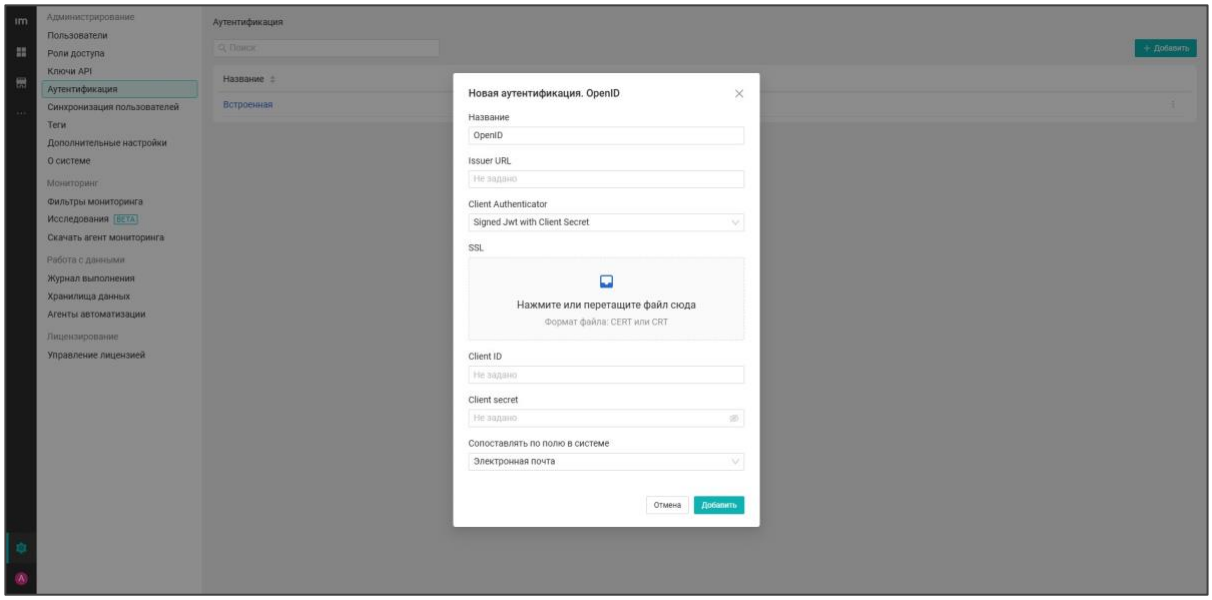
Добавление аутентификации *OpenID* дает возможность входить в систему через протокол OpenID с использованием *KeyCloak*.

OpenID — это протокол для децентрализованной идентификации пользователей в интернете, который позволяет использовать одну учетную запись для входа на различные сайты, что упрощает процесс аутентификации и повышает удобство для пользователей. Подробную информацию можно найти на официальном сайте OpenID.

KeyCloak — это система управления идентификацией и доступом, которая поддерживает различные протоколы, в том числе OpenID, что позволяет централизованно управлять учетными записями пользователей и обеспечивает безопасность аутентификации. Подробно ознакомиться с KeyCloak можно на официальном сайте.

При выборе этого типа в интерфейсе появляются пункты:

- Issuer URL (адрес, который уникально идентифицирует сервер OpenID провайдера)
- Client Authenticator (тип клиентской аутентификации)
- SSL
- Client ID (идентификатор клиента)
- Client secret (клиентский ключ)
- Сопоставлять по полю в системе (выбор атрибута, по которому будут синхронизироваться пользователи, входящие в систему через OpenID)



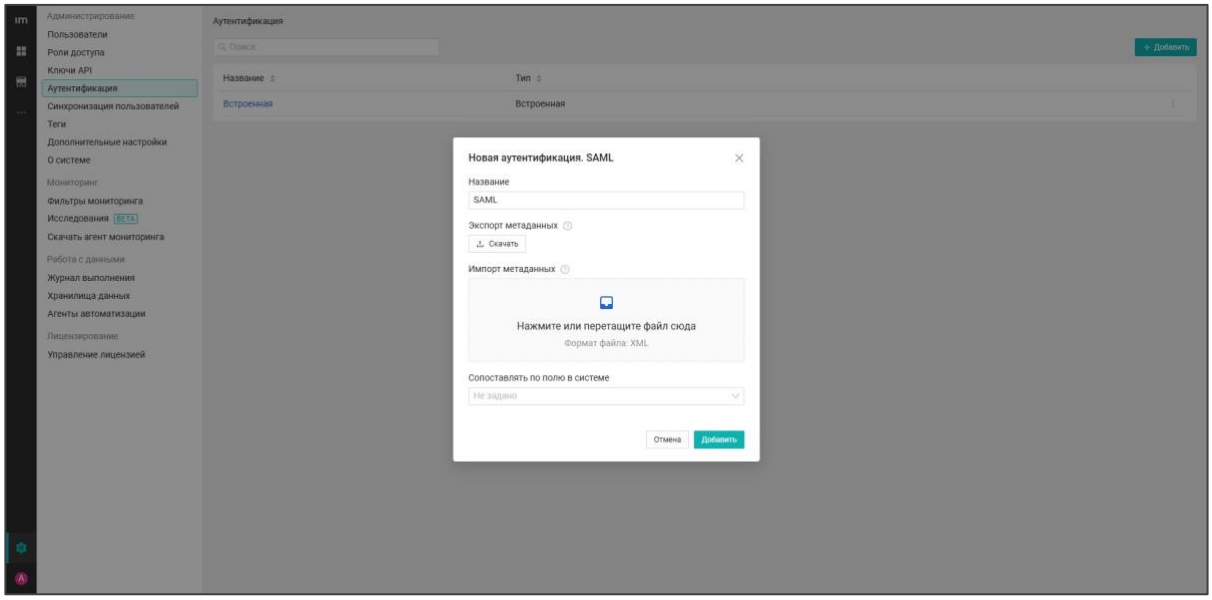
Вход в систему осуществляется через кнопку **Войти через OpenID**.

Подробные инструкции по настройке аутентификации OpenID представлены на странице Настройка аутентификации с использованием OpenID Connect и Keycloak.

SAML аутентификация

Добавление аутентификации *SAML* дает возможность входить в систему с помощью своих учетных данных в AD, используя протокол SAML. При выборе в интерфейсе появляются пункты:

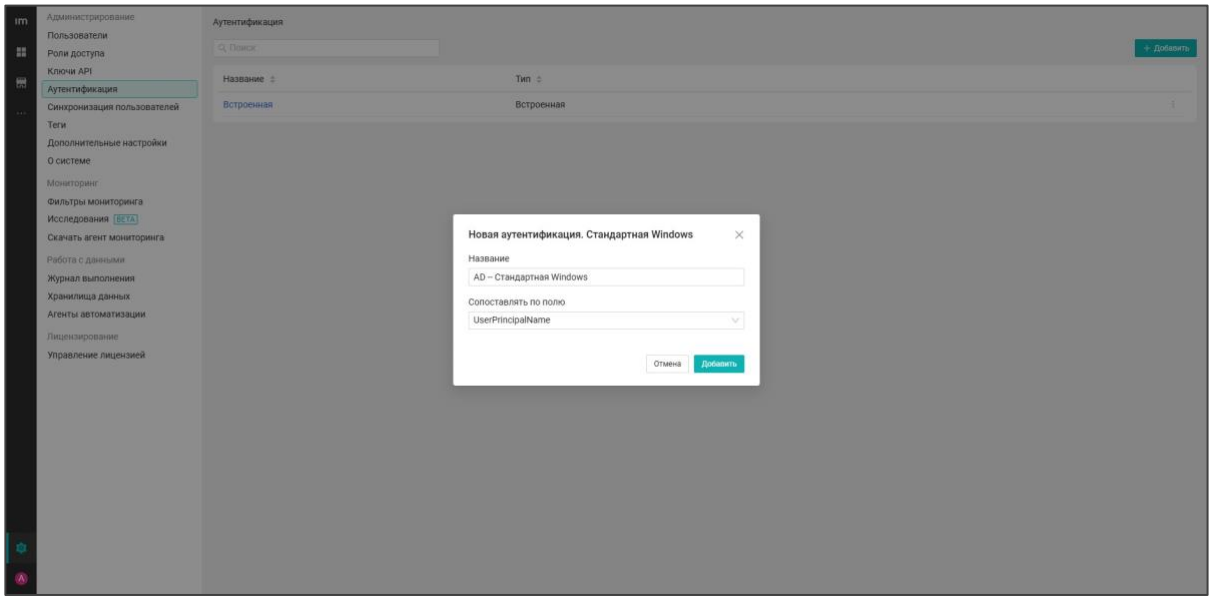
- Экспорт метаданных (скачивание XML-файла — метаданные поставщика сервиса)
- Импорт метаданных (загрузка в систему XML-файла — метаданные поставщика учетной записи)
- Сопоставлять по полю в системе (выбор атрибута, по которому будут синхронизироваться пользователи, входящие в систему через SAML)



Важно. Атрибут для сопоставления нужно выбирать того же формата, что и параметр nameId вашего idP.

Стандартная Windows-аутентификация

Добавление *Стандартной Windows-аутентификации* дает возможность входить в систему с помощью своих учетных данных в AD. При добавлении аутентификации укажите ее название и выберите сопоставляемые атрибуты (выбор атрибута, по которому будут синхронизироваться пользователи, входящие в систему через стандартную аутентификацию).



Примечание.

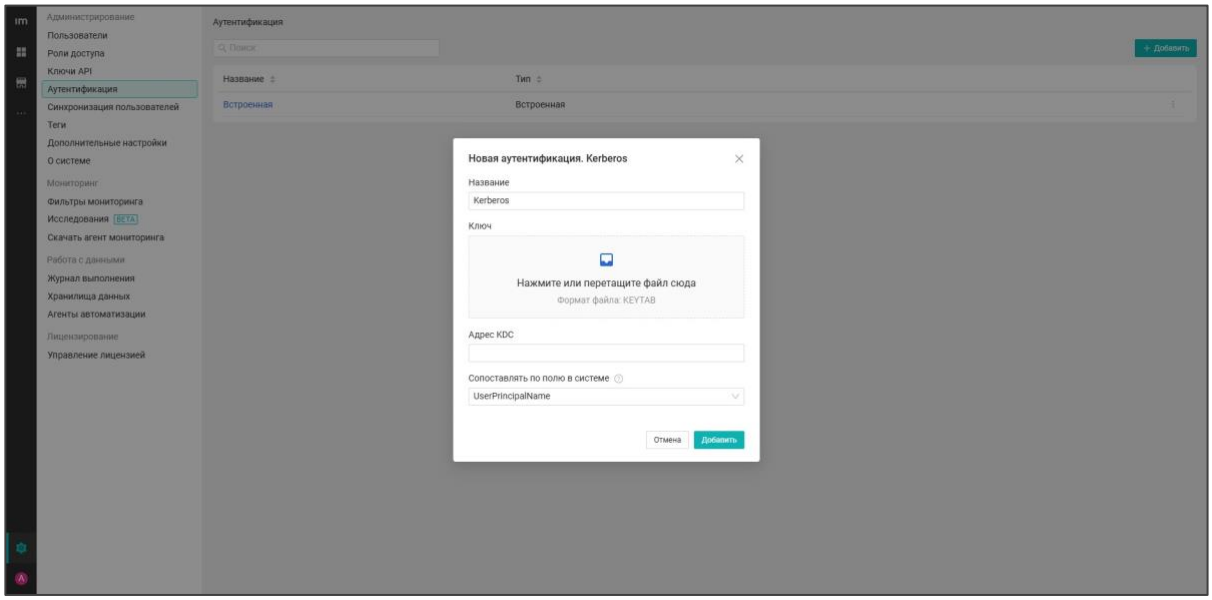
- Стандартная аутентификация не отображается, если на сервере установлена ОС Linux.
- Если в системе уже добавлена аутентификация с типом *Стандартная*, то при создании следующих аутентификаций возможно выбрать только Kerberos.

Kerberos-аутентификация

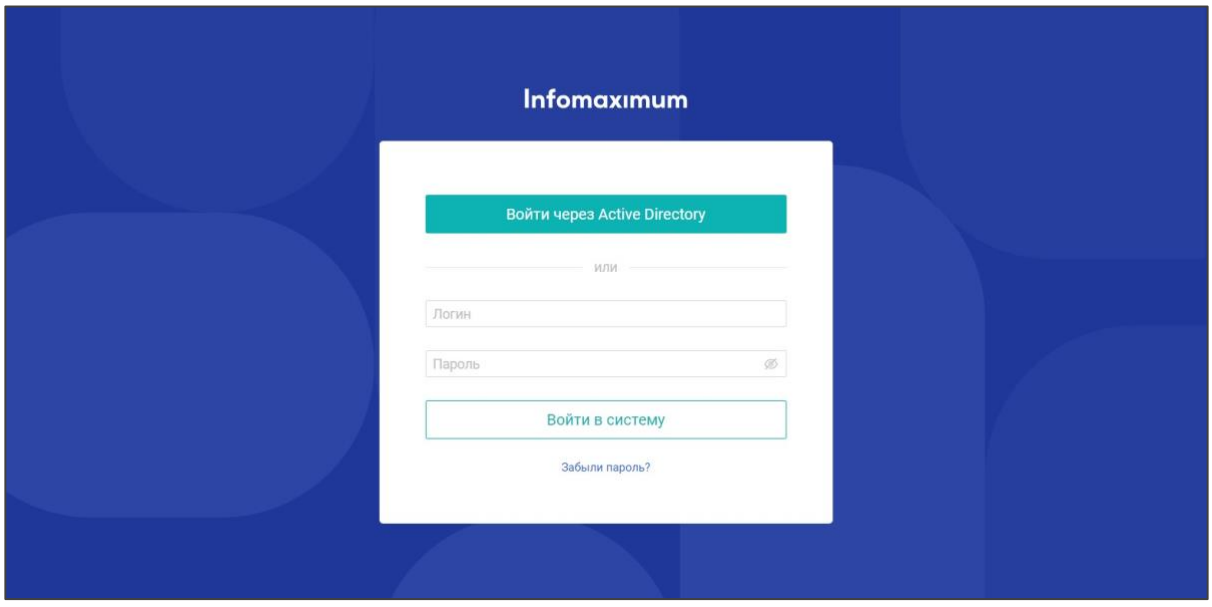
Добавление аутентификации *Kerberos* дает возможность входить в систему с помощью своих учетных данных в AD, используя протокол Kerberos. При добавлении данного типа аутентификации в интерфейсе появляются пункты:

- Ключ (.keytab)
- Адрес KDC
- Сопоставлять по полю в системе (выбор атрибута, по которому будут синхронизироваться пользователи, входящие в систему через Kerberos-аутентификацию)

В случае включения аутентификации *Kerberos* загрузите ранее созданный keytab-файл, укажите имя домена или адрес выдачи ключей и выберите атрибут, по которому будут синхронизироваться пользователи, входящие в систему. При отключении аутентификации ранее загруженный keytab-файл удаляется.



Вход в систему будет осуществляться через кнопку **Войти через Active Directory**.



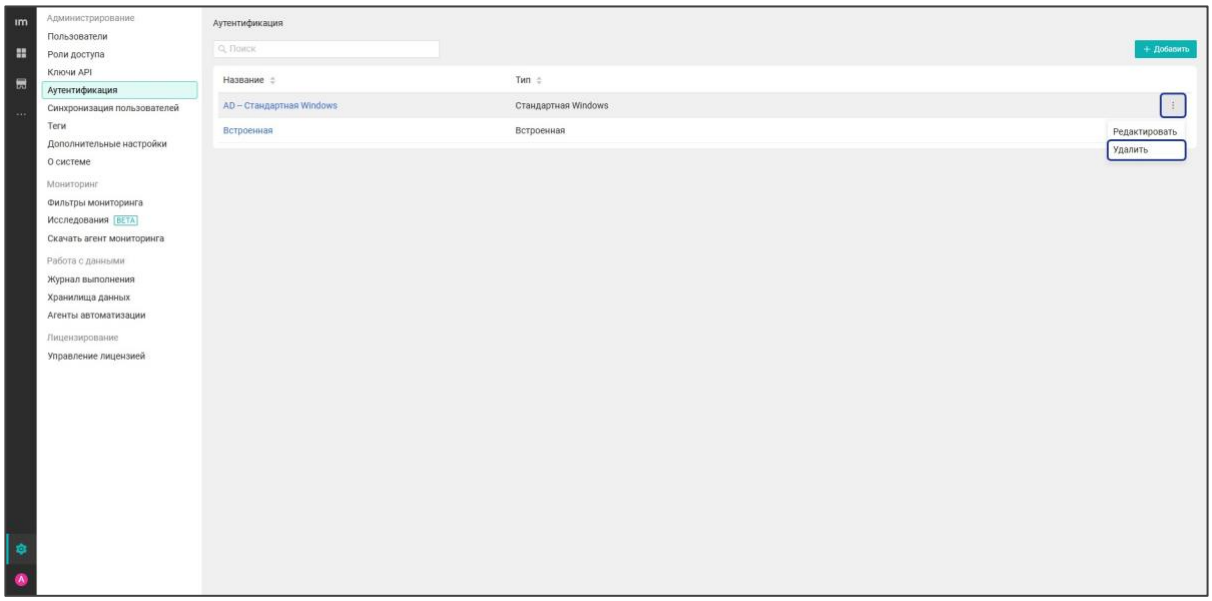
Сопоставление атрибутов

Логика работы сопоставления атрибутов:

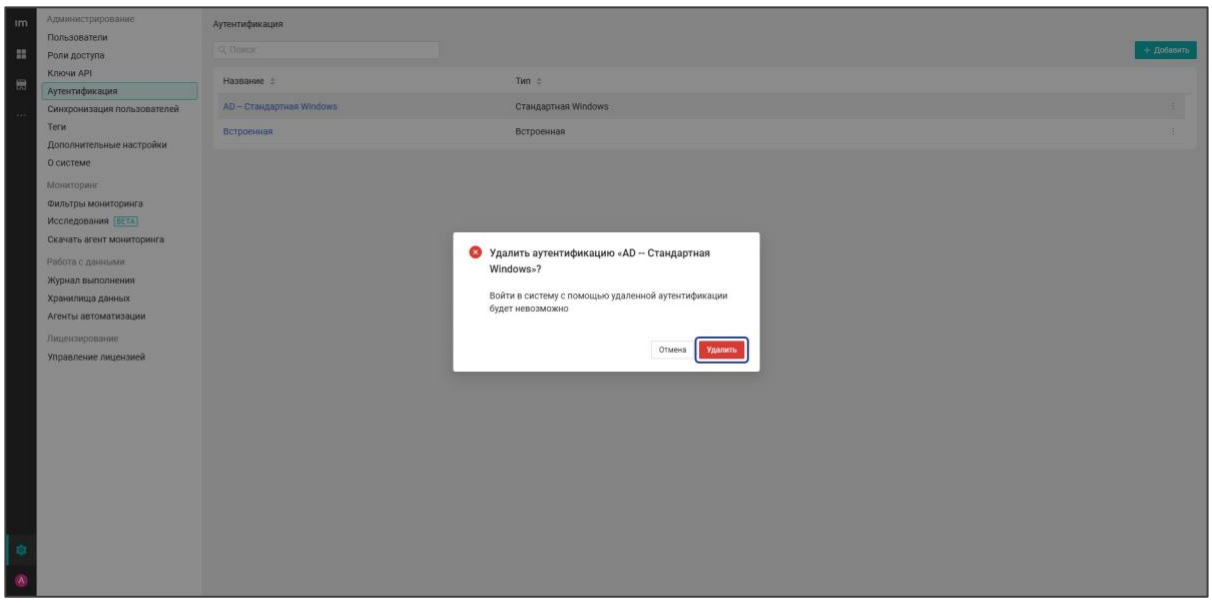
- По умолчанию выбран логин
- Можно выбрать почту или любое кастомное поле. В поле должен быть строковый тип данных
- Сопоставляет выбранное поле с атрибутом UserPrincipleName в AD. Если они совпадают, то сотруднику разрешен вход в систему

Особенности удаления аутентификации

Чтобы удалить аутентификацию, кликните по значку контекстного меню напротив нее и выберите **Удалить**.



Подтвердите удаление в появившемся модальном окне.



Примечание.

- Последнюю/единственную аутентификацию нельзя удалить.

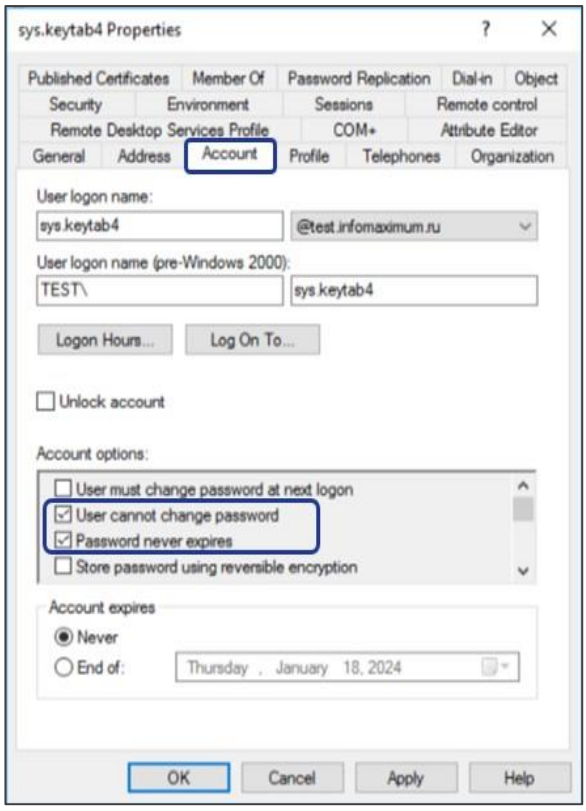
- Если аутентификация назначена сотрудникам ее можно удалить, при этом пользователей выкидывает из системы.

Пример создания keytab-файла для Kerberos аутентификации в Active Directory

Keytab-файл (от «key table», «таблица ключей») — способ хранения долговременных ключей для одного или нескольких принципалов Kerberos. Данный инструмент позволяет сервисам, работающим не на Windows, использовать функционал проверки подлинности Kerberos в инфраструктуре Active Directory.

Для создания keytab-файла в Active Directory выполните следующие действия:

1. Создайте в Active Directory учетную запись с правами обычного пользователя. Она будет использоваться в качестве сервисной учетной записи при создании keytab. Задайте ей известный пароль, а также укажите параметры учетной записи «User cannot change password» (запретить смену пароля пользователем) и «Password never expires» (срок действия пароля не ограничен).



2. Создайте keytab-файл при помощи утилиты командной строки ktpass. Данная утилита доступна в операционных системах Windows Server. В операционных системах рабочих станций может потребоваться установка дополнительных компонентов, чтобы утилита ktpass стала доступна.

Пример команды создания key-tab файла:

```
ktpass /mapuser proceset.test.keytab@TEST.INFOMAXIMUM.RU /princ
HTTP/proceset.test.infomaximum.ru@TEST.INFOMAXIMUM.RU /pass
Strong_Pa$$word /crypto All /ptype KRB5_NT_PRINCIPAL /out
c:\proceset.keytab.keytab
```

Где:

- после /mapuser необходимо указать UPN (UserPrincipalName) созданной сервисной учетной записи. Имя домена обязательно должно быть в верхнем регистре
- после /princ необходимо указать адрес сервера Proceset, который будет использоваться пользователи системы. Адрес указывается в таком же формате, как в примере с префиксом "HTTP/". Имя домена обязательно в верхнем регистре
- после /pass необходимо указать пароль сервисной учетной записи
- после /out необходимо указать путь, по которому будет сохранен keytab-файл

Подробно ознакомиться с ktpass можно на официальном сайте.

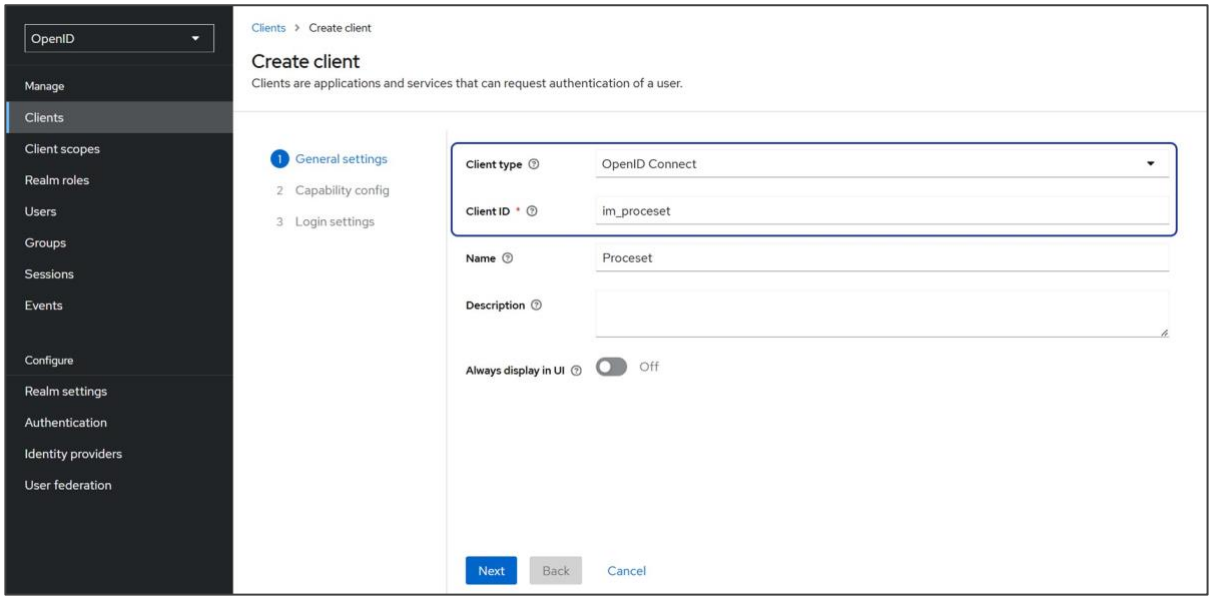
Настройка аутентификации с использованием OpenID Connect и Keycloak

Настройка аутентификации с использованием Client ID и Client Secret через OpenID Connect (OIDC) позволяет повысить безопасность авторизации и централизованно управлять доступом пользователей к системе Proceset через Keycloak. Этот метод особенно полезен, если требуется аутентификация большого числа пользователей.

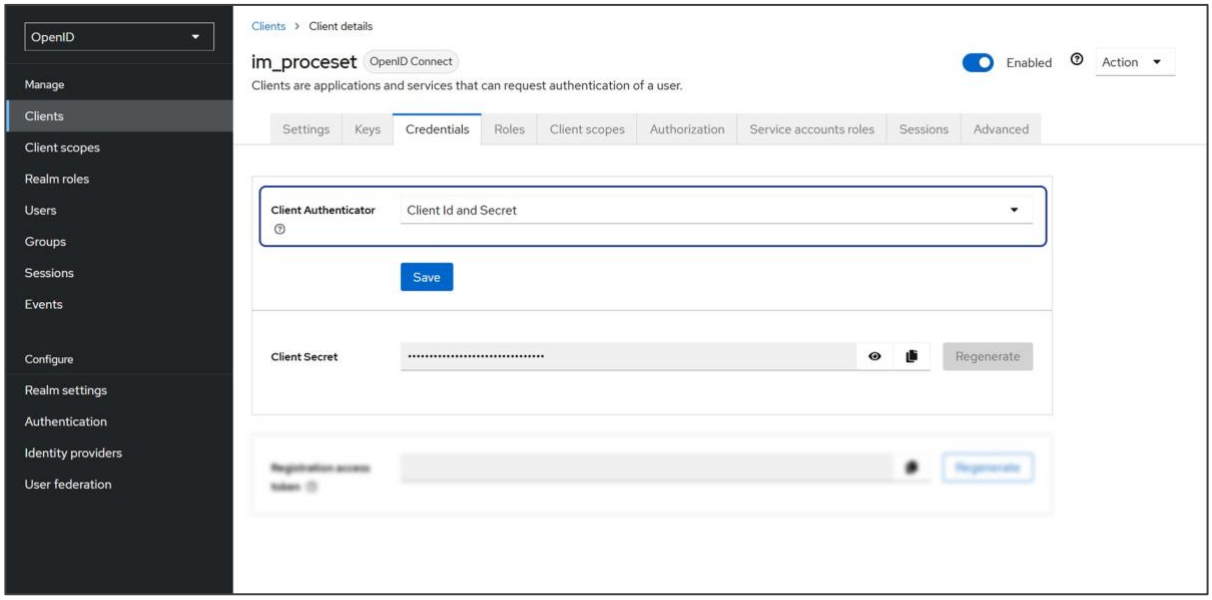
Для настройки аутентификации через OIDC и Keycloak с помощью механизма Client ID и Client Secret выполните описанные ниже шаги.

Настройка клиента в Keycloak

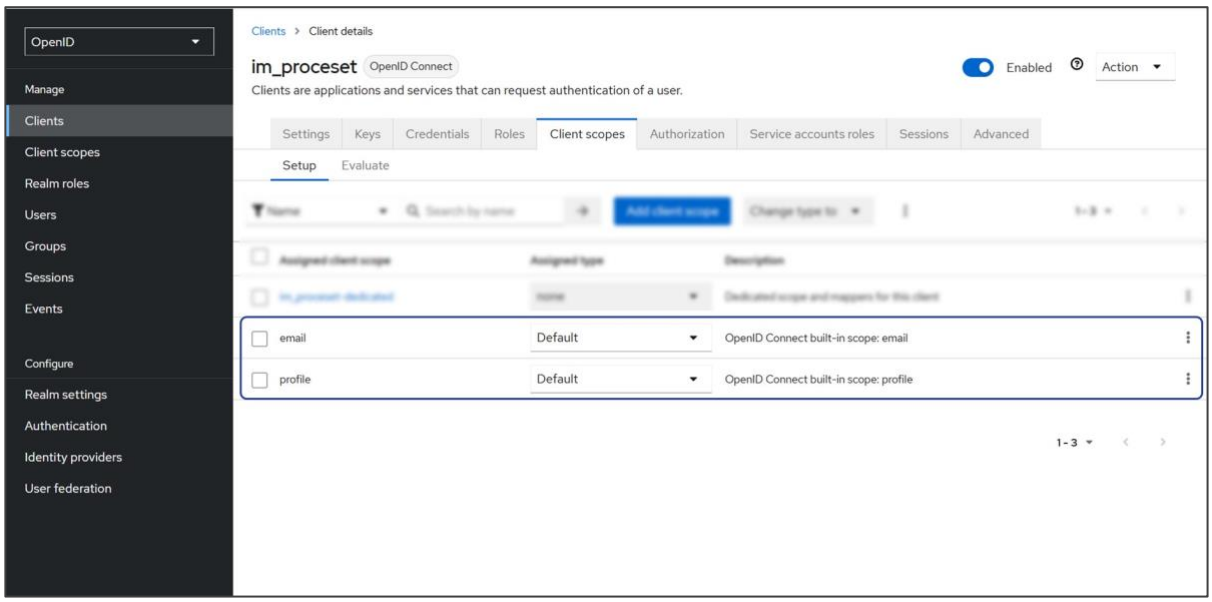
1. Во вкладке настроек нового клиента выберите тип клиента **OpenID Connect** и задайте **Client ID** для идентификации клиента в Keycloak.



2. Во вкладке **Credentials** в поле **Client Authenticator** выберите тип аутентификации **Client ID and Secret**. Этот параметр задает способ аутентификации клиента Proceset в Keycloak.

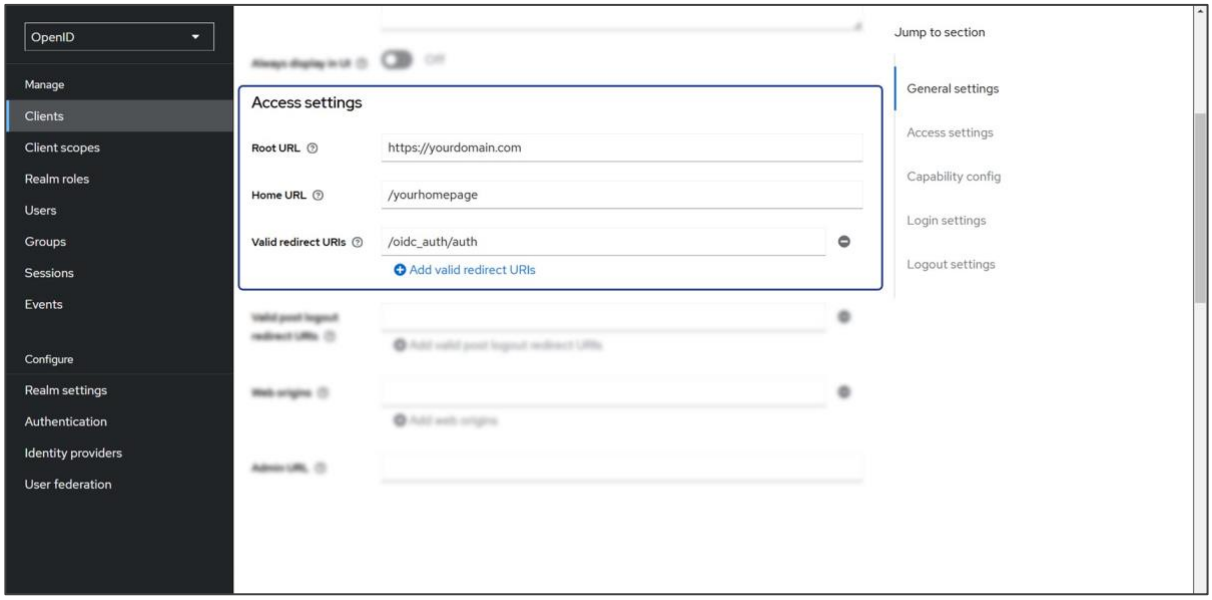


3. Во вкладке **Client scopes** оставьте области (scopes) **email** и **profile**. Остальные области можно удалить. Эти параметры определяют, какие данные передаются клиенту при аутентификации.



4. Перейдите во вкладку **Settings** и в разделе **Access Settings** заполните следующие поля:

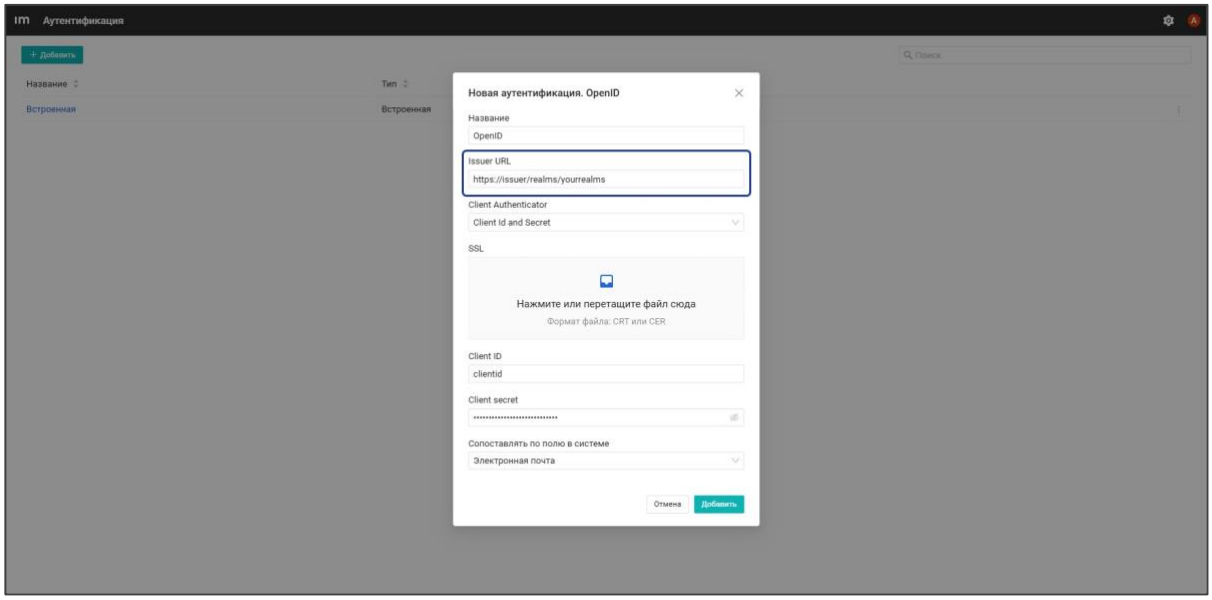
- **Root URL** — адрес домена
- **Home URL** — адрес домашней страницы
- **Valid redirect URIs** — /oidc_auth/auth



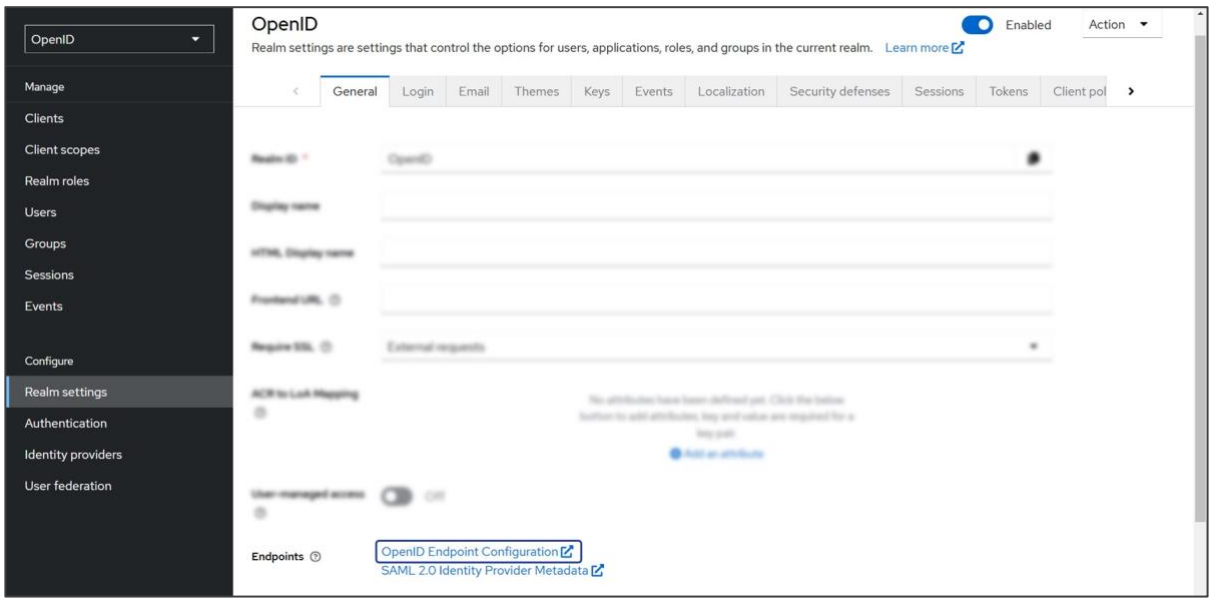
Добавление аутентификации в Proceset

1. Перейдите в раздел настроек *Аутентификация* и добавьте новый тип аутентификации — OpenID.

2. В открывшемся окне в поле **Issuer URL** введите URL из атрибута **issuer**.



Чтобы увидеть значение атрибута, перейдите во вкладку **Realm Settings** в Keycloak и откройте ссылку **OpenID Endpoint Configuration**. Этот URL позволяет системе Proceset обращаться к OIDC-провайдеру.



3. Заполните поля **Client Authenticator**, **Client ID** и **Client secret** значениями из Keycloak.

4. Нажмите **Добавить**.

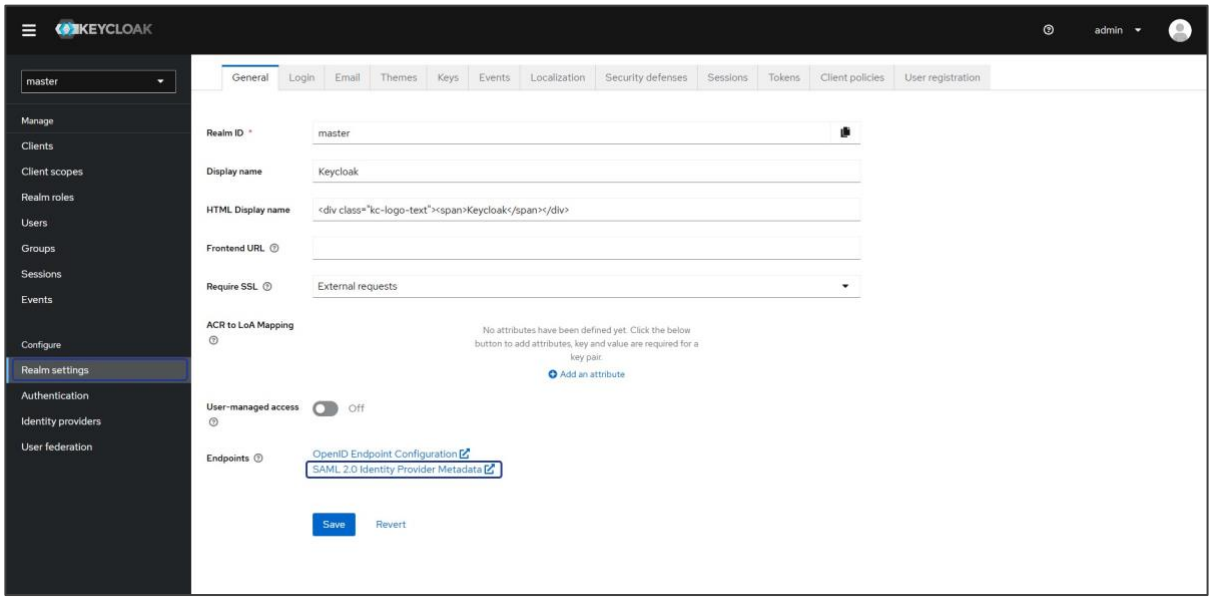
5. Назначьте созданную OpenID-аутентификацию хотя бы одному пользователю Proceset, чтобы она стала доступной для использования.

Аутентификация по протоколу OIDC с использованием Client ID и Client Secret успешно настроена. Теперь пользователи Proceset могут проходить безопасную аутентификацию через Keycloak.

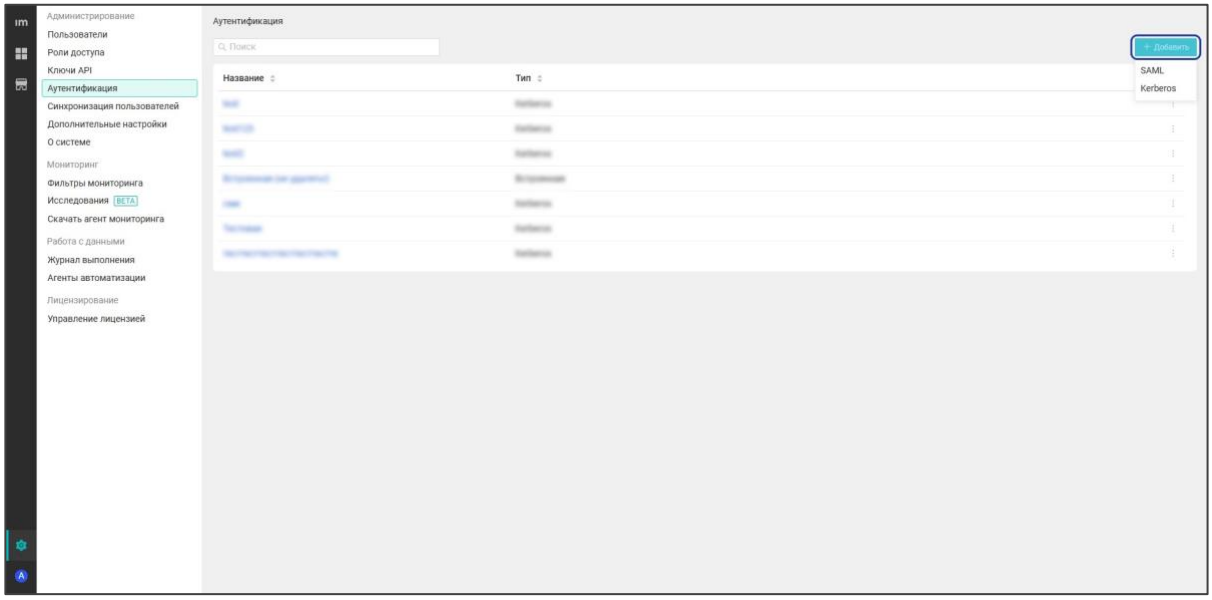
Настройка аутентификации SAML в Keycloak

Для настройки аутентификации SAML в Keycloak выполните шаги, описанные ниже.

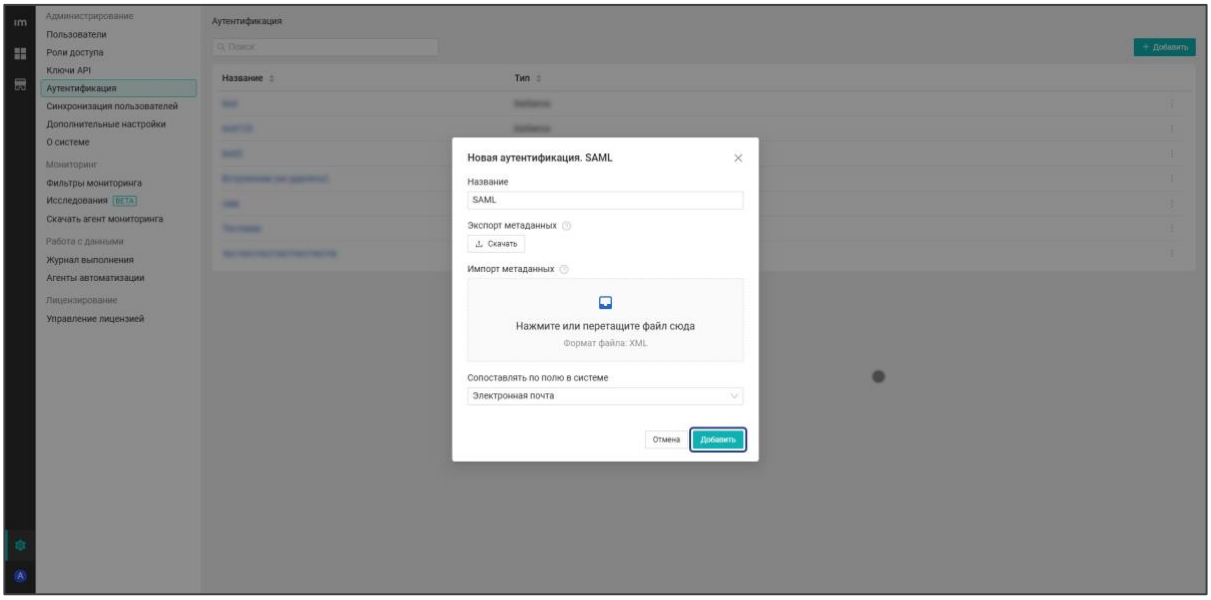
1. Войдите в Keycloak. Перейдите в раздел *Realm settings* и скачайте файл по ссылке *SAML 2.0 Identity Provider Metadata*.



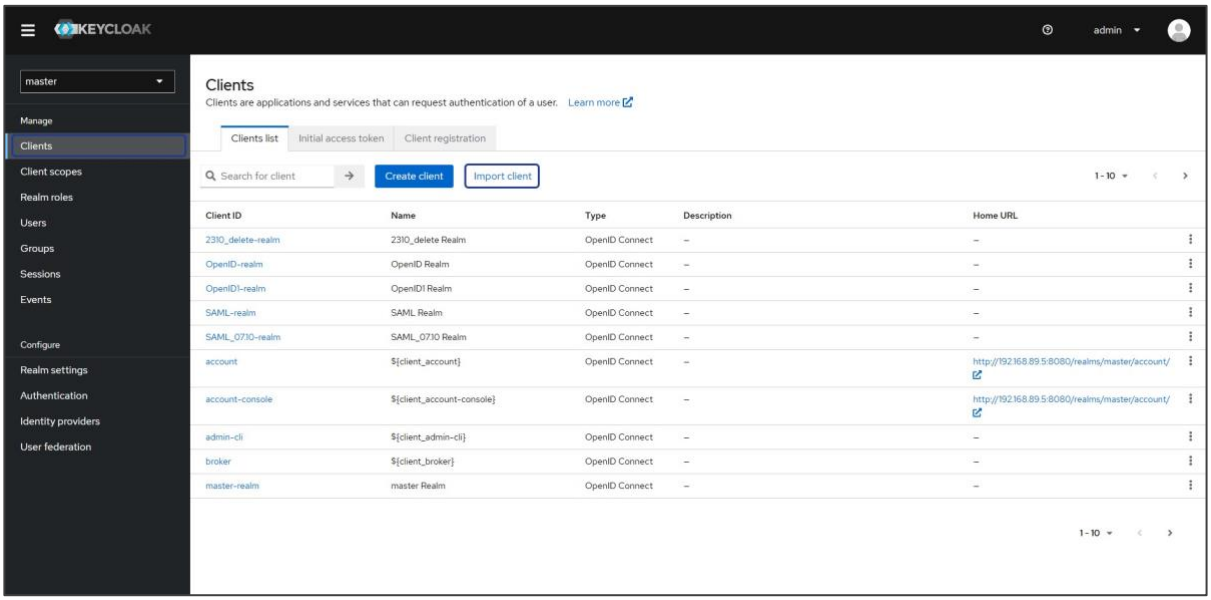
2. Войдите в систему Proceset. Откройте раздел Аутентификация и кликните по кнопке Добавить, а затем выберите SAML.



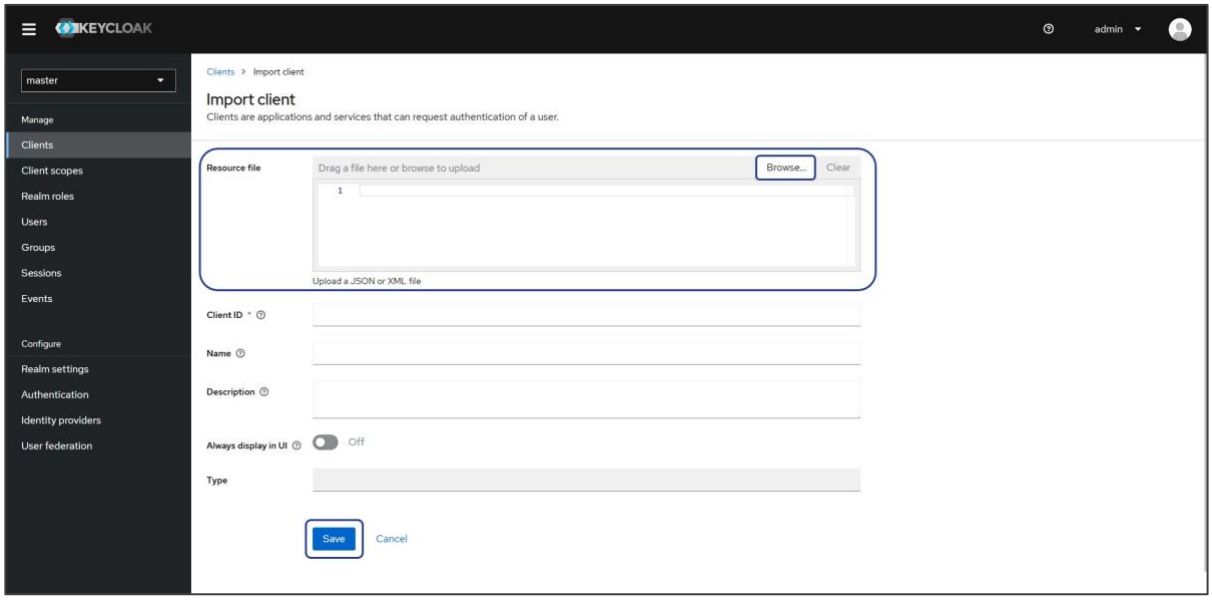
3. Экспортируйте файл метаданных поставщика услуг. Он понадобится для загрузку в Keycloak. Импортируйте скачанный файл из пункта 1 в Proceset, выберите атрибут для сопоставления и добавьте аутентификацию.



4. В Keycloak перейдите в раздел *Clients*, во вкладку *Import clients*.

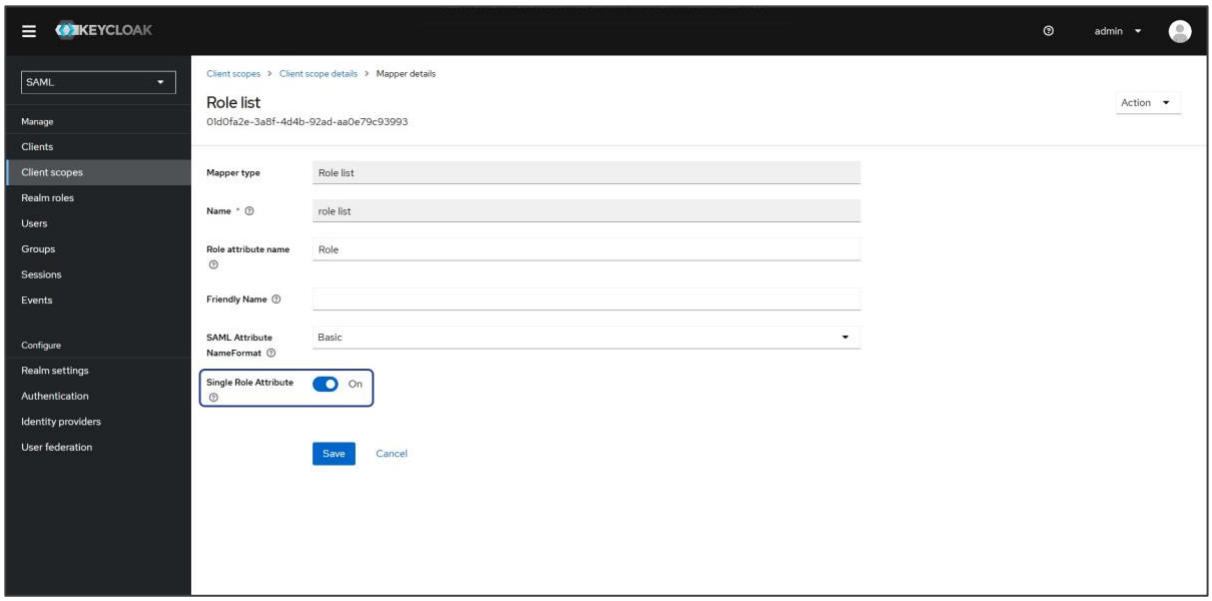


5. В поле **Resource file** загрузите выгруженные метаданные из Proceset и нажмите **Save**.



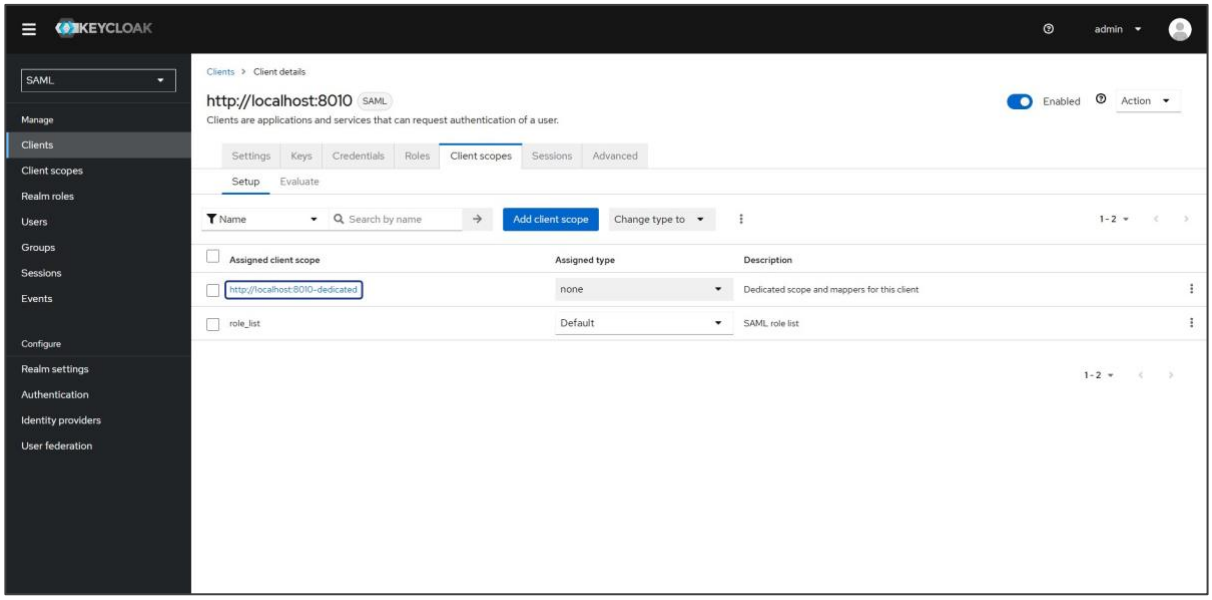
Аутентификация настроена и готова к использованию.

Важно. Если при авторизации через SAML возникает ошибка {"code":"saml_auth_error","error":"Found an Attribute element with duplicated Name"}, в Keycloak перейдите в раздел Mapper details и активируйте параметр Single Role Attribute.



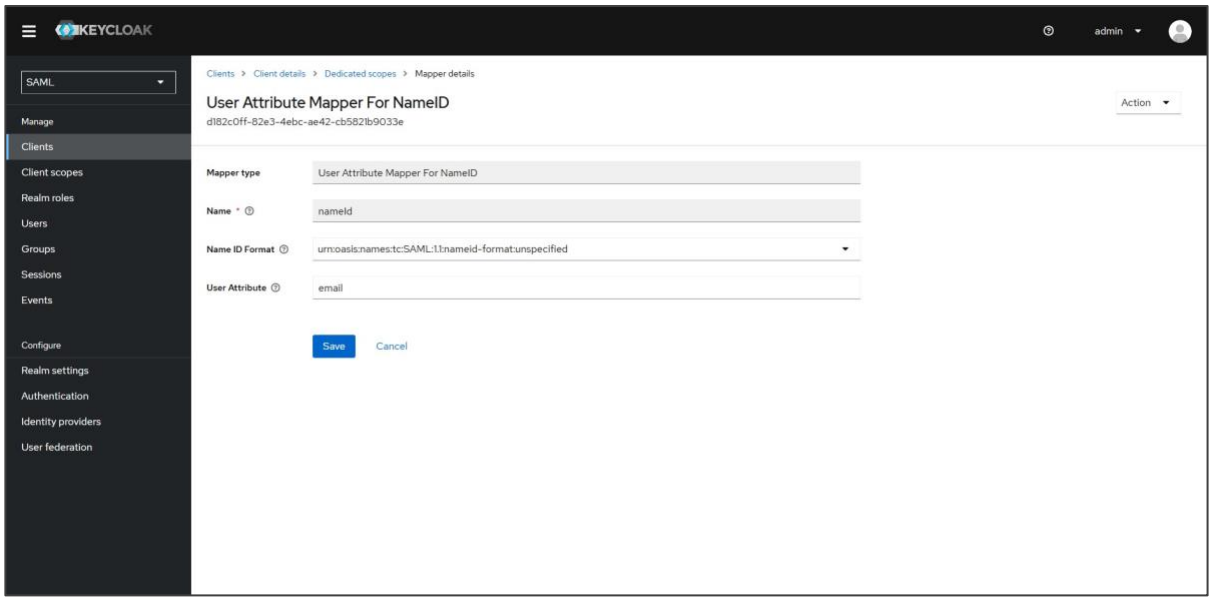
Атрибут для сопоставления необходимо выбрать того же формата, что и параметр nameId вашего idP. Для синхронизации Keycloak с Proceset через почту необходимо совпадение формата параметра nameID с форматом поля Электронная почта в профиле сотрудника:

1. Перейдите в раздел *Clients scopes* и выберите интересующую область действия клиента.



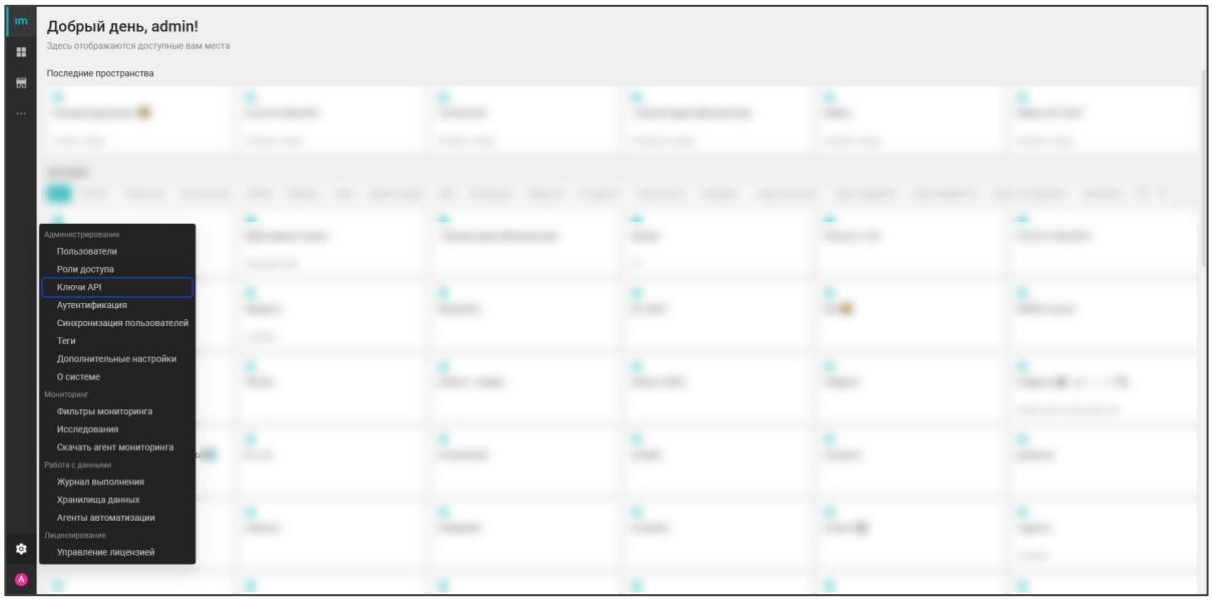
2. Выберите **Configure a new mapper**, а затем **User Attribute Mapper For NameID**.

3. В поле **User Attribute** укажите email и нажмите **Save**.



Добавление ключей API

На странице *Ключи API* можно сгенерировать ключи API. *Ключ API* — это один из способов аутентификации в системе, который может быть использован внешними ИС для работы с системой. Ключи API используются в том числе агентами мониторинга для аутентификации на сервере приложения.



Атрибуты ключа API:

- Название ключа — должно быть уникальным (можно изменить в профиле ключа, во вкладке *Основное*)
- Значение ключа — уникальная последовательность символов, генерируется автоматически при создании ключа
- Режим аутентификации — с помощью какого атрибута будет проходить аутентификация
- Доступ к пространствам — к каким пространствам данных имеет доступ ключ

Возможные действия:

1. Добавить ключ.

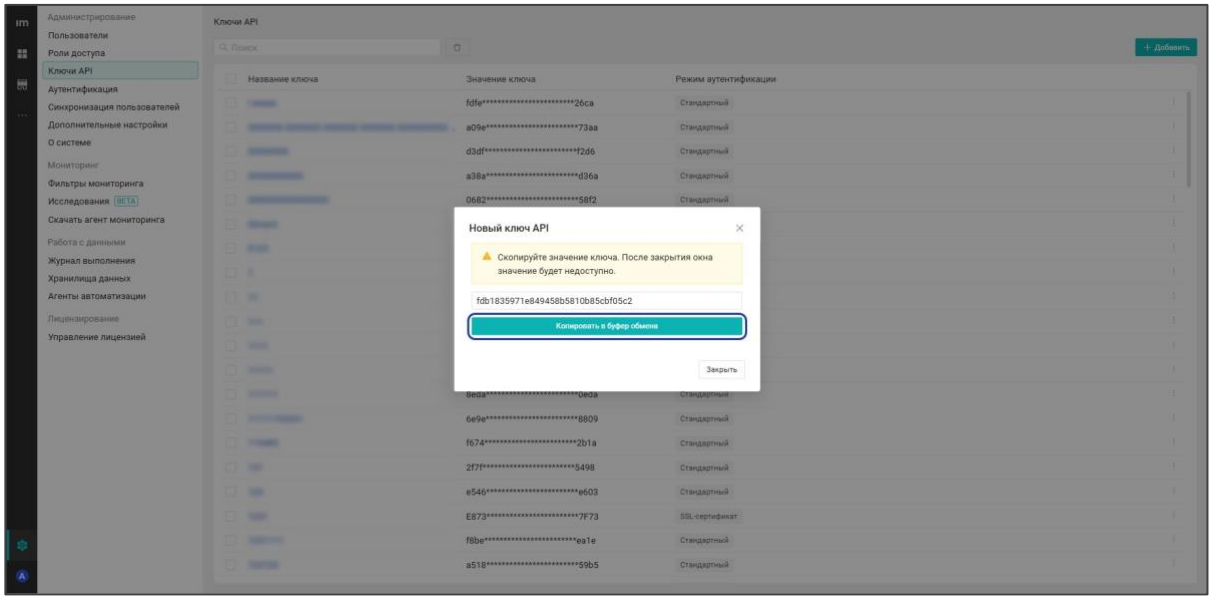
2. При нажатии значка контекстного меню появляется список выбора действий с данным элементом:

- При нажатии **Настроить** происходит переход в профиль ключа во вкладку *Привилегии*
- Удалить (кнопка активна, когда выбран хотя бы один ключ)

3. Поиск необходимого ключа.

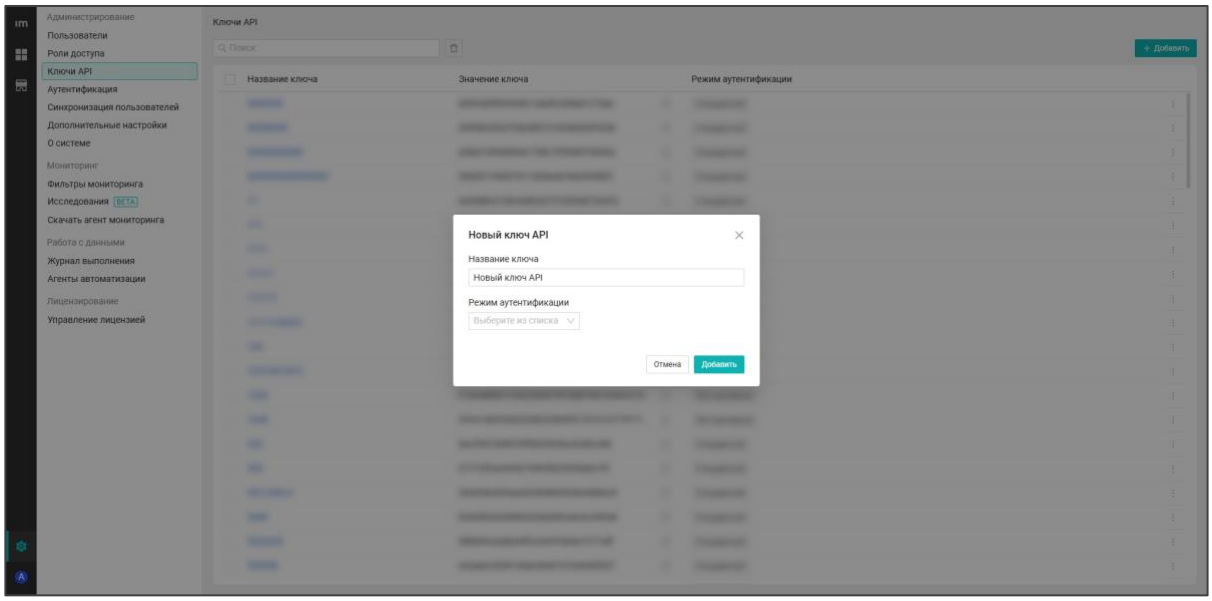
Когда нет загруженных ключей в системе, при добавлении стандартного ключа его значение генерируется автоматически.

Важно. Значение ключа отображается полностью только один раз — в момент генерации. Если хотите просмотреть ключ позже, сохраните его. После генерации он больше не отображается на странице полностью.



Необходимо контролировать уникальность названия ключа API:

- Сертификат является файлом с расширением .crt или .cert
- Обновить сертификат API ключа возможно. Для этого нужно загрузить новый сертификат в профиле ключа. Старый сертификат становится недействительным
- Для предустановленного безопасного API ключа используется имя *Агент мониторинга*

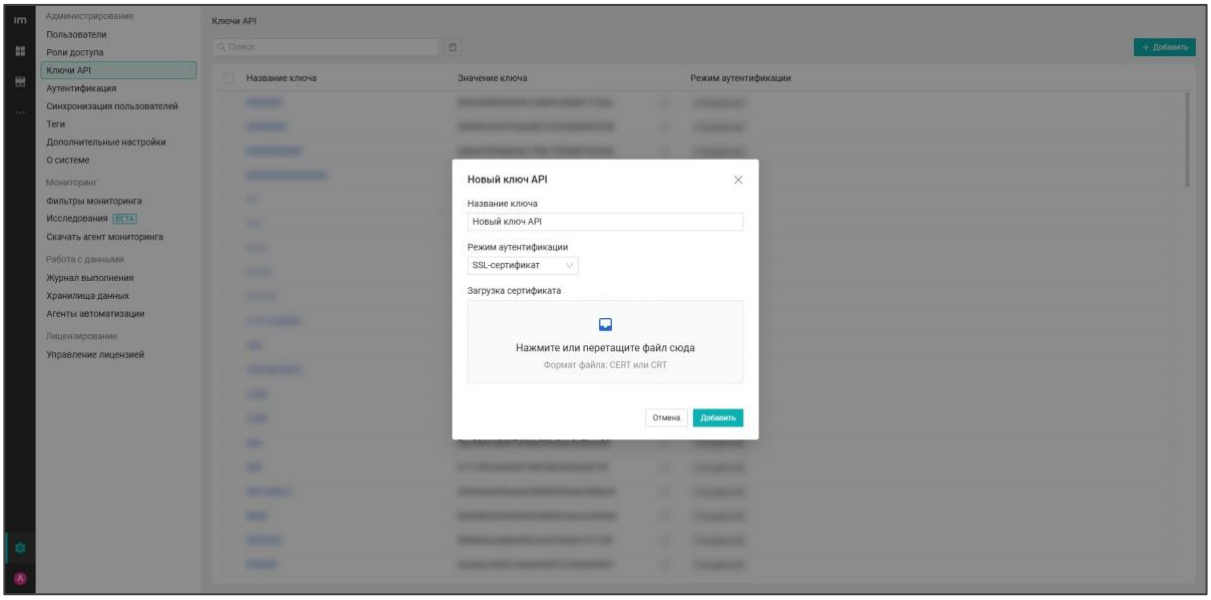


При добавлении ключа выберите режим аутентификации:

- SSL-сертификат
- Стандартный
- Active Directory
- Стандартная Windows

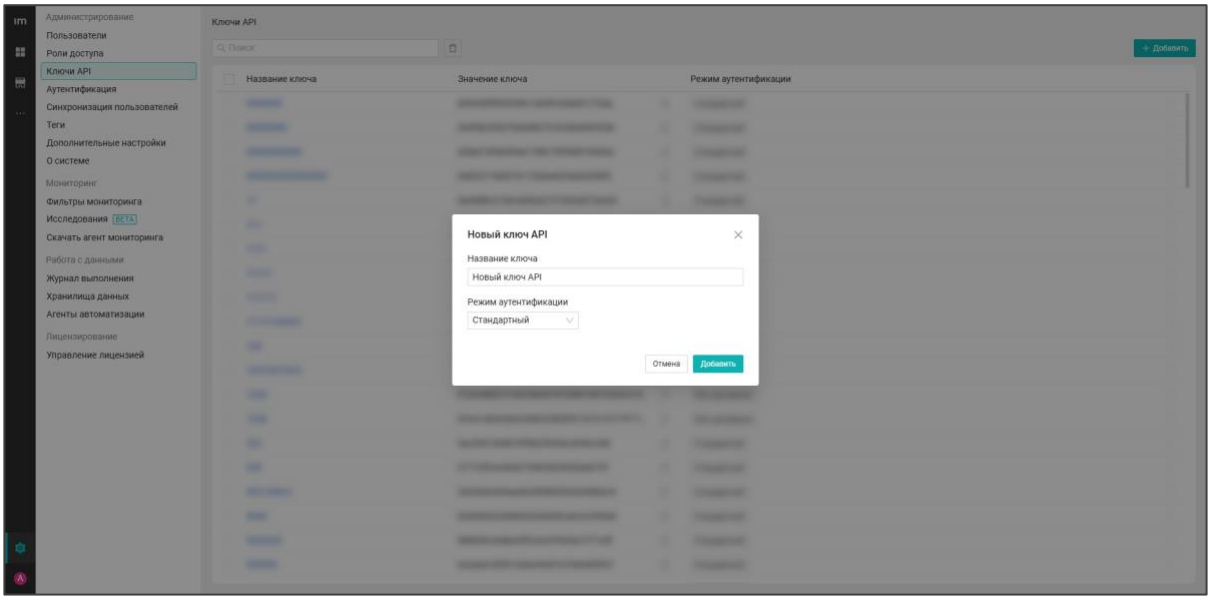
При выборе SSL-сертификата можно:

- Задать название ключа API
- Загрузить сертификат (можно использовать самоподписанный сертификат)
- Удалить загруженный файл

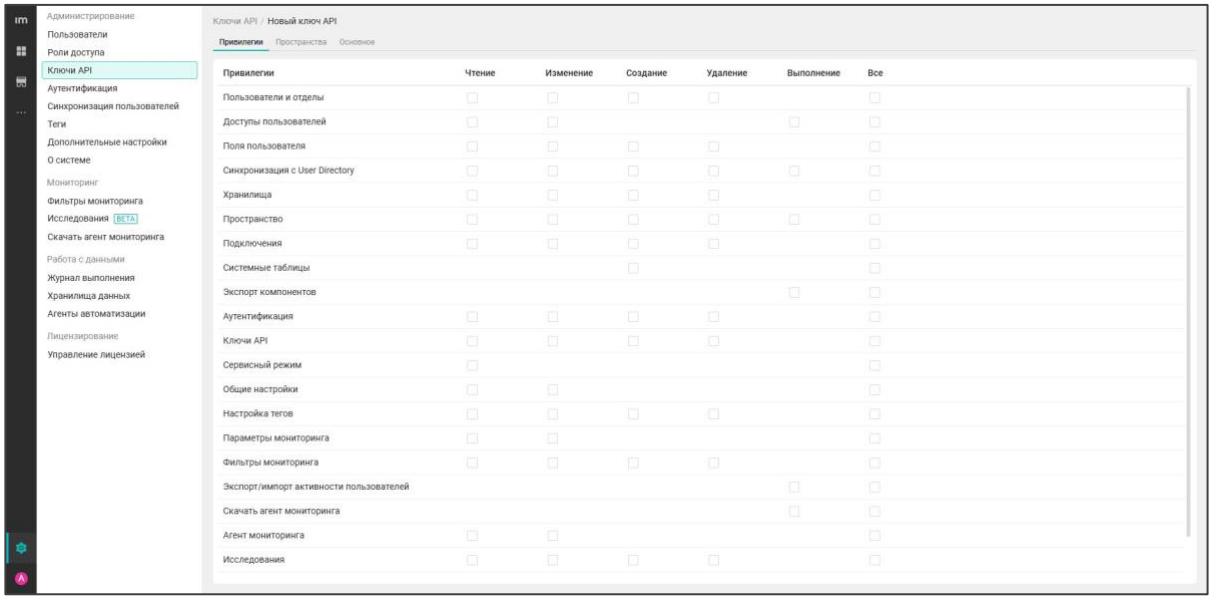


В профиле ключа API с аутентификацией по SSL-сертификату отображается его отпечаток.

При выборе *Стандартный* можно задать название ключа API.



Далее в привилегиях ключа API установите необходимые операции доступа для ключа API. Подробнее с привилегиями в системе можно ознакомиться в разделе *Ролевая модель*.



Во вкладке *Пространства* для ключа API настраивается доступ на просмотр или изменение пространств. Чтобы предоставить доступ, нажмите кнопку **+ Доступ**, выберите необходимые пространства и нажмите **Сохранить**.

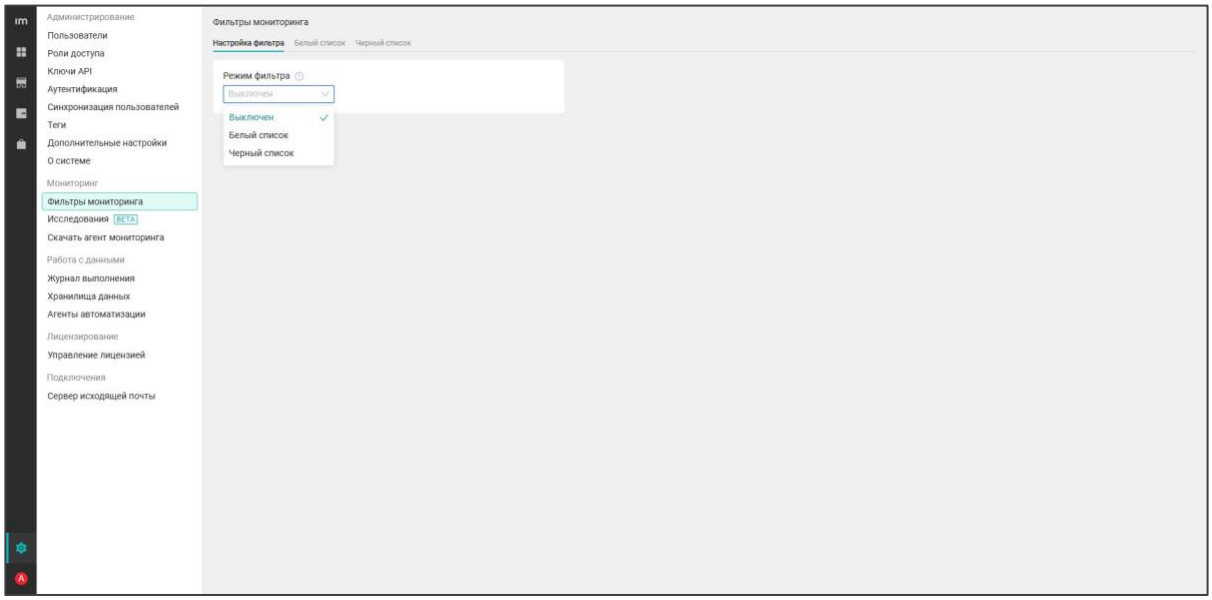
Во вкладке *Основное* можно изменить название ключа и просмотреть его режим аутентификации.

На стороне внешней системы необходимо произвести настройку HTTPS-соединения, хранилища сертификатов, а также клиентских сертификатов. После чего можно выполнять GraphQL-запросы к системе. Для возможности соединения в системе в обязательном порядке должна быть выполнена настройка клиентской авторизации.

Раздел «Мониторинг»

Фильтры мониторинга

На странице *Фильтры мониторинга* можно включить и настроить списки программ, по которым агент мониторинга будет собирать или игнорировать активность пользователей.

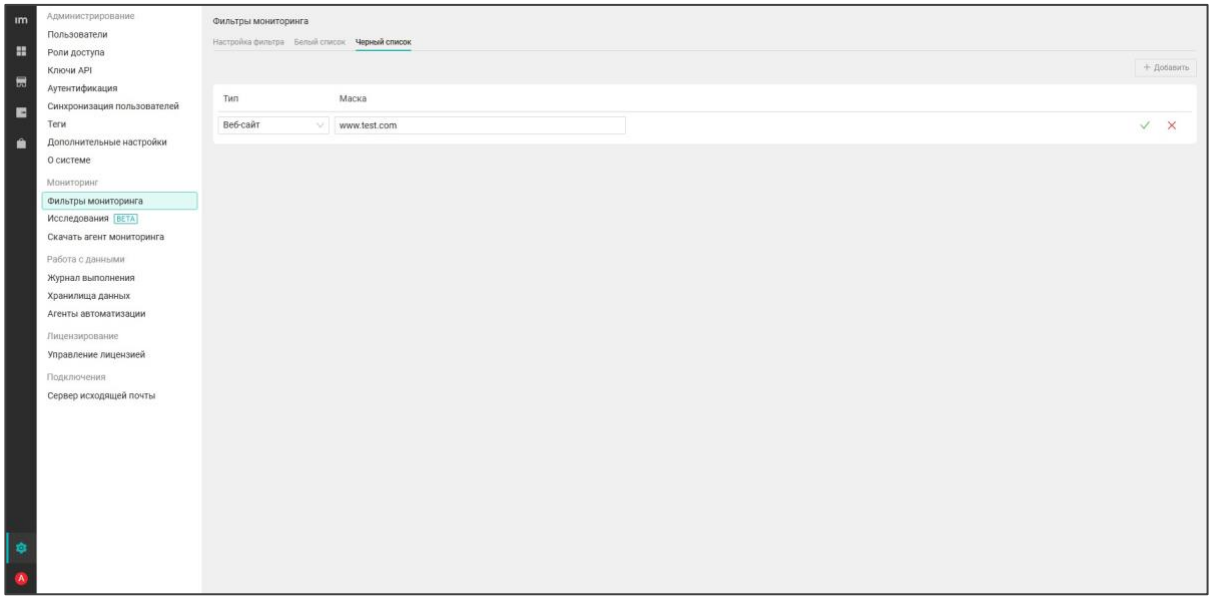


Белый и черный списки — списки активностей, статистику по которым нужно или не нужно собирать в системе. Они обновляются на агентах мониторинга раз в час. Чтобы ускорить обновление настроек на определенном компьютере, перезапустите службу Proceset Agent.

Если выбрано:

- *Выключен* — активность не фильтруется агентами мониторинга
- *Белый список* — агенты мониторинга будут собирать информацию только по программам, находящимся в белом списке
- *Черный список* — агенты мониторинга будут собирать информацию по работе со всеми программами, кроме тех, что указаны в черном списке

При выборе белого или черного списка необходимо указать приложения и сайты, по которым будет или не будет собираться активность.

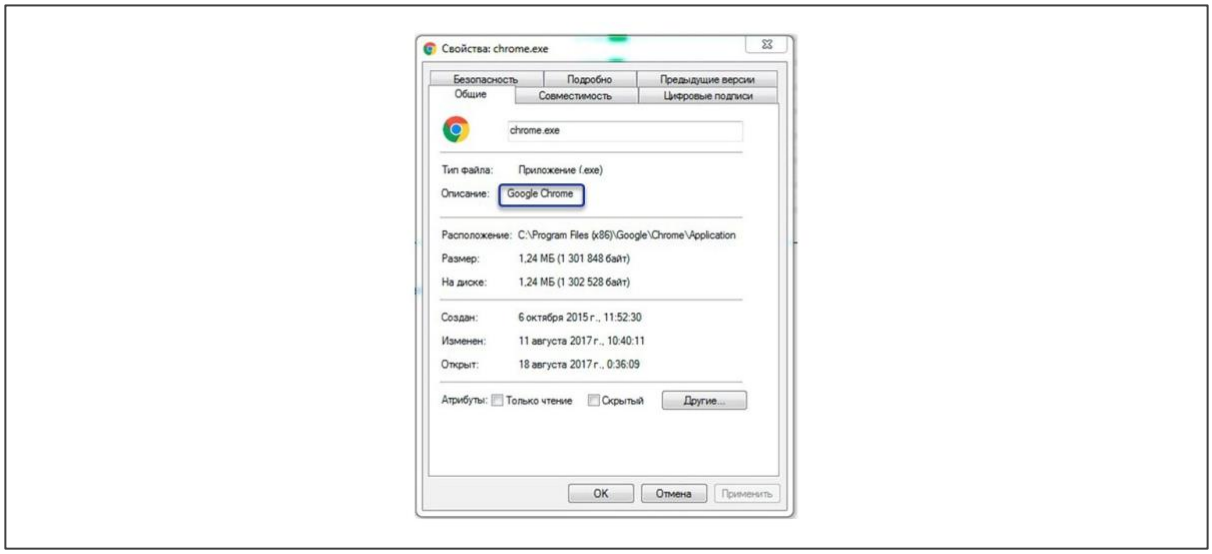


В таблице укажите:

- Тип (*приложение* или *веб-сайт*)
- Маску (название программы или URL сайта)

Некоторые программы, как Microsoft Word, в разных версиях могут называться по-разному. Для того, чтобы объединить все версии, необходимо использовать знак *«*»*, поставив его после названия (например, Word*). При этом будут игнорироваться все символы, стоящие после *«*»*. Таким образом, программы с названиями *Word 2003, Word 11, Word 2016* автоматически будут объединены.

Приложения для добавления в черный и белый списки нужно задавать в виде масок по названию в том виде, в котором они заданы в свойствах exe-файла во вкладке *Общие* в строке *Описание*.



Сбор скриншотов агентом мониторинга

Примечание. Эти функции доступны в бета-версии. Попробуйте их в работе и поделитесь своим мнением — обратная связь помогает нам развивать продукт.

Агент мониторинга позволяет собирать скриншоты при определенных действиях пользователей. Узнать больше о сборе скриншотов можно на странице Сбор скриншотов агентом мониторинга

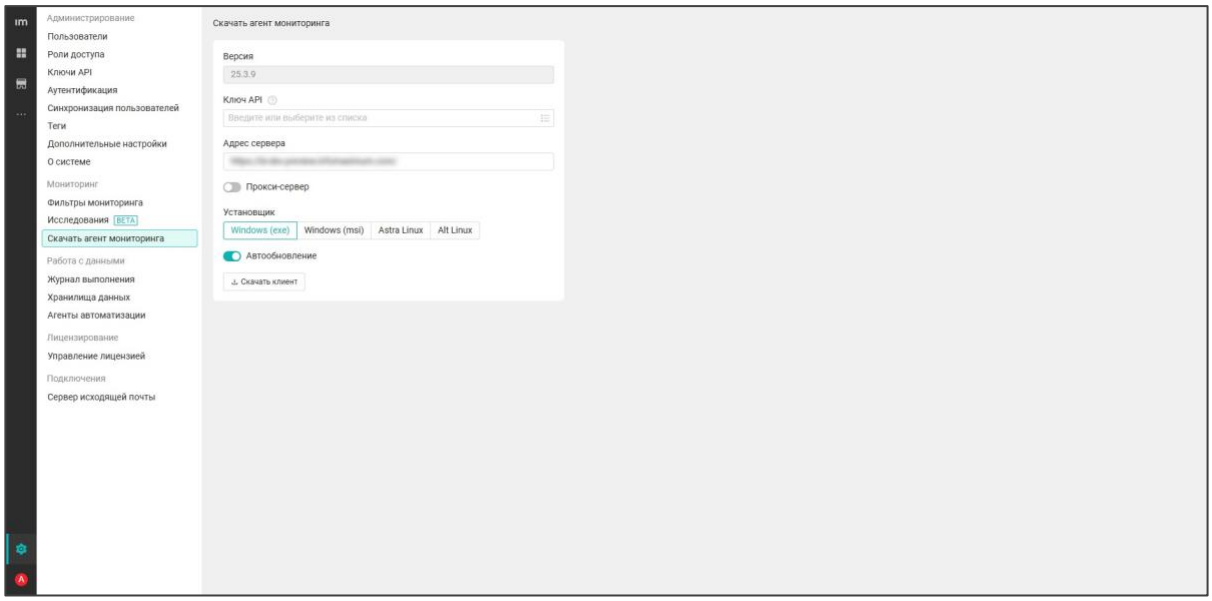
Исследования

Примечание. Эти функции доступны в бета-версии. Попробуйте их в работе и поделитесь своим мнением — обратная связь помогает нам развивать продукт.

Исследования позволяют настроить сбор скриншотов агентом мониторинга. Подробности настройки сбора скриншотов описаны на странице Сбор скриншотов агентом мониторинга.

Скачать агент мониторинга

Страница *Скачать агент мониторинга* подробно описывается на странице Дистрибутив агента мониторинга.



Раздел «Подключения»

В разделе *Подключения* можно настроить сервер исходящей почты и контакты технической поддержки.

Сервер исходящей почты

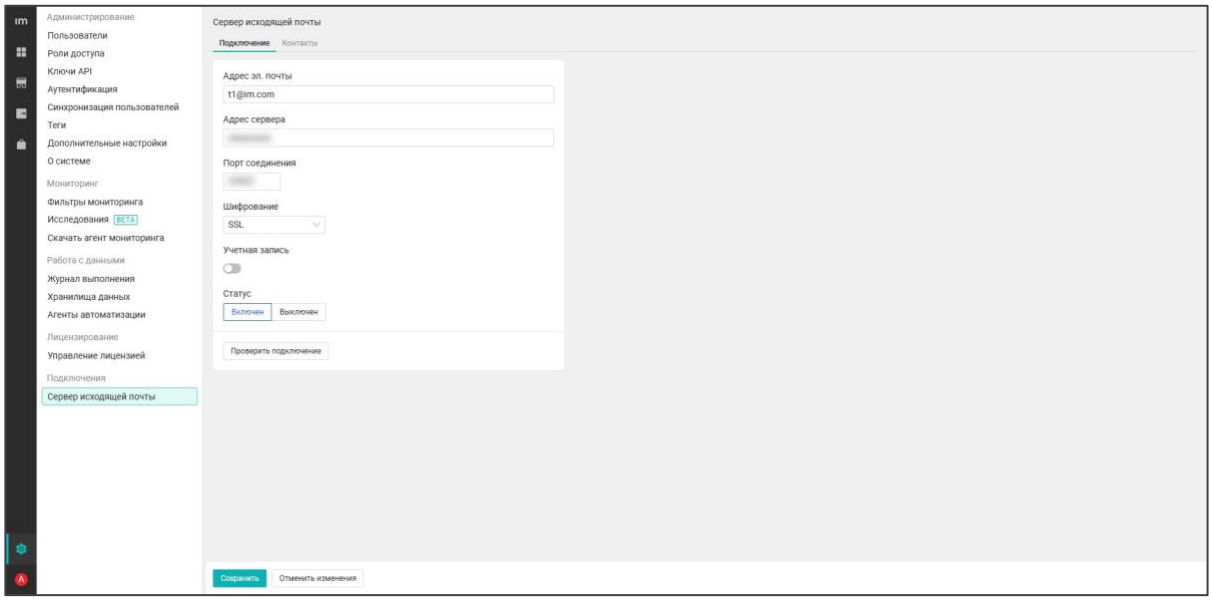
На странице *Сервер исходящей почты* настраиваются параметры SMTP-сервера, который система будет использовать для рассылки писем по e-mail. Для настройки во вкладке *Подключение* укажите параметры:

- Адрес электронной почты — от имени которой будут отправляться письма
- Адрес сервера — FQDN или IP-адрес SMTP-сервера
- Порт соединения
- Шифрование (Выкл/TLS/SSL)
- Учетная запись (Вкл/Выкл). Если учетная запись включена, то появляются поля:
    - Имя пользователя — используемое для аутентификации на SMTP-сервере
    - Пароль (если пароль ранее уже был задан, то появляется кнопка **Изменить пароль**)
- Статус (Включен/Отключен)

Если в параметре **Шифрование** выбрать пункт *Выключено*, то можно подключаться по шифрованному (STARTTLS) и по нешифрованному каналам в зависимости от того, что настроено на порте соединения. В случае подключения к шифрованному каналу проверка получаемых сертификатов не производится, т. е. безопасное соединение не устанавливается.

Статус позволяет включать и отключать сервер исходящей почты. При статусе *Отключен* можно также проверить соединение.

Используемая библиотека: https://www.simplejavamail.org/configuration.html.



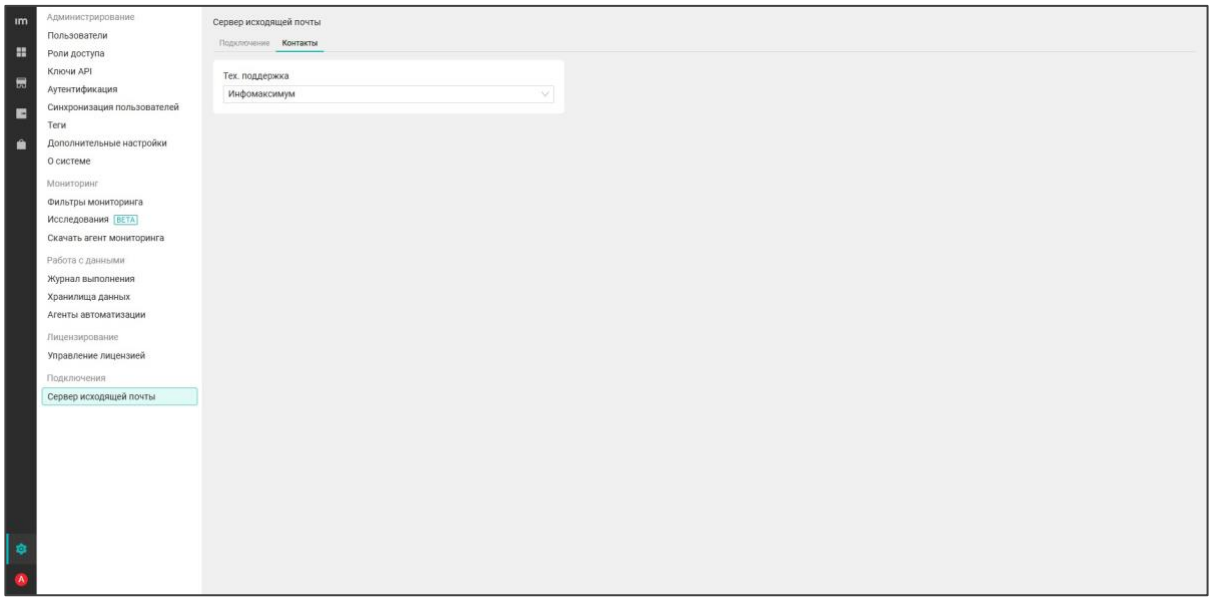
Когда заполнены все обязательные поля, становятся доступны кнопки:

- **Сохранить** — при сохранении происходит проверка соединения. Если сервер исходящей почты настроен правильно, то на указанную почту придет письмо об успешном тестировании соединения. Введенные настройки сервера исходящей почты сохраняются в любом случае, независимо от статуса подключения

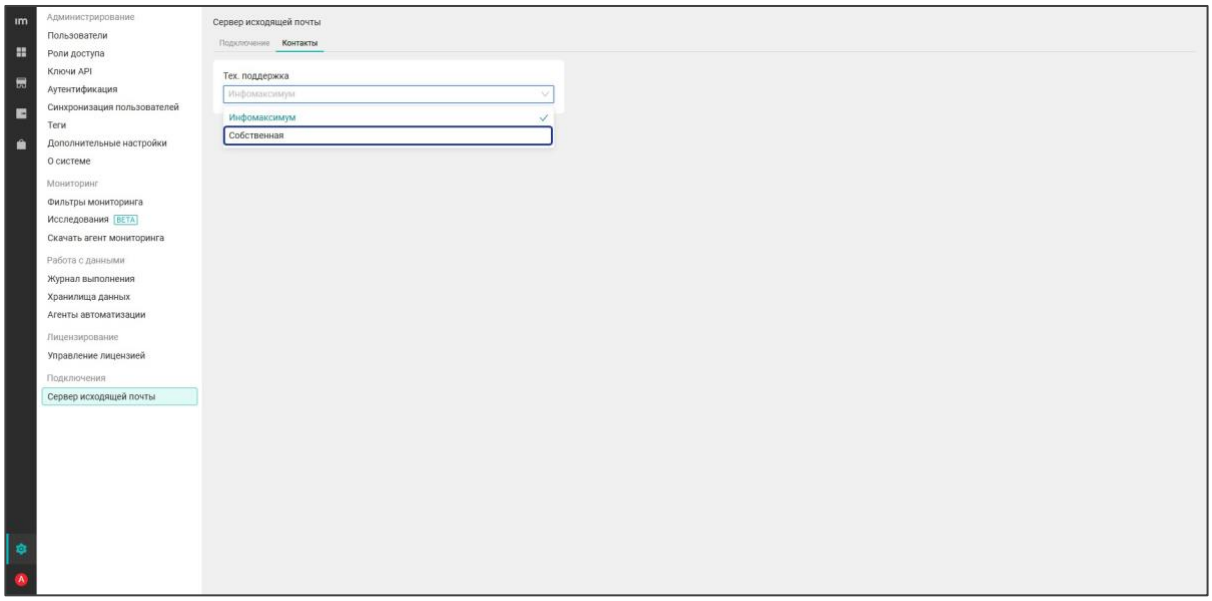
- **Проверить подключение** — происходит только тестирование соединения без сохранения настроек. Результат проверки выводится в виде сообщения

Контакты технической поддержки

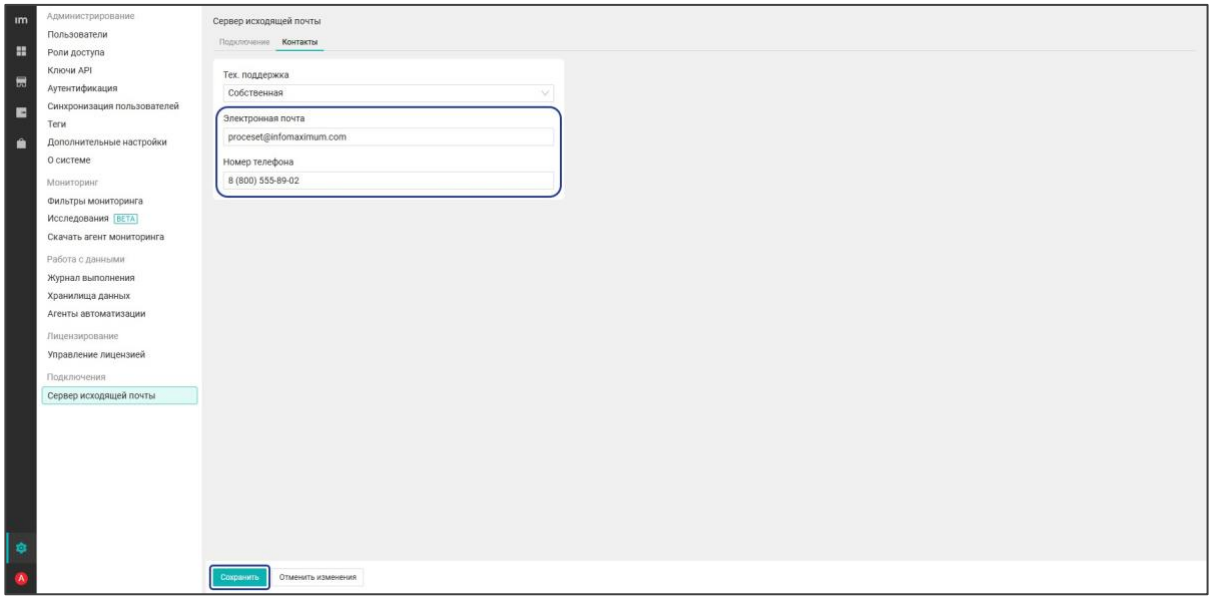
Во вкладке *Контакты* задаются контакты технической поддержки. По умолчанию в контактах тех. поддержки выбраны электронная почта и номер телефона компании Infomaximum.



Чтобы изменить контакты тех. поддержки, выберите **Собственная** в поле **Тех. поддержка**.



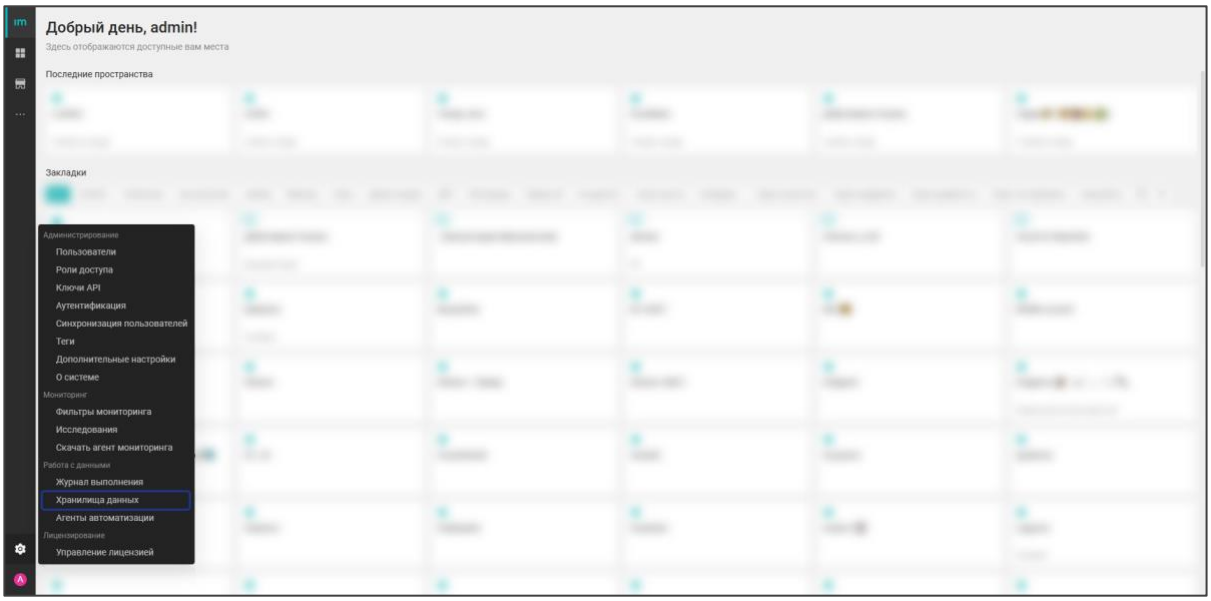
Введите электронную почту, номер телефона и нажмите **Сохранить**.



Примечание. При выборе значения **Собственная** поля **Электронная почта** и **Номер телефона** являются обязательными для заполнения.

Хранилища данных

На странице *Хранилища данных* находится список серверов ClickHouse.



Чтобы добавить новое подключение, нажмите + **Добавить** в правом верхнем углу и введите:

- Название
- Кластерный режим (Вкл/Выкл)
- Хост
- Порт
- Имя пользователя
- Пароль
- SSL (Вкл/Выкл)

В хранилище данных могут быть добавлены как валидные, так и невалидные соединения. Чтобы проверить соединение, перейдите в его профиль и кликните **Проверить подключение**. Если проверка завершится неудачно, появится сообщение об ошибке.

Первое подключенное хранилище используется для синхронизации данных, собранных агентом мониторинга. Чтобы выбрать новое хранилище на ОС Windows, в конфигурационном файле *com.infomaximum.subsystem.monitoring.json* в значении параметра storage_guid укажите идентификатор интересующего подключения, а затем перезапустите контейнер. Чтобы изменить GUID хранилища на ОС Linux, при обновлении или перезапуске контейнера передайте новое значение переменной окружения MN_STORAGE_GUID.

Доступ к списку серверов, возможности добавления и редактирования новых подключений регламентируются привилегией *Хранилища*.

Профиль сервера ClickHouse

Чтобы зайти в профиль сервера ClickHouse, кликните по его названию или нажмите значок контекстного меню и выберите пункт **Редактировать**.

Вкладка Основное

Во вкладке Основное можно настроить параметры сервера:

- Название соединения
- Хост
- Порт
- Имя пользователя
- Пароль
- SSL
- Аутентификация:
  - Стандартная (по умолчанию)
  - Сквозная (логин/пароль)
- Лимит памяти на запрос в скриптах (МБ)
- Лимит памяти на запрос в дашбордах (МБ)
- Количество одновременных подключений
- Размер очереди на выполнение
- Время сессии (в минутах)

Во вкладке также возможно **Проверить подключение**.

Сохранение подключения доступно по кнопке в левом нижнем углу, когда заполнены все обязательные поля.

После сохранения автоматически проводится проверка подключения.

Примечание. Невозможно удалить подключение, которое используется.

Сквозная аутентификация

Примечание. Эта функция доступна в бета-версии. Попробуйте ее в работе и поделитесь своим мнением — обратная связь помогает нам развивать продукт.

Сквозная аутентификация — это вид аутентификации, с помощью которого для пользователей настраивается построчный доступ к данным в модуле «Бизнес-аналитика». Сквозная аутентификация связывает между собой ClickHouse и Proceset, позволяя создавать и синхронизировать права доступа для каждой учетной записи Proceset и ClickHouse. Таким

образом происходит проверка и сопоставление учетных записей, а также отображение только доступной пользователю информации из базы данных.

В режиме стандартной аутентификации обращение к ClickHouse для всех пользователей происходит под технической учетной записью. При сквозной аутентификации обращение к ClickHouse происходит от имени соответствующей учетной записи security-пользователя. Учетная запись security-пользователя необходима для создания пользователей и настройки доступов.

Примечание.

- Чтобы автоматически массово синхронизировать доступы Proceset, в ClickHouse добавлена проверка подключения сквозной аутентификации и наличия пространств, использующих это подключение. Если пространства уже существуют, переключиться на сквозную аутентификацию невозможно.
- Для работы сквозной аутентификации требуется отдельное хранилище ClickHouse, отличное от используемого в режиме стандартной аутентификации.

Построчный доступ

Построчные доступы для пользователей предоставляются через **Инструмент раздачи построчного доступа**, доступный по запросу.

Построчный доступ представляет из себя настройку доступов на основе атрибутов пользователей.

Получение атрибутов происходит через системные таблицы Proceset. Данные из системы синхронизируются каждые 5 минут. Атрибуты включают:

- Доступы к сотрудникам
- Роли доступа
- Логин
- Пользовательские поля

Также можно добавить атрибуты из загруженного в базу данных справочника.

Совет. Перед использованием в продакшн-среде обязательно включите функционал построчного доступа (RLS) в тестовой копии базы данных ClickHouse. Это поможет заранее выявить и устранить возможные ошибки при чтении данных и построении отчетов.

Пример использования построчного доступа

В BI-системе можно настроить отчеты так, чтобы каждый пользователь видел только доступные ему данные на уровне определенной БД. Это особенно важно при работе с конфиденциальной информацией: отчетами по эффективности сотрудников, задачам, продажам и т.д.

Для настройки построчного доступа необходимо:

1. Настроить таблицу прав доступа, где права доступа будут распределены, например, по логину, роли или отделу.

2. Связать эту таблицу с основными данными через BI-систему или на уровне источника данных.

3. Ограничить доступ к таблицам. Убедитесь, что пользователи имеют доступ только к тем источникам данных, с которыми они должны работать.

Система автоматически определяет пользователя, открывшего отчет, и применяет необходимый фильтр. Например:

- Сотрудник — отображаются только свои данные
- Руководитель отдела — отображаются данные своей команды
- HR или руководство — отображаются данные сотрудников всей компании

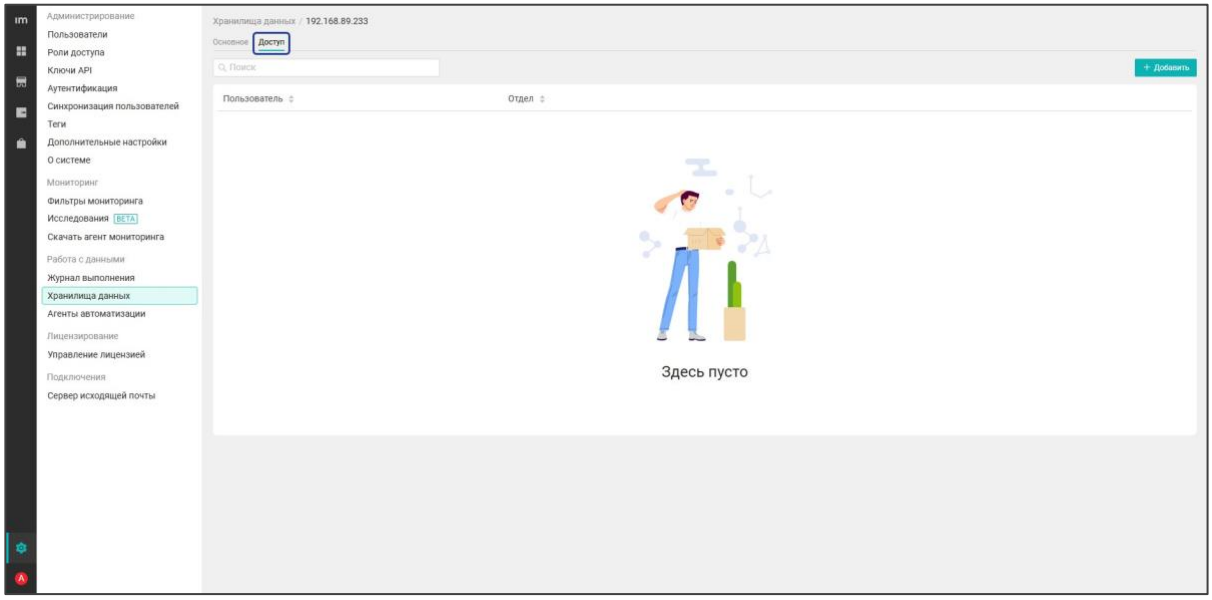
Примечание. Все пользователи работают с одним и тем же отчетом, но содержимое подстраивается под их роль.

Преимущества использования построчного доступа:

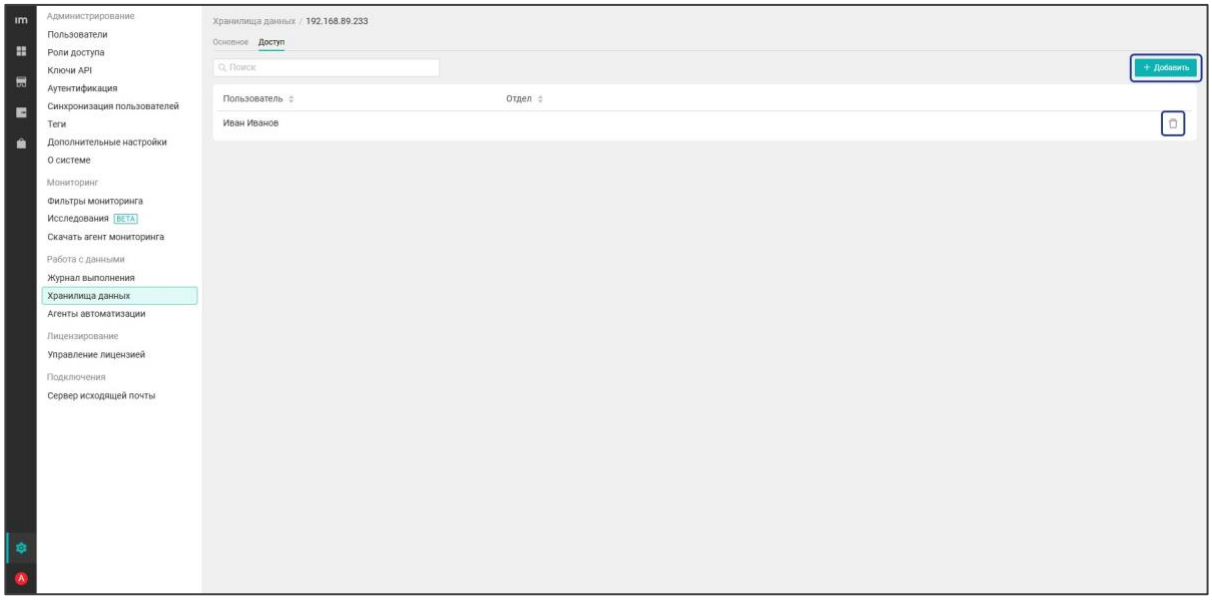
- Безопасность — никто не увидит чужие данные
- Гибкость — несколько видов доступа для одного отчета
- Простота поддержки — не требуется создавать отдельные отчеты для каждого отдела или роли
- Масштабируемость — простой процесс добавления новых сотрудников и ролей

Вкладка Доступы

Во вкладке *Доступы* настраивается доступ для пользователей на создание пространств на выбранном сервере. Для предоставления доступа пользователю также должна быть назначена привилегия *Пространство* с операцией **Создание (C)**.



Чтобы предоставить пользователям доступ к серверу, нажмите **+ Добавить**, выберите сотрудников из списка и нажмите **Сохранить**. Чтобы запретить сотруднику доступ к серверу, нажмите иконку корзины.

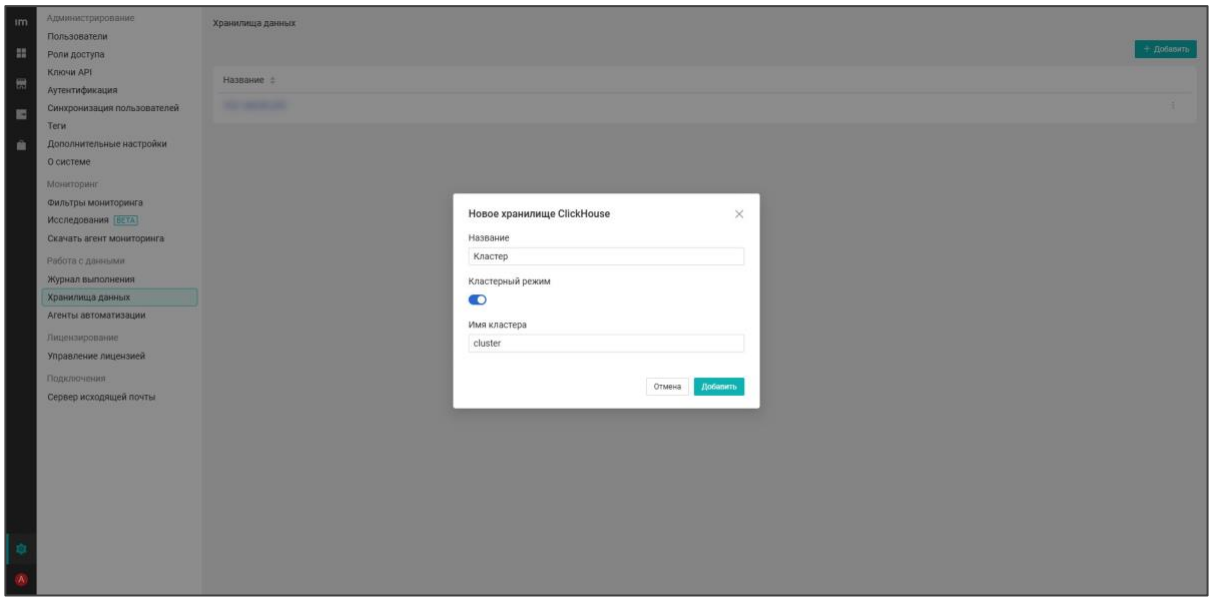


Кластерный режим

Кластерный режим позволяет указать несколько серверов ClickHouse в одном подключении. Это помогает распределить нагрузку с одного сервера на несколько.

Чтобы создать подключение с кластерным режимом, необходимо при добавлении подключения включить соответствующий переключатель и указать:

- Название подключения
- Имя кластера



Во вкладке *Основное* отображается:

- Название сервера
- Имя кластера
- Аутентификация
- Время сессии (минуты)

Во вкладке **Соединения** отображается список подключенных серверов. Каждое соединение настраивается отдельно через кнопку **+ Добавить**.

Соединения удаляются без ограничений.

При выборе режима аутентификации **Сквозная (логин/пароль)** доступны следующие настройки:

- Управление пользователями (только автоматически)
- Security-пользователь
    - Имя security-пользователя
    - Пароль security-пользователя

Во вкладке *Доступ* назначается доступ для пользователей к хранилищу данных (к серверам). При настроенном доступе пользователь может взаимодействовать с пространствами, которые относятся к этому хранилищу. Настройка доступа аналогична той, которая применяется для обычных подключений. При использовании сквозной аутентификации для выбранных пользователей из списка можно разрешить просматривать базу данных пространства, отметив галочкой соответствующий пункт внизу страницы.

Рекомендации по работе с ClickHouse в кластерном режиме

Чтобы данные дублировались по всем нодам ClickHouse, используйте в кластерном режиме 1 шард и много реплик.

Также необходимо использовать движки семейства Replicated.

В кластерном режиме много соединений в рамках одного подключения, и при запросе на дашборд запросы будут распределяться по этим соединениям. Поэтому при создании, удалении, переименовании таблицы через SQL-запрос необходимо прописывать в тексте запроса «ON CLUSTER cluster_name».

Пример:

```
CREATE TABLE table_name ON CLUSTER cluster_name
DROP TABLE table_name ON CLUSTER cluster_name
RENAME TO table_name ON CLUSTER cluster_name
TRUNCATE TABLE table_name ON CLUSTER cluster_name
```

Если не прописать данный параметр, то запрос выполнится в отдельно взятой ноде, на других нодах изменения не применятся.

При пересоздании ранее удаленной таблицы на движке ReplicatedReplacingMergeTree может возникнуть ошибка. Ее можно обойти с помощью передачи дополнительных параметров для zookeeper: zoo_path — путь до таблицы в ClickHouse Keeper.

Пример:

```
ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/{database}/table_name_xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx', '{replica}')
```

Особенности работы Карты процесса при использовании кластерного режима

При работе с Картой процесса могут возникать ошибки, если используется кластерный режим с балансировщиками. Чтобы избежать ошибок, в конфигурационном файле *com.infomaximum.subsystem.bidata.json установите для флага "use_clickhouse_balancer"*

значение true. В таком случае **Карта процесса** будет работать и отображаться корректно, но медленнее, чем обычно.

Если работать с **Картой процесса** не требуется, рекомендуем установить для флага значение false — скорость выполнения расчетов и отображения виджетов увеличится.

Подключение к ClickHouse в пространстве

О подключении ClickHouse в пространстве смотрите на странице **Подключения**.

Администрирование

Раздел содержит материалы по администрированию системы Proceset: управлению доступом и пользователями, лицензированию, работе с конфигурацией и базами данных, а также другим административным задачам.

Типы релизов

Для системы Proceset предусмотрены два типа релизов новых версий:

- *LTS (Long-Term Support)* — версии с длительным сроком поддержки, предназначенные для стабильного использования в течение продолжительного времени
- *Regular* — версии с более коротким сроком поддержки, предоставляющие ранний доступ к новым функциям

На этой странице описаны особенности каждого типа релизов, сроки их поддержки и рекомендации по выбору версии для различных сценариев использования.

LTS-релизы

LTS-релизы выходят по фиксированному графику четыре раза в год: в марте, июне, сентябре и декабре.

Каждая LTS-версия поддерживается в течение 12 месяцев. В этот период для версии предоставляются:

- Обновления безопасности
- Исправления ошибок
- Техническая поддержка

LTS-сборки проходят расширенное тестирование и проверку на совместимость.

Regular-релизы

Regular-версии также выпускаются регулярно. Они подходят для пользователей, которым важно оперативно получать новые функции.

Особенности промежуточных версий:

- Ранний доступ к новым возможностям, включая экспериментальные функции
- Сокращенный срок поддержки — до 6 месяцев
- Объем тестирования может быть меньше по сравнению с LTS-релизами

Рекомендации по выбору

Для стабильной и долгосрочной эксплуатации рекомендуется устанавливать LTS-версии. Фиксированный график выпусков и поддержки позволяет планировать внутренние процессы и снижать риски при обновлении.

Если вам важно получать новые функции как можно раньше, выбирайте Regular-версии, но учитывайте их ограничения по сроку поддержки и уровню стабильности.

Лицензирование системы

Система лицензирования Proceset предоставляет улучшенный контроль за лицензиями, упрощает их назначение, переназначение и предоставляет более подробные данные об их использовании. Всего в Proceset 8 типов лицензий, 6 из которых определяют доступы к функционалу бизнес-аналитики, а 2 — к мониторингу действий пользователей:

- Бизнес-пользователь базовый
- Бизнес-пользователь процессный
- Бизнес-пользователь мультипроцессный
- Аналитик базовый
- Аналитик процессный
- Аналитик мультипроцессный
- Мониторинг базовый
- Мониторинг расширенный

Бизнес-пользователь базовый

Возможности

Бизнес-пользователь базовый может просматривать все дашборды и виджеты внутри них кроме **Карты процесса**, **Сферы процессов**, **Воронки**, а также изменять все виджеты. Может просматривать и изменять все скрипты.

Ограничения

Бизнес-пользователь базовый не может опубликовать дашборды и скрипты.

Поведение системы при наличии лицензии

При просмотре дашборда, если на страницу добавлены **Карта процесса**, **Сфера процессов**, **Воронка**, бизнес-пользователь базовый видит занимаемое место этими виджетами. Вместо контента в них отображаются сообщения об отсутствии лицензии на просмотр виджета.

При попытке опубликовать дашборд или скрипт возникает ошибка, сообщающая об отсутствии лицензии на данные действия.

Бизнес-пользователь процессный

Возможности

Бизнес-пользователь процессный может просматривать все дашборды и виджеты внутри них, кроме **Сферы процессов**, **Воронки**, а также изменять все виджеты. Может просматривать и изменять все скрипты и блоки внутри них.

Ограничения

Бизнес-пользователь процессный не может опубликовать дашборды и скрипты.

Поведение системы при наличии лицензии

При просмотре дашборда, если на страницу добавлены **Сфера процессов**, **Воронка**, бизнес-пользователь процессный видит занимаемое место этими виджетами. Вместо контента в них отображаются сообщения об отсутствии лицензии на просмотр виджета.

При попытке опубликовать дашборд или скрипт возникает ошибка, сообщающая об отсутствии лицензии на данные действия.

Бизнес-пользователь мультипроцессный

Возможности

Бизнес-пользователь мультипроцессный может просматривать и изменять все дашборды и виджеты внутри них. Может просматривать и изменять все скрипты и блоки внутри них.

Ограничения

Бизнес-пользователь мультипроцессный не может опубликовать дашборды и скрипты.

Поведение системы при наличии лицензии

При попытке опубликовать дашборд или скрипт возникает ошибка, сообщающая об отсутствии лицензии на данные действия.

Матрица доступов к функционалу бизнес-пользователя

В матрице представлен доступный функционал в зависимости от типа лицензии бизнес-пользователя. Наличие доступа к функционалу отмечено ✓, отсутствие доступа — ✕.

Доступный функционал		Бизнес-пользователь		
		Базовый	Процессный	Мультипроцессный
Просмотр дашбордов Дополнительно требуется одно из условий: Привилегия Пространство R Доступ на просмотр конкретного пространства	Прочие виджеты	✓	✓	✓
	Карта процесса	✕	✓	✓
	Сфера процесса	✕	✕	✓
	Воронка	✕	✕	✓
Публикация дашбордов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства		✕	✕	✕
Просмотр скриптов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства	Все блоки	✓	✓	✓
Публикация скриптов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства		✕	✕	✕

Аналитик базовый

Возможности

Аналитик базовый может просматривать дашборды и виджеты внутри них, кроме **Карты процесса**, **Сферы процессов**, **Воронки**, а также изменять все виджеты. Может просматривать и изменять все скрипты и блоки внутри них.

Ограничения

Для публикации доступны дашборды, в которых нет **Карты процесса**, **Сферы процессов**, **Воронки**.

Поведение системы при наличии лицензии

При просмотре дашборда, если на страницу добавлены **Карта процесса**, **Сфера процессов**, **Воронка**, аналитик базовый видит занимаемое место этими виджетами. Вместо контента в них отображаются сообщения об отсутствии лицензии на просмотр виджета.

При попытке опубликовать дашборд с **Картой процесса**, **Сферой процессов**, **Воронкой** возникает ошибка, сообщающая об отсутствии лицензии на данное действие.

Аналитик процессный

Возможности

Аналитик процессный может просматривать все дашборды и виджеты внутри них, кроме **Сферы процессов**, **Воронки**, а также изменять все виджеты. Может просматривать, изменять и публиковать все скрипты.

Ограничения

Для публикации доступны дашборды, в которых нет **Сферы процессов**, **Воронки**.

Поведение системы при наличии лицензии

При просмотре дашборда, если на страницу добавлены **Сфера процессов** и **Воронка**, аналитик процессный видит занимаемое место этими виджетами. Вместо контента в них отображаются сообщения об отсутствии лицензии на просмотр виджета.

Аналитик процессный не может опубликовать дашборды При попытке опубликовать дашборд со **Сферой процессов** и **Воронкой** возникает ошибка, сообщающая об отсутствии лицензии на данное действие.

Аналитик мультипроцессный

Возможности

Аналитик мультипроцессный может просматривать и изменять все дашборды и виджеты внутри них, а также просматривать и изменять все скрипты и блоки внутри них. Может опубликовать все дашборды и скрипты. Ограничений нет.

Поведение системы при наличии лицензии

Так как для лицензии доступен весь функционал бизнес-аналитики, особенностей в поведении нет.

Матрица доступов к функционалу аналитика

В матрице представлен доступный функционал в зависимости от типа лицензии аналитика. Наличие доступа к функционалу отмечено ✓, отсутствие доступа — ✕.

Доступный функционал		**Аналитик**		
		Базовый	Процессный	Мультипроцессный
Просмотр дашбордов Дополнительно требуется одно из условий: Привилегия Пространство R Доступ на просмотр конкретного пространства	Прочие виджеты	✓	✓	✓
	Карта процесса	✕	✓	✓
	Сфера процесса	✕	✕	✓
	Воронка	✕	✕	✓
Публикация дашбордов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства	Прочие виджеты	✓	✓	✓
	Карта процесса	✕	✓	✓
	Сфера процесса	✕	✕	✓
	Воронка	✕	✕	✓
Просмотр скриптов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства	Все блоки	✓	✓	✓
Публикация скриптов Дополнительно требуется одно из условий: Привилегия Пространство W Доступ на изменение конкретного пространства	Все блоки	✓	✓	✓

Мониторинг базовый

Лицензия представляет собой сбор статистики активности сотрудника, включая клики мышкой, переходы по вкладкам и другие пользовательские действия. Подробнее о сборе информации описано на странице Мониторинг.

Мониторинг расширенный

Лицензия позволяет не только собирать статистику по активности, но и данные форм, если они присутствуют. Подробнее о сборе информации описано на странице **Мониторинг**.

Лицензионный ключ

Чтобы активировать систему, необходимо ввести лицензионный ключ. Примененный в системе ключ ограничивает срок доступа к продукту до определенного числа, указанного при генерации ключа. Также ключ может быть бессрочным. При создании лицензии задаются типы лицензии:

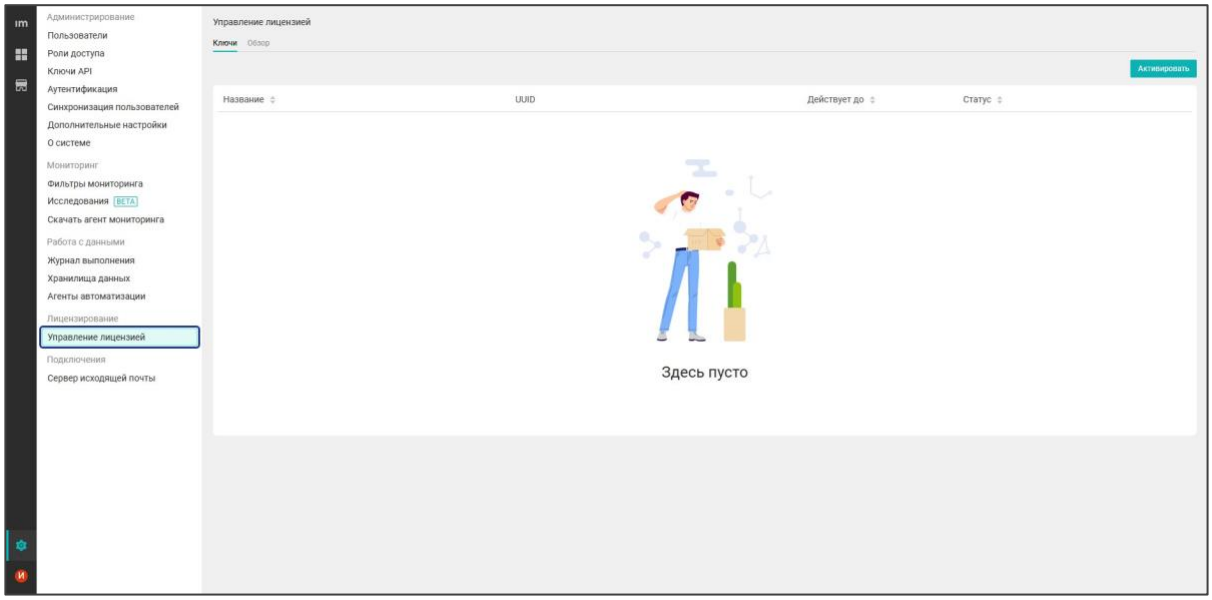
- Бизнес-пользователь базовый
- Бизнес-пользователь процессный
- Бизнес-пользователь мультипроцессный
- Аналитик базовый
- Аналитик процессный
- Аналитик мультипроцессный

В зависимости от типа лицензии пользователю доступны определенные возможности в системе, которые описаны на странице Лицензирование системы.

Лицензионный ключ предоставляют сотрудники Инфомаксимум. Его можно запросить по номеру телефона **8 (800) 555-89-02** или по адресу электронной почты **support@infomaximum.com**.

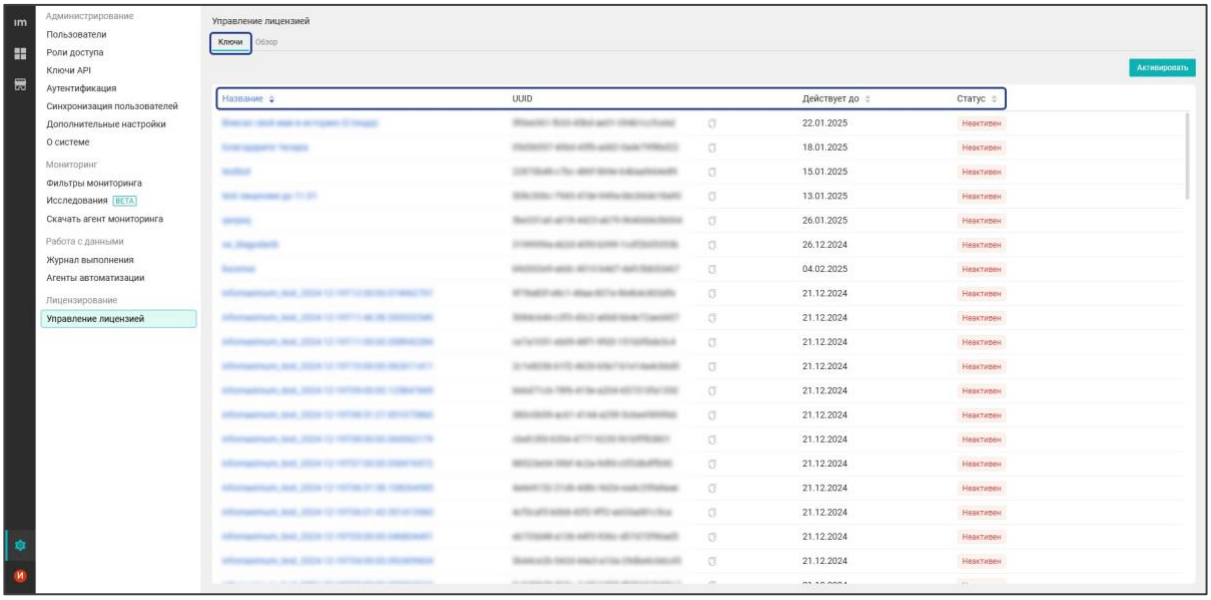
Активация ключа

Активация ключа происходит в разделе *Лицензирование*. Раздел находится во вкладке *Настройки* и доступен только пользователям с ролью доступа **Администратор**.

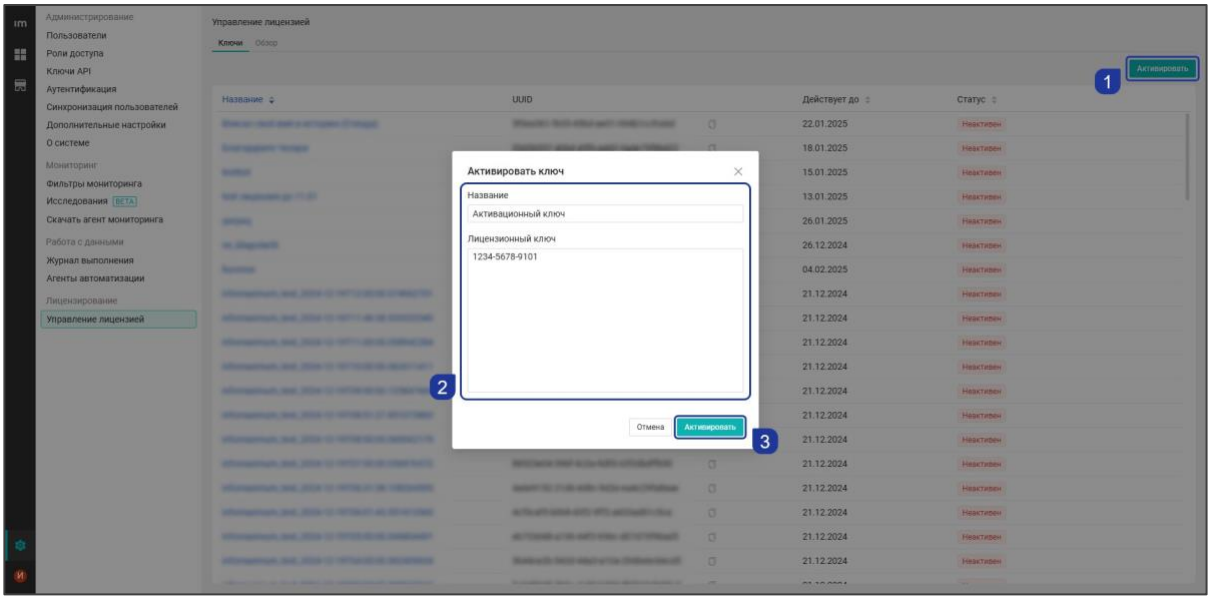


При переходе в *Управление лицензией* открывается вкладка *Ключи*. Во вкладке отображаются:

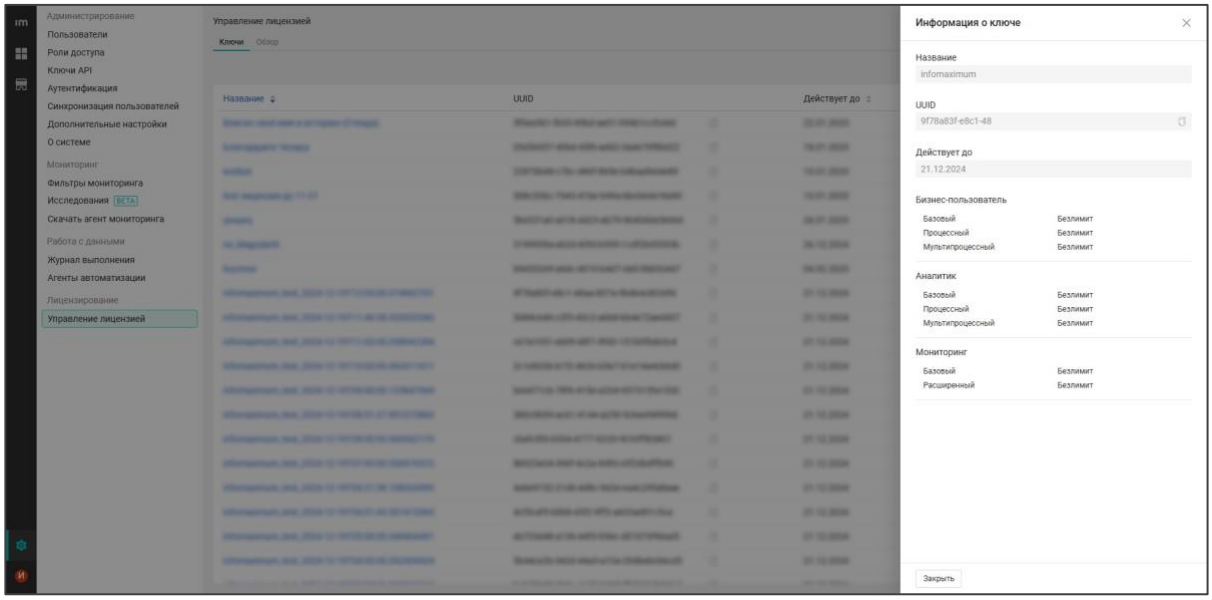
- Название ключа — можно указать любое
- UUID — идентификатор ключа
- Действует до — дата последнего дня действия лицензии
- Статус — при активации ключа статус становится *Активен*. Если время действии лицензии истекло, ключ получает статус *Неактивен*



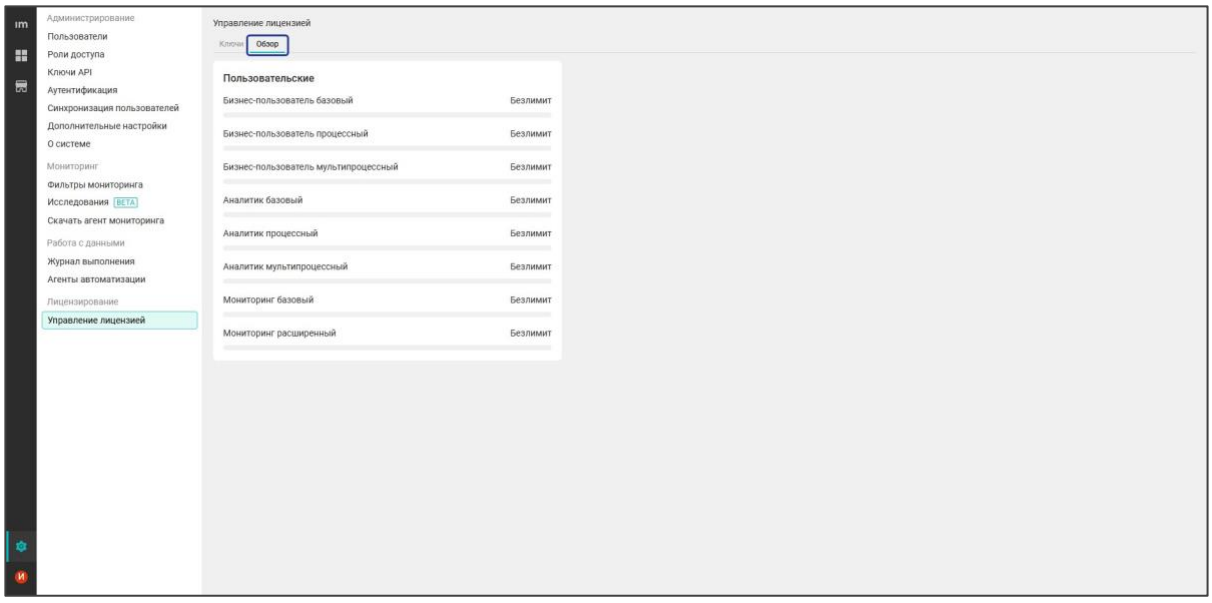
Для активации ключа нажмите соответствующую кнопку в правом верхнем углу. В появившемся окне введите название и предоставленный ключ, затем нажмите **Активировать**.



Чтобы узнать количество лицензий, доступных для назначения пользователям в рамках одного ключа, кликните по его названию. В открывшемся боковом окне представлен список лицензий и их доступное количество.



Во вкладке *Обзор* отображаются приобретенные лицензии и их суммированное количество от всех активированных ключей. Если лицензия закончилась, то напротив названия типа лицензии отображается 0. Если для ключа предоставлено неограниченное количество определенной лицензии, напротив названия типа лицензии указано «Безлимит».



Примечание.

- Если срок действия ключа истекает в то время, когда пользователь находится в системе, ему перестает быть доступен лицензируемый функционал до тех пор, пока не будет активирован новый ключ.
- Если необходимо продлить действие лицензии до завершения срока текущей, обратитесь к менеджеру Инфомаксимум или к технической поддержке. Вам пришлют новый лицензионный ключ. При применении нового ключа продлевается срок лицензии до даты, указанной при его генерации.
- Если срок действия ключа истек, заданная пользователю лицензия сохраняется в настройках его профиля.

Назначение лицензий

Назначение лицензий возможно через массовое назначение или через профиль конкретного пользователя.

Лицензию на мониторинг можно назначить пользователям с помощью GraphQL-запроса:

```
mutation {
  employee {
    update_employee_license_roles(
      target_all: false
      target_employee_ids: []
      target_department_ids: []
      license_roles_to_add: []
      license_roles_to_remove: []
    )
  }
}
```

В параметре license_roles_to_add укажите:

- MONITORING_SIMPLE — для назначения пользователю лицензии на базовый мониторинг
- MONITORING_EXTENDED — для назначения пользователю лицензии на расширенный мониторинг

В параметре target_employee_ids укажите идентификатор пользователя, в параметре target_department_ids — идентификатор отдела.

Примечание. В параметрах target_employee_ids, target_department_ids, license_roles_to_add и license_roles_to_remove в квадратных скобках можно указать несколько значений одновременно.

Бета-версия функциональности

Некоторые функции доступны в бета-версии и временно предоставляются без дополнительной лицензии.

Примечание.

- Эти функции могут быть изменены или удалены в будущих версиях продукта.
- В будущем для доступа может понадобиться отдельная лицензия или специальный тариф.
- Не рекомендуется использовать бета-функциональность в продуктивной среде без предварительной оценки стабильности и влияния на бизнес-процессы.

Предлагаем протестировать бета-функциональность и будем благодарны за обратную связь. Ваш отзыв поможет нам сделать продукт лучше.

Работа с базами данных

Резервное копирование БД

Резервное копирование встроенной файловой базы данных

Сервер приложения Proceset раз в сутки в 00:00:00 выполняет резервное копирование (бэкап встроенной файловой базы данных). Резервное копирование по умолчанию осуществляется в системную папку: *C:/ProgramData/Infomaximum/backup*. Каталог системной папки может быть изменен.

Параметры встроенной файловой базы данных настраиваются в файле: *com.infomaximum.platform.component.database.json*.

Путь к файлу по умолчанию:

C:/ProgramData/Infomaximum/config/com.infomaximum.platform.component.database.json, где:

- "periodical_backup_enabled":true/false — значение, которое определяет, делается ли периодический бэкап или непериодический
- "backup_path" — расположение бэкапа базы данных (относительно папки ProgramData)

Резервное копирование ClickHouse (Docker-контейнера)

Резервное копирование базы данных ClickHouse не предусмотрено штатными средствами системы. Резервное копирование данных, которые хранятся в ClickHouse, возможно на уровне Docker-контейнера. Копируется весь volume, в котором расположена БД ClickHouse.

Важно. Перед началом создания бэкапа необходимо убедиться в наличии образа Ubuntu 20.04 в системе.

При наличии интернета его можно загрузить, выполнив команду:

```
# docker pull ubuntu:20.04
```

Процедура резервного копирования:

1. Создайте новую папку для сохранения бэкапа:

```
mkdir /tmp/clickhouse-backup
```

2. Удалите сервис:

```
docker service rm infomaximum-clickhouse
```

3. Запустите новый контейнер с прокидыванием volume:

```
docker run -it --rm --mount source=infomaximum-clickhouse,target=/clickhouse -v /tmp:/target ubuntu:20.04 bash
```

4. Внутри контейнера выполните:

```
tar -czvf /target/infomaximum-clickhouse.tar.gz /clickhouse
```

5. Выйдите из контейнера:

```
exit
```

6. Повторно создайте сервис. Предварительно скачайте необходимый образ (ссылка предоставляется дополнительно). Выполните команду:

```
docker load -i infomaximum-clickhouse-(название файла)tar.gz
```

7. Созданный бекап расположен:

- /tmp/infomaximum-clickhouse.tar.gz

Получение данных пользовательской активности

Для получения данных пользовательской активности воспользуйтесь инструментами автоматизации.

Контроль целостности баз данных

Контроль целостности встроенной файловой базы данных по контрольным суммам осуществляется при каждой загрузке системы. В процессе работы контроль целостности БД осуществляться не может. В состав системы встроены средства отладки, которые невозможно удалить, но можно контролировать их активацию. Для этого необходимо отслеживать все изменения в службе **«Infomaximum»**, в частности команду запуска.

Контроль целостности БД ClickHouse входит в поставку.

Сохранение копии обезличенной базы данных

Для сохранения обезличенной БД системы необходимо, чтобы у пользователя была назначена роль доступа с включенной привилегией *«Общие настройки»* с операцией **W** (изменение).

Чтобы сохранить обезличенную БД, войдите через GraphQL при выполнении следующего запроса и укажите путь (директорию) для запуска процесса, куда сохранится обезличенная БД.

```
mutation{
   database{
      copy_depersonalized_database(path:«c:/database»)
   }
}
```

Где *«c:/database»* — путь сохранения обезличенной БД.

Для выполнения запроса GraphQL у пользователя должна быть назначена роль доступа с включенной привилегией *«Инструмент GraphQL»* с операцией **E**.

Сохранение происходит на сервер, где установлена система. Папка для сохранения создается заранее и должна быть пустой.

Работа с системой

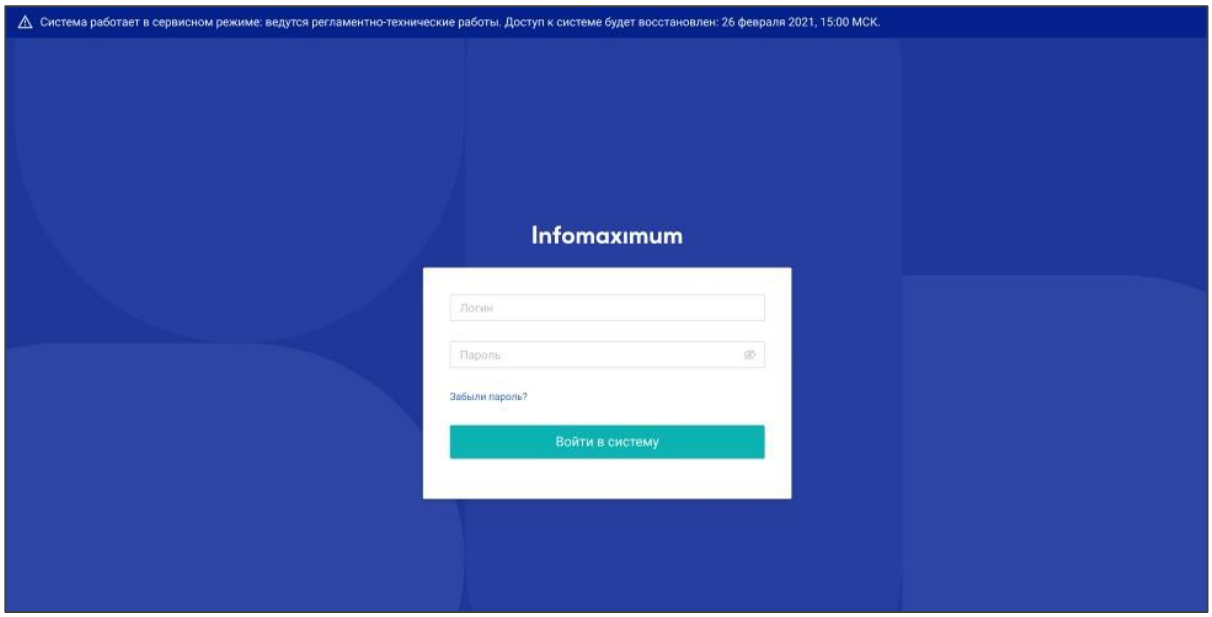
Сервисный режим системы

Сервисный режим функционирования используется для выполнения операций подготовки и проведения регламентов, испытаний или значительной перестройки системы. В данном режиме также осуществляется техническое обслуживание, реконфигурация, модернизация системы. Режим позволяет проводить диагностирование инцидентов или проблем, связанных со сбоями или авариями в работе системы.

Включение/выключение сервисного режима осуществляется в файле: com.infomaximum.subsystem.frontend.json в параметре "service_mode": true или false.

Если «Сервисный режим» включён, то при переходе в веб-интерфейс системы пользователь видит форму логина и информационное окно вверху страницы: «Система работает в сервисном режиме: ведутся регламентно-технические работы».

Текст, который отображается у пользователей при включённом сервисном режиме, можно отредактировать в этом же конфигурационном файле com.infomaximum.subsystem.frontend.json в параметре "service_mode_message":"Текст сообщения". В тексте можно использовать язык разметки Markdown. Подробнее особенности информационного окна и возможности Markdown описаны на странице Настройка системы.



Авторизация пользователя может произойти только в том случае, если у пользователя роль доступа, в которой значение привилегии *«Сервисный режим»* в состоянии **R** (по умолчанию данная привилегия включена у роли доступа *«Администратор»*).

Пользователи, у которых роль доступа без привилегии *«Сервисный режим»*, не смогут авторизоваться, если включен *«Сервисный режим»*. При отключении состояния *«Сервисный режим»* Система работает стационарно, не проверяет наличие привилегии *«Сервисный режим»* при авторизации пользователей.

Важно.

- При включенном сервисном режиме не принимается активность от агентов мониторинга, что позволяет не переключать порт сервера системы во время обновления.
- Если сотрудники работают с интерфейсом системы и в этот момент включается Сервисный режим, всех пользователей выбрасывает из системы на страницу авторизации.

Настройка системного времени

Синхронизация системного времени происходит между модулями ММАП (Агент мониторинга) и МНиАО. Системное время в модулях принимает значение, установленное на сервере. Система получает значение времени в формате Unix от операционной системы, на которой развернут сервер. Сервером является ПК или специализированное оборудование, на котором развернута система. Настройка и изменение системного времени регламентируется посредством политики Windows.

Получение системного времени

Для просмотра системного времени, установленного в системе, необходимо воспользоваться инструментом GraphQL.

Сформируйте запрос через веб-интерфейс:

```
{
  server{
    time
  }
}
```

Получится значение time:

```
{
  "data": {
    "server": {
      "time": 1690181560211
    }
  }
}
```

Значение time представлено в формате Unix. Все события записываются в журнал безопасности и базу данных в локальном времени сервера.

Конфигурационные файлы системы

Некоторые настройки системы можно изменять через конфигурационные файлы или переменные окружения, в зависимости от операционной системы:

- Windows — изменения вносятся в конфигурационные файлы, расположенные по умолчанию в каталоге `C:\ProgramData\Infomaximum\config`

Примечание: Путь к файлам может отличаться, если система установлена на другой диск или в другой каталог.

- Linux — настройки задаются через переменные окружения при запуске Docker-контейнера

Ниже приведены основные файлы и параметры и переменные окружения, которые можно изменить. Обратите внимание, что некоторые переменные окружения не связаны с конкретными конфигурационными файлами.

Пример передачи параметров на Linux:

```
-e JVM_MAX_MEMORY='4G'
```

Конфигурационные файлы

Ниже приведены основные файлы и параметры, которые можно изменить, а также соответствующие им переменные окружения при их наличии.

com.infomaximum.subsystem.frontend.json

Параметр	Переменная окружения	Описание
"protocol"	—	Протокол веб-интерфейса — HTTP или HTTPS
"port"	—	Порт, на котором будет работать веб-интерфейс
"ssl_cert_store"	—	Путь к PFX-файлу с сертификатом и закрытым ключом для работы по протоколу HTTPS. Поддерживаемые форматы: JKS, PKCS#12 Косую черту необходимо экранировать еще одной косой чертой
"ssl_cert_store_password"	—	Пароль от PFX-файла Если пароль пустой, укажите ""
"trust_store"	—	Полный путь к файлу хранилища доверенных сертификатов. Поддерживаемые форматы: JKS, PKCS#12
"trust_store_password"	—	Пароль хранилища доверенных сертификатов Если пароли хранилища сертификатов и хранилища доверенных сертификатов одинаковые, то это поле можно не заполнять
"crl"	—	Полный путь к файлу отозванных сертификатов (необязательное)

Параметр	Переменная окружения	Описание
"url"	FE_URL	Адрес сервера Proceset Необходимо указать полностью, с протоколом и портом Используется при создании системой всех ссылок, которые в нее ведут. Например, на основе параметра создаются ссылки в рассылаемых почтовых сообщениях для сброса пароля и приглашения
"service_mode"	FE_SERVICE_MODE	Включение/выключение сервисного режима По умолчанию выключен. Чтобы включить, укажите true Подробнее в разделе Работа с системой
"service_mode_message"	FE_SERVICE_MODE_MESSAGE	Сообщение, которое отображается у пользователя на странице авторизации при включенном сервисном режиме В сообщении можно использовать язык разметки Markdown
"session_timeout"	FE_SESSION_TIMEOUT	Время жизни пользовательской сессии По умолчанию период неактивной сессии для каждого пользователя составляет 7 дней Можно указать в часах (h) или днях (d)
"support_iframe"	—	Позволяет встроить интерфейс системы Proceset во фрейм на стороннем сайте Для включения укажите значение true По умолчанию указано false — возможность выключена
"web_dir"	—	Путь к каталогу с файлами веб-интерфейса относительно рабочего каталога Proceset
"add_exclude_cipher_suites"	—	Исключенные алгоритмы шифрования
"host"	—	Сетевой интерфейс, на котором доступна система По умолчанию — 0.0.0.0, т.е. для всех. Если указано значение 127.0.0.1 — доступно только на локальном хосту
"add_exclude_protocols"	—	Исключенные протоколы SSL-соединения
"request_timeout"	—	Таймаут для HTTP-запроса
"temp_dir"	—	Каталог временных файлов веб-сервера относительно рабочего каталога Proceset
"disable_websocket"	—	Выключение/включение функционала веб-сокетов (подписок) для веб-сервера По умолчанию включено, т.е. false

Параметр	Переменная окружения	Описание
"use_unsupported_applications"	—	Позволяет включить отображение системы в Internet Explorer Для включения укажите true По умолчанию указано false (система не отображается в IE)
"prometheus"	FE_PROMETHEUS	Позволяет включить функционал сбора и предоставления метрик Prometheus Для включения укажите значение true По умолчанию указано false — функционал сбора и предоставления метрик выключен
"cors_policy"	FE_CORS_POLICY	Позволяет настроить CORS для междоменного обмена данными Укажите значение *, чтобы включить междоменный обмен данными с любыми сайтами. Для ограничения доступа перечислите через запятую список URL-адресов, с которыми необходим междоменный обмен данными, например ["https://infomaximum.com", "https://infomaximum.ru"] на Windows или "https://infomaximum.com, https://infomaximum.ru" на Linux Также поддерживается указание маски для всех поддоменов, например, ["https://(.*).infomaximum.com"] на Windows или "https://(.*).infomaximum.com" на Linux Подробнее в разделе JS-трекер
"graphql_disable"	FE_GRAPHQL_DISABLE	Отключение возможности делать GraphQL-запросы
"graphql_introspection_disabled"	FE_GRAPHQL_INTROSPECTION_DISABLE	Позволяет включить интроспекцию GraphQL Чтобы включить, укажите значение false По умолчанию указано true — интроспекция выключена

com.infomaximum.subsystem.core.json

Параметр	Переменная окружения	Описание
"reset_count_invalid_logon_duration"	—	Период, после которого происходит обнуление попыток входа Можно указать в днях, часах, минутах и секундах

Параметр	Переменная окружения	Описание
"restorelink_timeout"	—	Срок жизни ссылки на восстановление пароля По умолчанию срок жизни ссылки на восстановление пароля, которая приходит пользователю на почту, составляет 24 часа Можно указать в днях, часах, минутах и секундах
"secret_key_path"	—	Путь к ключу относительно рабочего каталога Proceset, с помощью которого система шифрует чувствительные данные
"logon_type"	—	Настройка способа входа в систему через встроенную аутентификацию Значение login — вход через логин, email — через почту
"system_notification_message"	CR_NOTIFICATION_MESSAGE	Настройка информационного уведомления, которое отображается после обновления системы Для ввода доступен любой текст, который будет показан в информационном окне Если значение параметра не заполнено или атрибут отсутствует в конфигурационном файле, то будет считаться, что уведомление отключено. В этом случае уведомление не показывается Подробнее в разделе Настройки системы
"employee_tree_depth"	—	Настройка вывода количества сотрудников в списке до «Показать еще» Диапазон находится в пределах от 1 до 100. По умолчанию установлено значение 20. В случае необходимости замените значение вручную Если удалить файл конфигурации модуля и перезапустить службу, то произойдет сброс до значения по умолчанию
"auth_without_role"	CR_AUTH_WITHOUT_ROLE	Настройка аутентификации без роли доступа Если значение отсутствует или выбрано true (по умолчанию), то аутентификация без ролей доступа разрешена, если false — для аутентификации требуется хотя бы одна роль доступа (может использоваться без привилегий) Подробнее в разделе Пользователи

Параметр	Переменная окружения	Описание
"open_org_structure"	CR_OPEN_ORG_STRUCTURE	Переключение режима открытости/закрытости орг. структуры При открытом режиме можно запрашивать список сотрудников без проверки доступа. В профиле сотрудника сняты ограничения на просмотр отдела и доступ к сотрудникам При закрытом режиме запрос списка сотрудников без проверки доступа запрещен. Просмотр отдела в профиле сотрудника возможен только при наличии прав. Добавление сотрудников — через email без поиска по списку (доступен запрос профиля по email) По умолчанию — true

com.infomaximum.subsystem.monitoring.json

Параметр	Переменная окружения	Описание
"phased_agent_upgrade"	—	Позволяет настроить поэтапное обновление агентов Укажите в блоке настройки: — "stable_version" — стабильная версия агента (должна быть указана и проходить валидацию по шаблону XXX.XXX.XXX) — "host_mask" — массив FQDN имен компьютеров в формате регулярных выражений, которые будут обновляться на версию выше, чем указано в параметре "stable_version" (не может быть пустым)
"agent_log_config"	—	Блок настройки уровня технических логов sessionInspector.log на агентах мониторинга В блоке "advanced" можно задать уровень логирования для отдельного пользователя. Для этого укажите id пользователя в квадратных скобках рядом с нужным уровнем Параметр "default_level" позволяет изменить уровень логирования для всех пользователей на сервере. По умолчанию установлено значение — ERROR, но доступны также уровни INFO, WARNING, CRITICAL и OFF Количество собираемых данных зависит от выбранного уровня логирования
"parsing_activity_enabled"	—	Включение/выключение фоновых процессов по разбору и синхронизации архивов активности в ClickHouse

Параметр	Переменная окружения	Описание
"agent_activity_queue_dir"	—	Для переноса папки с сырой активностью на другой диск укажите в параметре новое расположение Косую черту необходимо экранировать еще одной косой чертой
"activity_queue_retention"	—	Срок хранения сырых данных активности Сырые данные необходимы при возникновении ошибки, чтобы после ее исправления загрузить исходные данные повторно Значение по умолчанию — 14 дней (14d), минимальное значение — 1 день (1d)
"partition_life_circle_time"	—	Количество дней хранения логов агентов мониторинга в таблице monitoring_agent_inspector_log По умолчанию 14 дней
"rdb_ch_synchronization"	MN_SYNC_CH_DELAY MN_SYNC_CH_INTERVAL	Настройка синхронизации собранной агентами активности между встроенной файловой базой данных и ClickHouse Параметры первой (после старта системы) и последующих синхронизаций с собранной агентами активностью между встроенной файловой базой данных и ClickHouse { "delay": "1m", "interval": "5m" }, где: "delay" — синхронизация после старта "interval" — период между первой и последующей синхронизациями
"parameters_hashing"	—	Глобальный параметр хеширования, отвечает за включение и отключение хеширования данных из форм ввода, собираемых агентом мониторинга Значение "default" установлено по умолчанию и означает, что хеширование включено, "none" — данные не хешируются и передаются в открытом виде на сервер Любое изменение конфигурации хеширования будет применено после перезагрузки сервера

Параметр	Переменная окружения	Описание
"storage_guid"	MN_STORAGE_GUID	Определяет идентификатор хранилища данных ClickHouse, подключенного к системе Proceset, в котором хранится активность, собранная агентом мониторинга. Определяет, к какому серверу ClickHouse будет подключаться система При изменении значения этого параметра новая активность начнет записываться в новое хранилище. Система автоматически будет переносить накопленные данные в новое хранилище, начиная с самой ранней записи Параметр "storage_guid" используется совместно с параметром "monitoring_database_name", который указывает имя базы данных внутри выбранного хранилища Переменная activity_table хранит актуальный путь к таблице monitoring_activity, используя значения параметров "storage_guid" и "monitoring_database_name". Это позволяет скриптам автоматизации обращаться к данным без необходимости вручную менять путь к таблице при переносе данных
"monitoring_database_name"	MN_DB_NAME	Определяет имя базы данных для хранения активности внутри хранилища ClickHouse ("storage_guid"), где хранится таблица monitoring_activity Если параметр отсутствует или указана пустая строка, используется значение по умолчанию — "main" Если параметр задан, новая активность будет записываться в указанную базу данных Требования к имени базы данных: 1. Длина не более 255 символов 2. Состоит только из латинских букв в верхнем или нижнем регистре, цифр и символов подчеркивания «_» 3. Начинается с буквы, а не с цифры или символа подчеркивания Переменная activity_table формирует актуальный путь к базе данных и таблице monitoring_activity, используя значение этого параметра Имена таблиц monitoring_activity и monitoring_agent_inspector_log не изменяются

Параметр	Переменная окружения	Описание
"researches_enabled"	MN_RESEARCHES	Включение/выключение функционала исследований По умолчанию — true Подробнее в разделе Сбор скриншотов агентом мониторинга
"monitoring_screenshot_blur"	—	Настройка размытия скриншотов, собираемых агентом мониторинга По умолчанию установлено значение 50 Можно задать значение от 1 до 100. Если размытие не требуется, укажите 0

com.infomaximum.subsystem.bidata.json

Параметр	Переменная окружения	Описание
"temp_table_lifetime"	—	Время жизни временных таблиц, создаваемых в ClickHouse при загрузке CSV-файлов Значение по умолчанию: 2h (2 часа)
"max_transition_count_for_process_map"	—	Карта процесса: максимальное количество переходов для вычислителя Значение по умолчанию: 300
"max_variant_count_for_process_map"	—	Карта процесса: максимальное количество цепочек вариантов Значение по умолчанию: 100
"max_variant_length_for_process_map"	—	Карта процесса: максимальная длина цепочки вариантов Значение по умолчанию: 100
"use_clickhouse_balancer"	—	Карта процесса: использовать балансировщики ClickHouse Значение по умолчанию: false - true — в запросах при использовании кластерного хранилища будут задействоваться обычные таблицы (CREATE TABLE …) - false — в запросах при использовании кластерного хранилища будут задействоваться TEMPORARY-таблицы (CREATE TEMPORARY TABLE …) Для некластерного подключения всегда используются TEMPORARY-таблицы

com.infomaximum.subsystem.dashboard.json

Параметр	Переменная окружения	Описание
"rows_export_limit"	DB_ROWS_LIMIT	Настройка лимита количества строк при выгрузке таблиц в форматах .xlsx и .csv из дашборда По умолчанию — 5000 Допустимо любое значение Подробнее в разделе Экспорт табличного отчета
"temp_table_lifetime"	—	Определяет срока хранения временных таблиц По умолчанию — 1 день Можно указать значения в днях (d) и часах (h)

com.infomaximum.subsystem.automation.json

Параметр	Переменная окружения	Описание
"is_production"	AU_PROD	Определяет, какие блоки будут доступны в скриптах: все или только стабильные версии Если указано false, появляется доступ к экспериментальному функционалу. По умолчанию установлено true
"script_version_lifetime"	—	Срок хранения версий скрипта. Может указываться в секундах (s), минутах (m), часах (h) Если указать значение 0d, будет задан бесконечный срок хранения
"exchange_retry_count"	—	Количество автоматических повторных запросов при возникновении определенных ошибок Exchange 0 — отключает повторы (ошибка приводит к немедленному завершению блока с ошибкой) 1 — одна попытка повтора >1 — соответствующее число повторов с интервалом 1 минута. По умолчанию — 25

Параметр	Переменная окружения	Описание
"exchange_ignore_exceptions"	—	Включение режима игнорирования ошибок при работе с календарями Exchange Если указано true — ошибки при обработке отдельных календарей не прерывают выполнение скрипта; информация об ошибках фиксируется в логах Если указано false — ошибки приводят к прерыванию обработки (значение по умолчанию) Примечание: После изменения параметра требуется перезапуск центрального сервера

com.infomaximum.subsystem.clickhouse.standalone.json

Параметр	Описание
"repeat_sql_query_count"	Позволяет установить количество повторений запросов при возникновении ошибки
"repeat_sql_query_time"	Время между повторными запросами Указывается в миллисекундах
"log_queries"	Отвечает за запись SQL запросов в таблицу system.query_log, которые генерируют дашборды и виджеты в таблицу system.query_log сервера ClickHouse Если выбрано значение false (по умолчанию), то информация не записывается, если true — записывается
"checkout_timeout"	Ограничивает время, в течение которого Proceset будет ожидать подключения к ClickHouse в том случае, если пул доступных соединений к ClickHouse исчерпан Указывается в миллисекундах

com.infomaximum.platform.component.database.json

Параметр	Описание
"periodical_backup_enabled"	Включение/выключение создания периодической перезаписываемой резервной копии встроенной файловой базы данных, которое выполняется в полночь
"backup_path"	Путь к каталогу, в котором будет сохраняться периодическая резервная копия встроенной файловой базы данных относительно рабочего каталога Proceset По умолчанию — "backup". При необходимости можно указать другой каталог Косую черту необходимо экранировать еще одной косой чертой

cluster.json

Параметр	Переменная окружения	Описание
"name"	CL_NAME	Имя текущей ноды в кластере Proceset Подробнее в разделе Установка модуля автоматизации на другой сервер

Параметр	Переменная окружения	Описание
"name"	CL_PORT	Порт текущей ноды для работы в кластере Proceset Подробнее в разделе Установка модуля автоматизации на другой сервер
"nodes"	CL_REMOTE_NODES	Список других нод в кластере Proceset

Переменные окружения

Примечание. Некоторые переменные окружения не связаны с конкретными конфигурационными файлами.

Переменная окружения	Описание
CR_LOG_LEVEL	Уровень технических логов (main.log) Возможные значения: trace, debug, info, warn, error По умолчанию — debug Подробнее в разделе Ротация журнала безопасности
JVM_MAX_MEMORY	Лимит оперативной памяти на java-машину По умолчанию — 4G На Windows значение изменяется через реестр Подробнее в разделах Установка приложения Infomaximum на Linux и Указание доступного для приложения Infomaximum объема оперативной памяти

Редактирование конфигурационных файлов на ОС Linux

Примечание. Не все параметры можно изменить через переменные окружения, так как не все настройки системы вынесены в них.

Если представленных переменных окружения недостаточно, можно использовать переменную SKIP_CONFIGURE. Если указать ее в команде запуска Docker-контейнера, все переменные окружения, кроме JVM_MAX_MEMORY, будут проигнорированы. В этом случае конфигурацию нужно будет редактировать вручную в файлах, расположенных в каталоге /var/lib/infomaximum/config/ внутри контейнера.

Чтобы использовать переменную SKIP_CONFIGURE:

1. Запустите систему Proceset обычным способом. Это нужно для того, чтобы конфигурационные файлы заполнились необходимыми данными.

2. Скопируйте конфигурационные файлы из контейнера на хост. Используйте команду:

```
docker cp a382:/var/lib/infomaximum/config/ /home/user/config
```

Где a382 — это идентификатор контейнера, который можно узнать с помощью команды docker ps, и /home/user/config — это каталог на вашем хосте.

3. Остановите контейнер.

```
docker service rm infomaximum-app
```

4. Откройте и отредактируйте необходимые конфигурационные файлы в указанном каталоге.

5. При следующем запуске укажите, где находятся файлы с настройками с помощью параметра --mount, и добавьте параметр, который запретит системе менять настройки: -e SKIP_CONFIGURE='yes'.

Пример команды:

```
docker service create --name infomaximum-app \
--secret infomaximum_app_https_certificate \
--secret infomaximum_app_https_certificate_password \
--mount type=volume,src=infomaximum-app-data-t,target=/var/lib/infomaximum/data/ \
--mount type=volume,src=infomaximum-app-log-t,target=/var/log/infomaximum/ \
--mount type=bind,source=/home/user/config,target=/var/lib/infomaximum/config/ \
--publish published=443,target=8010,mode=host \
--restart-max-attempts 5 \
--restart-condition "on-failure" \
-e JVM_MAX_MEMORY='4G' \
-e FE_URL="https://proceset.local" \
-e SKIP_CONFIGURE='yes' \
dockerhub.office.infomaximum.com/infomaximum/infomaximum_app:d240710
```

Предупреждение. Поскольку конфигурационные файлы будут находиться на сервере, все переменные, которые использовались при предыдущем запуске службы, сохранятся в этих файлах. Однако новые переменные, указанные при следующем запуске контейнера, могут не сработать. Чтобы изменения применились, укажите новые значения вручную в файлах настроек или запустите службы без параметра SKIP_CONFIGURE и без привязки каталога с настройками. После этого повторите процесс: скопируйте обновленные файлы на сервер, внесите нужные изменения и снова запустите службу со SKIP_CONFIGURE.

Метрики Prometheus

Prometheus — это набор инструментов для мониторинга и оповещения систем с открытым исходным кодом. Система мониторинга собирает информацию о состоянии серверов и систем, а также может получать предупреждения о проблемах.

Объекты мониторинга называются целевыми объектами. Главное отличие от остальных систем мониторинга — метод сбора данных. Prometheus сам берет нужную ему информацию с серверов и устройств, обращаясь к целевым объектам при помощи языка PromQL.

Метрики хранятся в форме временных рядов, где каждая метрика сохраняется с соответствующим временным штампом и сопровождается дополнительными парами «ключ-значение», которые называются метками (labels).

Метрики доступны для получения по пути /metrics с указанием параметра api_key={api_key_value}. У APIключа должна быть привилегия «Метрики Prometheus» с операцией на **Чтение**. Пример окончательного URL может выглядеть так:

```
https://test.infomaximum.com/metrics?api_key=4952343e6e1448a69518f7b2523e340d
```

Собираются только метрики Java Virtual Machine.

Примеры метрик

Ниже приведены примеры метрик, которые собирает система.

jvm_memory_used_bytes — количество используемой памяти в байтах:

```
# HELP jvm_memory_used_bytes Used bytes of a given JVM memory area.
# TYPE jvm_memory_used_bytes gauge
jvm_memory_used_bytes{area="heap"} 1.7151512E8
```

jvm_threads_started_total — количество потоков, которые были запущены в JVM:

```
# HELP jvm_threads_started_total Started thread count of a JVM
# TYPE jvm_threads_started_total counter
jvm_threads_started_total 4842.0
```

jvm_threads_state — количество потоков в определенном состоянии:

```
# HELP jvm_threads_state Current count of threads by state
# TYPE jvm_threads_state gauge
jvm_threads_state{state="BLOCKED"} 0.0
jvm_threads_state{state="NEW"} 0.0
jvm_threads_state{state="RUNNABLE"} 32.0
```

process_cpu_usage_percentage — нагрузка процессора в процентах:

```
# HELP process_cpu_usage_percentage CPU usage in percentage. CPU load between 0 and 100.
# TYPE process_cpu_usage_percentage gauge
process_cpu_usage_percentage 4.633415099482148
```

node_filesystem_size_bytes — количество свободной, общей и занятой памяти на диске в байтах:

```
# HELP node_filesystem_size_bytes Filesystem size in bytes.
# TYPE node_filesystem_size_bytes gauge
node_filesystem_size_bytes{type="Free"} 1.12989249536E11
node_filesystem_size_bytes{type="Total"} 5.11440576512E11
node_filesystem_size_bytes{type="Usage"} 3.98451326976E11
```

node_memory_bytes — количество свободной, общей и занятой оперативной памяти в байтах:

```
# HELP node_memory_bytes Process memory in bytes.
# TYPE node_memory_bytes gauge
node_memory_bytes{type="Free"} 1.7631875072E10
node_memory_bytes{type="Total"} 3.3142767616E10
node_memory_bytes{type="Usage"} 1.5510892544E10
```

http_server_requests_seconds — количество http-запросов к системе Proceset и время выполнения этих запросов в секундах:

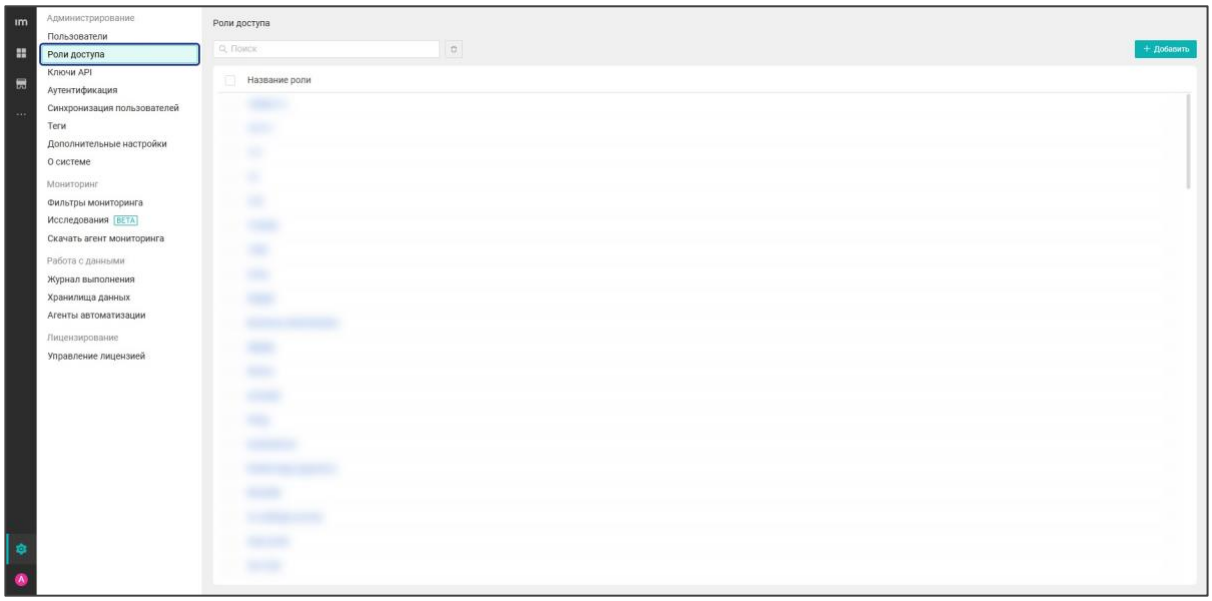
```
# TYPE http_server_requests_seconds histogram
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.005"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.01"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.025"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.05"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.1"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.25"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="0.5"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="1.0"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="2.5"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="5.0"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="10.0"} 2
http_server_requests_seconds_bucket{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304",le="+Inf"} 2
http_server_requests_seconds_count{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304"} 2
http_server_requests_seconds_sum{method="GET",path="/_build/static/css/main.d0c31d4330cd73f2.css",status="304"} 0.0031729999999999996
```

Ошибки при сборе метрик

Код ошибки	Описание
auth_ambiguity	Метрики запрошены с использованием API-ключа при входе в систему Proceset
illegal_apikey_exception	API-ключ содержит менее 8 символов
invalid_credentials	API-ключ не существует в системе
access_denied	У API-ключа недостаточно привилегий

Роли доступа

На странице *Роли доступа* создаются роли доступа к функционалу системы.



Каждой роли доступа можно присвоить свое уникальное имя и настроить привилегии.

Название роли доступа можно поменять в ее профиле во вкладке *Основное*.

Роли **Администратор** и **Администратор ИБ** предустановлены в системе.

При необходимости также можно создать отдельные настраиваемые роли с разными уровнями возможностей.

Примечание.

- Нельзя создавать отдельные настраиваемые привилегии
- Для некоторых привилегий доступны не все операции доступа

Примеры:

- **Бизнес-администратор** — имеет расширенный доступ к пространствам, автоматизации, данным ClickHouse, всем сотрудникам и их настройкам. Однако, в

отличие от администратора и администратора ИБ, он не имеет доступа к глобальным настройкам системы, например ролям доступа, настройкам синхронизаций и аутентификаций и т.д.

- **Аналитик данных** — может создавать пространство, дашборды, работать с данными ClickHouse, автоматизацией. Аналитик данных видит всех пользователей системы, но не может менять для них настройки
- **Бизнес-аналитик** — может создавать дашборды, просматривать данные ClickHouse и дашборды других пользователей
- **Сотрудник (или пользователь)** — обычный пользователь, который может просматривать только те дашборды и сотрудников, к которым ему дали доступ

Информацию о том, какие сотрудники имеют конкретную роль доступа, можно получить через дашборд по ролям и доступам.

Ролевая модель

Ролевая модель предусматривает следующие разграничения для пользователей:

- К функциональным возможностям системы
- К данным по пользователям
- К аналитическим отчетам системы
- К инструменту GraphQL (API)

Каждому пользователю системы могут назначаться роли доступа, доступные пользователи и пространства данных. Возможные действия пользователя в системе будут определяться набором этих привилегий.

Для каждой роли доступа устанавливается набор разрешенных операций доступа по отношению к группе конкретных объектов. Пользователю может быть назначено более одной роли доступа. В результате пользователь получит права каждой из назначенных ролей. Для назначения доступны следующие операции доступа:

- **R** — чтение
- **W** — изменение
- **C** — создание
- **D** — удаление
- **E** — выполнение

Для Ключей API невозможно назначение ролей доступа. Для каждого Ключа API привилегии доступа назначаются отдельно. Они могут быть:

- **R** — чтение
- **W** — изменение
- **C** — создание
- **D** — удаление
- **E** — выполнение

Система контроля доступа является закрытой. Изначально объект не доступен никому. Наличие отдельных привилегий доступа зависит от подключенных модулей. Привилегии указаны в таблице:

Модуль Платформа

Назначение	Действия
Привилегия «Роли доступа»	
Отвечает за настройку ролей доступа в системе В веб-интерфейсе настройка параметров производится: Настройки/Администрирование/Роли доступа Назначение привилегии невозможно для Ключей API	Операция R дает доступ просматривать: 1. Раздел Роли доступа: список созданных ролей и профиль роли доступа (название и привилегии) 2. Раздел Пользователи, профиль пользователя, вкладка Доступ: Предоставляет доступ к списку ролей доступа, созданных в системе (если пользователю также доступна привилегия Доступы пользователей W – изменение) Операция W дает доступ: 1. На изменение названия и настройку роли в профиле роли доступа Операция C дает доступ создавать: 1. Роли доступа Операция D дает доступ удалять: 1. Роли доступа
Привилегия «Ключи API»	
Отвечает за настройку Ключей API, с помощью которых осуществляются внешние и внутренние интеграции В веб-интерфейсе настройка параметров производится: Настройки/Администрирование/Ключи API	Операция R дает доступ просматривать: 1. Список ключей API и их профили (название, значение ключа, режим аутентификации, доступ к пространствам, привилегии) Операция W дает доступ: 1. На изменение названия ключа и доступа к пространствам, к настройке привилегии ключа во вкладке Профиль ключа Операция C дает доступ: 1. На создание ключа API и загрузку сертификата, если выбран безопасный тип ключа Операция D дает доступ: 1. На удаление ключа
Привилегия «Аутентификация»	
Отвечает за настройку параметров аутентификации в системе В веб-интерфейсе просмотр и настройка параметров производится: Настройки/Администрирование/Аутентификация	Операция R дает доступ просматривать: 1. Раздел Аутентификация: список аутентификаций (тип «Встроенная» и др.) и профиль аутентификации (тип «Встроенная») 2. Профиль аутентификации: название, тип, сложный пароль (Минимальная длина пароля, если Сложный пароль: Вкл.), срок действия пароля, ограничение попыток входа 3. Раздел Пользователи: аутентификации в списке пользователей (если включена привилегия Доступы пользователей R — чтение) 4. Раздел Пользователи, профиль пользователя, вкладка Доступ: параметр Аутентификация (если включена привилегия Доступы пользователей R — чтение), список аутентификаций, созданных в системе (если включена привилегия Доступы пользователей W — изменение)

Назначение	Действия
	Операция W дает доступ изменять: 1. В профиле аутентификации типа «Встроенная»: название, сложный пароль (Вкл/Выкл), минимальную длину пароля, срок действия пароля, количество допустимых попыток входа 2. В профиле аутентификации типа «Kerberos»: название, тип аутентификации, ключ (.keytab), адрес центра выдачи ключей, сопоставление атрибутов по полю Операция C дает доступ создавать: 1. Аутентификацию Операция D дает доступ удалять: 1. Аутентификацию
Привилегия «Общие настройки»	
Отвечает за настройку параметра «Формат инициалов» и за настройку прав доступа к параметрам агента автоматизации В веб-интерфейсе параметры представлены на страницах Настройки/Администрирование/Дополнительные настройки и Настройки/Работа с данными/Агенты автоматизации	Операция R дает доступ просматривать: 1. Раздел Дополнительные настройки (формат инициалов) 2. Страницу О системе 3. Раздел Агенты автоматизации и состояния агентов Операция W дает доступ: 1. На изменение в разделе Дополнительные настройки (формат инициалов) 2. На подключение/отключение агентов автоматизации в разделе Агенты автоматизации
Привилегия «Сервер исходящей почты»	
Отвечает за настройку почтового сервера системы Настройка параметров производится через веб-интерфейс: Настройки/Подключения/Сервер исходящей почты	Операция R дает доступ просматривать: 1. Во вкладке Подключение: адрес электронной почты, адрес сервера, порт соединения, шифрование, имя пользователя 2. Во вкладке Контакты: адрес электронной почты и номер телефона Операция W дает доступ изменять: 1. Во вкладке Подключение: адрес электронной почты, адрес сервера, порт соединения, шифрование (Выключено/SSL/TLS), имя пользователя, пароль 2. Во вкладке Контакты: адрес электронной почты и номер телефона. Операция E дает доступ: 1. На проверку соединения
Привилегия «Пользователи и отделы»	
Отвечает за настройку прав доступа к параметрам Пользователей, Основное в профиле пользователя, Профиль отдела, Источники сбора активности В списке пользователей отображаются только те пользователи и их параметры, к которым имеет доступ пользователь. В веб-интерфейсе настройка параметров	Операция R дает доступ просматривать: 1. Раздел Пользователи: список пользователей и отделов, профиль пользователя (имя, фамилию, отчество, язык системы, отдел, табельный номер, электронную почту, номер телефона, кастомные поля), профиль отдела (название отдела и расположение) 2. Информацию о пользователях с группами безопасности Active Directory через GraphQL-запрос Операция W дает доступ:

Назначение	Действия
производится: Настройки/Администрирование/Пользователи	1. Объединять пользователей и отделы в разделе Пользователи 2. Изменять профиль пользователя (имя, фамилию, отчество, язык системы, отдел, табельный номер, электронную почту, номер телефона, данные в кастомных полях), профиль отдела (название отдела и расположение) Операция C дает доступ создавать: 1. Пользователей и отделы С привилегиями Доступы пользователей — W и «Аутентификация» — W пользователь также может задавать аутентификацию и активировать настройку «Доступ ко всем пользователям» в профиле пользователя Операция D дает доступ удалять: 1. Пользователей и отделы
Привилегия «Доступы пользователей»	
Отвечает за возможность настройки доступов пользователей в системе	Операция R дает доступ просматривать: 1. В профиле пользователя во вкладке Доступ назначенные роли доступа, аутентификации (если включена привилегия «Аутентификация» R – чтение), настройку «Доступ ко всем пользователям» (Вкл/Выкл), доступных пользователей (если параметр «Доступ ко всем пользователям» в положении Выкл) 2. Системные переменные в редакторе формул и способах ввода данных для запуска скрипта Операция W дает доступ изменять: 1. Роли доступа (дает возможность удалять и назначать роли доступа, если пользователю также доступна привилегия Роли доступа R – чтение), логин, аутентификации (дает возможность удалить или добавить аутентификации, если включена привилегия «Аутентификация» W — изменение), доступных пользователей, задать или сбросить пароль (если пользователю также доступна привилегия «Аутентификация» W – изменение) Операция E дает доступ: 1. На отправку приглашений
Привилегия «Инструмент GraphQL»	
Отвечает за настройку доступа к инструменту GraphQL в системе Назначение привилегии невозможно для Ключей API	Операция E дает доступ: 1. На использование инструмента GraphQL
Привилегия «Настройка тегов»	
Отвечает за создание, настройку и удаление тегов в системе В веб-интерфейсе настройка параметров производится: Настройки/Администрирование/Теги	Операция R дает доступ просматривать: 1. Список тегов и профиль тега (название тега и цвет тега) Операция W дает доступ изменять: 1. Профиль тега (название тега и цвет тега)

Назначение	Действия
	Операция C дает доступ создавать: 1. Теги Операция D дает доступ удалять: 1. Теги
Привилегия «Сервисный режим»	
Отвечает за настройку доступа в систему при включенном состоянии системы «Сервисный режим»	Операция R дает доступ: 1. Авторизоваться в систему при включенном «Сервисном режиме»
Привилегия «Поля пользователя»	
Отвечает за настройку раздела Поля пользователя	Операция R дает доступ просматривать: 1. Список полей в разделе Поля пользователя Операция W дает доступ редактировать: 1. Поля в разделе Поля пользователя Операция C дает доступ создавать: 1. Кастомные поля в разделе Поля пользователя Операция D дает доступ удалять: 1. Кастомные поля в разделе Поля пользователя
Привилегия «Метрики Prometheus»	
Отвечает за сбор и передачу метрик Prometheus Привилегию невозможно применить для ролей доступа, можно назначить исключительно для Ключей API	Операция R дает доступ просматривать: 1. Собираемые метрики
Привилегия «Управление лицензиями»	
Отвечает за доступ к разделу Управление лицензиями	Операция R дает доступ просматривать: 1. Список ключей и общее количество лицензий во вкладке Обзор Операция C дает доступ активировать: 1. Лицензионные ключи

Модуль ClickHouse

Назначение	Действия
Привилегия «Хранилища»	
Отвечает за настройку подключений в разделе настроек Хранилища данных Привилегия недоступна в SAAS-версии	Операция R дает доступ просматривать: 1. Список созданных подключений и их настройки Операция W дает доступ изменять: 1. Список созданных подключений и их настройки, доступы пользователей на создание пространств на выбранном подключении Операция C дает доступ создавать: 1. Подключение ClickHouse в разделе Хранилища данных Операция D дает доступ удалять: 1. Подключение ClickHouse в разделе Хранилища данных

Модуль Мониторинга

Назначение	Действия
Привилегия «Параметры мониторинга»	
Отвечает за доступ к параметрам мониторинга	Операция R дает доступ с помощью GraphQL-запросов получать: 1. Список собираемых агентом мониторинга данных, актуальную версию установленного агента мониторинга, список источников активности, статус параметра «Сбор активности» для конкретного сотрудника, уровень логирования агента мониторинга Операция W дает доступ: 1. Повторно обрабатывать данные активности с возможностью удаления файла в случае ошибки 2. Привязывать источник активности к конкретному сотруднику
Привилегия «Фильтры мониторинга»	
Отвечает за настройку фильтров мониторинга (белый и черный списки)	Операция R дает доступ просматривать вкладки: 1. Настройка фильтра, белый список, черный список Операция W дает доступ изменять: 1. Режим фильтра (выключен, белый список, черный список), активность в белом списке, активность в черном списке Операция C дает доступ добавлять: 1. Активность в белом списке и активность в черном списке Операция D дает доступ удалять: 1. Активность из белого и черного списков
Привилегия «Экспорт/импорт активности пользователей»	
Позволяет с помощью API экспортировать и импортировать активность пользователей	Операция E дает доступ выполнять: 1. GraphQL-запросы для экспорта и импорта активности пользователей
Привилегия «Скачать агент мониторинга»	
Отвечает за доступ к скачиванию дистрибутива агента мониторинга в системе	Операция E дает доступ: 1. На скачивание агента
Привилегия «Агент мониторинга»	
Отвечает за соединение агента мониторинга и сервера в системе Привилегию невозможно применить для ролей доступа, можно назначить исключительно для Ключей API	Операция R дает доступ получать: 1. Актуальную версию агента мониторинга, дистрибутив агента мониторинга для обновления, настройки для агента мониторинга, ФИО пользователя по его ID, список поддерживаемых протоколов агентских данных, значение параметра «Мониторинг» у пользователей, список фильтров мониторинга, настройки исследований, параметр размытия скриншотов в конфигурационном файле Операция W дает доступ: 1. Загружать данные с активностью от агента мониторинга, загружать дамп-файлы агента мониторинга 2. Создавать пользователей

Назначение	Действия
Привилегия «Исследования»	
Отвечает за доступ к исследованиям	Операция R дает доступ: 1. Просматривать исследования, настройки исследований, значение размытия скриншотов в конфигурационном файле 2. Загружать скриншоты по их UUID через URL-адрес Операция W дает доступ: 1. Изменять настройки исследований *Добавление сотрудников возможно, если у пользователя включена привилегия «Пользователи и отделы» с доступом R или W 2. Запускать/останавливать исследования Операция C дает доступ добавлять: 1. Исследования Операция D дает доступ удалять: 1. Исследования

Модуль Active Directory

Назначение	Действия
Привилегия «Синхронизация с User Directory»	
Отвечает за управление настройками Active Directory и других интеграций в разделе Синхронизация пользователей	Операция R дает доступ просматривать: 1. Список синхронизаций и их профили, которые включают настройки: название интеграции, автоматическая синхронизация, синхронизировать по полю в системе, синхронизировать по атрибуту в AD, убрать доступ при выключении в AD, убрать доступ при отсутствии источника, дать доступ при включении в AD, обезличивать при удалении в AD, время синхронизации, список контроллеров и их параметры (протокол подключения, сертификат, адрес, имя пользователя, пароль, описание), список доменных объектов, параметры для первичного сопоставления атрибутов: атрибут в AD и поле в системе 2. Список условий приоритизации Операция W дает доступ изменять: 1. Название интеграции, протокол подключения, сертификаты, адрес контроллера домена, имя пользователя, пароль, убрать доступ при выключении в AD, дать доступ при включении в AD, обезличивать при удалении в AD, доменные объекты, синхронизируемые атрибуты, синхронизацию в Active Directory профиля пользователя, первичное сопоставление атрибута (поле в системе и атрибут в AD) 2. Приоритетность источников во вкладке Приоритизация Операция C дает доступ создавать: 1. Интеграции, контроллеры домена, доменные объекты, синхронизируемые атрибуты, условия приоритизации Операция D дает доступ удалять: 1. Интеграции, сертификаты, контроллеры домена, доменные объекты, синхронизируемые атрибуты, условия приоритизации

Назначение	Действия
	Операция E дает доступ выполнять: 1. Тестирование подключения и синхронизацию

Модуль Бизнес-Аналитика

Назначение	Действия
Привилегия «Пространство»	
Отвечает за настройку прав доступа к пространствам	Операция R дает доступ: 1. Просматривать пространства, дашборды, таблицы в Модели данных, связи, показатели пространств и вкладку Основное 2. Просматривать все папки и вложенные в них папки и пространства 3. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью) 4. Просматривать последние пространства 5. Просматривать, добавлять, переименовывать и удалять закладки 6. Добавлять пространство в закладки Операция W дает доступ: 1. Просматривать дашборды, скрипты, Модель данных, таблицы, процессы и подключения *Работа с подключениями возможна, если у пользователя есть привилегия «Подключения» с операциями R или W 2. Переименовывать пространства, добавлять и изменять описание 3. Создавать, удалять и редактировать дашборды, скрипты, Модель данных, процессы, подключения и показатели пространств 4. Импортировать и экспортировать таблицы из Модели данных 5. Назначать доступы в пространствах *Назначение доступов возможно, если у пользователя включена привилегия «Пользователи и отделы» с доступом R или W 6. Просматривать вкладку Пакеты 7. Просматривать пакеты в пространствах, добавлять и удалять их. Добавление возможно как из Маркетплейса, так и при импорте конфигурации вручную 8. Работать с Журналом выполнения 9. Создавать папки в корне раздела и подпапках, переименовывать папки, перемещать папки и пространства между папками с помощью мыши *Предоставление операции W (изменение) автоматически включает для пользователя операцию R (чтение) 10. Работать с аналитическим ассистентом ProcesetAI Операция C дает доступ: 1. Просматривать все папки и вложенные в них папки 2. Создавать пространства (требуется привилегия «Хранилища») *После создания пространства пользователь, создавший его, автоматически получает локальный доступ на его изменение (операция W) Операция D дает доступ: 1. Удалять пространства в корзину и восстанавливать их с помощью перетаскивания мышью *Требуется локальный доступ на изменение или операции W привилегии Пространство 2. Окончательно удалять пространства из корзины *Требуется локальный доступ на изменение или

Назначение	Действия
	операции W привилегии Пространство 3. Удалять папки, перемещая мышью их содержимое в корзину *Требуется операция W привилегии Пространство Операция E дает доступ: 1. Дублировать пространства *Требуется локальный доступ на изменение или операция W привилегии Пространство, а также привилегия «Хранилища» 2. Экспортировать пространства *Требуется одно из условий: локальный доступ на изменение или операция W привилегии Пространство
Привилегия «Экспорт компонентов»	
Позволяет экспортировать данные виджетов в дашборде	Операция E дает доступ: 1. Выгружать данные виджетов Таблица и Сводная таблица в формате .csv
Привилегия «Маркетплейс»	
Позволяет загружать пакеты в Маркетплейс Привилегия доступна только для standalone-версии	Операция C дает доступ: 1. Загружать пакеты Операция D дает доступ: 1. Удалять пакеты
Привилегия «Приложения»	
Позволяет устанавливать приложения из Маркетплейса	Операция C дает доступ: 1. Устанавливать приложения Операция D дает доступ: 1. Удалять приложения

Локальные доступы (назначаются на определенное пространство)

Доступ	Что назначает доступ внутри пространства
Просмотр	1. Просматривать пространства, дашборды, таблицы в Модели данных, связи, показатели пространств и вкладку Основное 2. Просматривать папки, к пространствам которых есть доступ 3. Просматривать, добавлять, переименовывать и удалять закладки 4. Добавлять пространство в закладки 5. Просматривать последние пространства 6. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью)
Изменение	1. Просматривать дашборды, скрипты, Модель данных, таблицы, процессы и подключения *Работа с подключениями возможна, если у пользователя есть привилегия «Подключения» с операциями R или W 2. Переименовывать пространства, добавлять и изменять описание 3. Импортировать и экспортировать таблицы из Модели данных 4. Назначать доступы в пространствах *Назначение доступов возможно, если у пользователя включена привилегия «Пользователи и отделы» с операциями R или W 5. Просматривать пакеты в пространствах, добавлять и удалять их 6. Работать с Журналом выполнения 7. Удалять пространства в корзину и из нее, а также восстанавливать их с помощью перетаскивания мышью

Доступ	Что назначает доступ внутри пространства
	*Требуется привилегия «Пространство» с операцией D 8. Дублировать пространство *Требуется привилегия «Пространство» с операцией E и привилегия «Хранилища» 9. Экспортировать пространство *(Требуется привилегия «Пространство» с операцией E) 10. Перемещать пространство между папками с помощью мыши 11. Просматривать папки, к пространствам которых есть доступ 12. Просматривать, добавлять, переименовывать и удалять закладки 13. Добавлять пространство в закладки 14. Просматривать последние пространства 15. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью)
Создание дашбордов	1. Создавать дашборды в пространстве (доступ на изменение дашборда назначается автоматически при создании) 2. Просматривать созданные дашборды и Модель данных 3. Просматривать папки, к пространствам которых есть доступ 4. Просматривать, добавлять, переименовывать и удалять закладки 5. Добавлять пространство в закладки 6. Просматривать последние пространства 7. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью)

Локальный доступ к дашбордам

Доступ	Возможности
Чтение	1. Просматривать доступные дашборды в пространстве 2. Просматривать пространства, к дашбордам которых есть доступ 3. Просматривать последние пространства 4. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью) 5. Просматривать папки, к пространствам которых есть доступ 6. Просматривать, добавлять, переименовывать и удалять закладки 7. Добавлять пространство в закладки
Изменение	1. Просматривать доступные дашборды в пространстве, Модель данных и вкладку Основное 2. Просматривать пространства, к дашбордам которых есть доступ 3. Редактировать дашборды 4. Просматривать последние пространства 5. Добавлять пространства в избранное и удалять их из него (поддерживается перетаскивание мышью) 6. Просматривать папки, к пространствам которых есть доступ 7. Просматривать, добавлять, переименовывать и удалять закладки 8. Добавлять пространство в закладки

Таблица доступов

Действия	Операции
Просмотр пространства и папок	R
Добавление в избранное и удаление из него	R
Просмотр последних пространств	R
Просмотр, добавление, переименование и удаление закладок	R
Добавление пространства в закладки	R
Редактирование пространства	R, W

Действия	Операции
Импорт и экспорт таблиц из Модели данных	R, W
Назначение доступов к пространству	R, W и привилегия «Пользователи и отделы»
Создание папок	R, W
Перемещение папок	R, W
Перемещение пространств между папками	R, W
Переименование папок	R, W
Создание пространства	C и привилегия «Хранилища»
Перемещение пространства в корзину	R, W, D или D и локальный доступ к пространству на изменение
Перемещение папок в корзину	R, W, D
Удаление пространств из корзины	R, W, D или D и локальный доступ к пространству на изменение
Восстановление пространств из корзины	R, W, D или D и локальный доступ к пространству на изменение
Дублирование пространства	R, W, E и привилегия «Хранилища» или E, локальный доступ к пространству на изменение и привилегия «Хранилища»
Экспорт пространства	R, W, E или E и локальный доступ к пространству на изменение
Импорт пространства	Импорт при создании пространства: C и привилегия «Хранилища» Импорт в существующее пространство: R, W или локальный доступ к пространству на изменение

Модуль Автоматизация

Назначение	Действия
Привилегия «Подключения»	
Отвечает за настройку прав доступа к подключениям	Операция R дает доступ просматривать: 1. Список доступных подключений и их параметры (название, источник, хост, порт, имя базы данных, имя пользователя, пароль (кнопка Изменить пароль. По умолчанию вводимые символы скрываются точками, при нажатии на символ глаза открывается вводимый текст), переключатель с SSL) Операция W дает доступ: 1. Изменять у подключения: название, хост, порт, имя базы данных (выбор таблиц ограничивается указанной базой данных), имя пользователя, пароль (по умолчанию вводимые символы скрываются точками, при нажатии на символ глаза открывается вводимый текст), переключатель с SSL (при активации): корневой сертификат (загрузка сертификата) Операция C дает доступ: 1. Создавать подключения Операция D дает доступ: 1. Удалять подключения

Назначение	Действия
Привилегия «Системные таблицы»	
Отвечает за доступ к чтению и редактированию блока пакета «Системные таблицы»	Операция C дает доступ: 1. Добавлять в скрипт блок пакета «Системные таблицы» и импортировать скрипт с блоком «Системные таблицы»

Предустановленные роли доступа

По умолчанию после установки системы в модулях есть две сквозные роли доступа:

- «Администратор» (А)
- «Администратор ИБ» (АИБ)

Администратор имеет расширенный доступ ко всем настройкам системы. Он может:

- Создавать, изменять и удалять:
  - Пользователей, отделы
  - Роли доступа
  - Ключи API
  - Аутентификации
  - Пространства
  - Синхронизации с Active Directory и др.
- Изменять общие настройки системы, параметры мониторинга
- Использовать инструмент GraphQL

Также администратор имеет доступ к сервисному режиму.

Роль **Администратора ИБ** предназначена для персонала, основной обязанностью которого является обеспечение информационной безопасности.

Администратор ИБ выполняет следующие функции:

- Контроль за предоставлением пользователям доступа к системе
- Контроль прав доступа пользователей к функционалу системы
- Контроль прав доступа пользователей к данным системы
- Выявление событий несанкционированного доступа к системе
- Направление требований администратору системы по изменению настроек системы, направленных на реализацию мер информационной безопасности

Также Администратор ИБ имеет доступ к инструменту GraphQL.

Матрица прав доступа по привилегиям для предустановленных ролей приведена в таблице ниже.

Название привилегии	А	АИБ
Роли доступа	RWCD	R
Ключи API	RWCD	R
Аутентификация	RWCD	R
Общие настройки	RW	R
Сервер исходящей почты	RWE	R
Пользователи и отделы	RWCD	R
Доступы пользователей	RWE	R
Инструмент GraphQL	E	E
Настройка тегов	RWCD	R
Поля пользователя	RWCD	R
Хранилища (недоступна в SAAS-версии)	RWCD	R

Название привилегии	А	АИБ
Параметры мониторинга	RW	R
Фильтры мониторинга	RWCD	R
Экспорт/импорт активности пользователей	E	E
Скачать агент мониторинга	E	E
Агент мониторинга	—	–
Синхронизация User Directory	RWCDE	R
Сервисный режим	R	R
Пространство	RWCDE	–
Подключения	RWCD	–
Системные таблицы	C	–
Маркетплейс	CD	CD
Приложения	CD	CD
Экспорт компонентов	E	–
Метрики Prometheus	–	–
Исследования	RWCD	R
Управление лицензиями	RC	—

Для роли **Администратор** обязательно наличие доступа ко всем пользователям.

Для **Администратора ИБ** по умолчанию установлен выборочный доступ, но вы можете дать его ко всем пользователям в профиле сотрудника во вкладке Доступ.

Выборочный доступ к пользователям подразумевает под собой наличие у пользователя доступа к определенному списку пользователей. Список пользователей регламентируется заказчиком, доступ предоставляет администратор системы. Доступ ко всем пользователям подразумевает под собой доступ ко всем пользователям, которые существуют в системе.

Дополнительная информация по ролям доступа

- Первому пользователю системы назначается роль доступа **Администратор**
- При установке нового модуля никому не дается никакой роли доступа в нем
- Имя роли доступа должно быть уникально. Задать одинаковое значение наименования роли доступа невозможно
- Удалить роль доступа **Администратор** невозможно
- Перед тем, как назначить пользователей администраторами, в их профилях необходимо включить Доступ ко всем пользователям

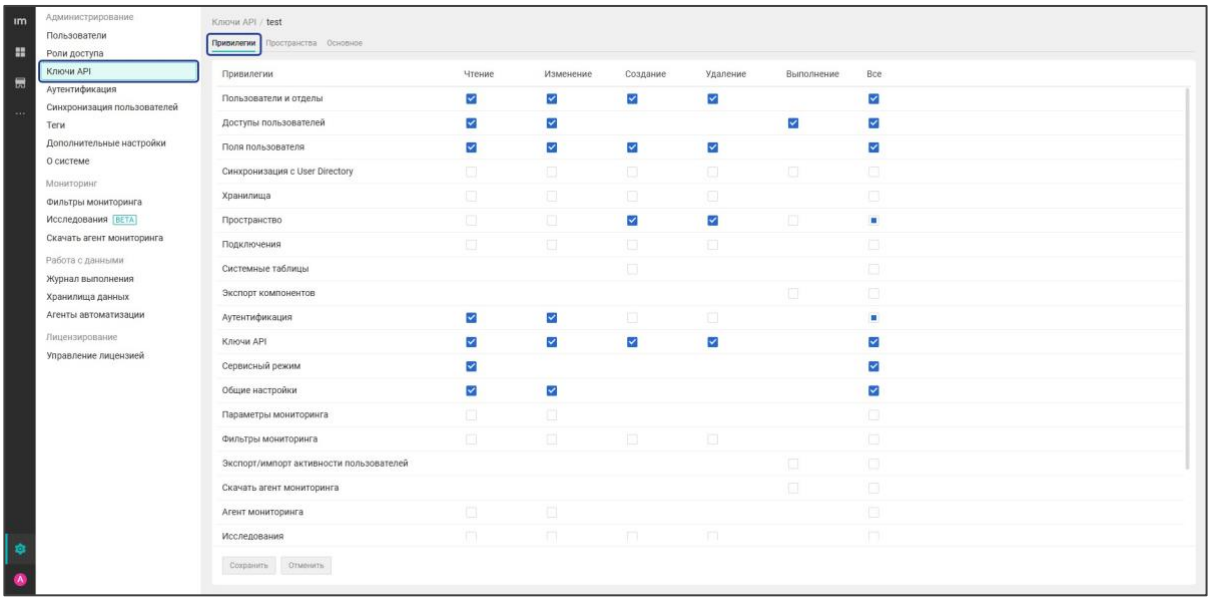
Если в системе остался один пользователь, у которого назначена роль с операцией доступа **W**, то в таком случае:

- Невозможно у этой роли доступа изменить/выключить данную привилегию
- У данного пользователя в профиле в *Основное* обязательно должны быть заполнены поля **Электронная почта**, **Пароль** (для возможности восстановления пароля и авторизации)
- У данного пользователя в профиле нельзя удалить роль доступа с привилегией *Роль доступа*, если она является последней из всех, в которой включена эта привилегия

Авторизация в системе доступна как с ролью доступа, так и без нее. Настройка авторизации осуществляется через конфигурационный файл.

Права доступа для Ключей API

Для каждого ключа API, по аналогии с пользователями системы, можно назначить права доступа к привилегиям и пространствам. Настройка прав доступа ключа API происходит в его профиле, в разделе Настройки / Ключи API веб-интерфейса системы.

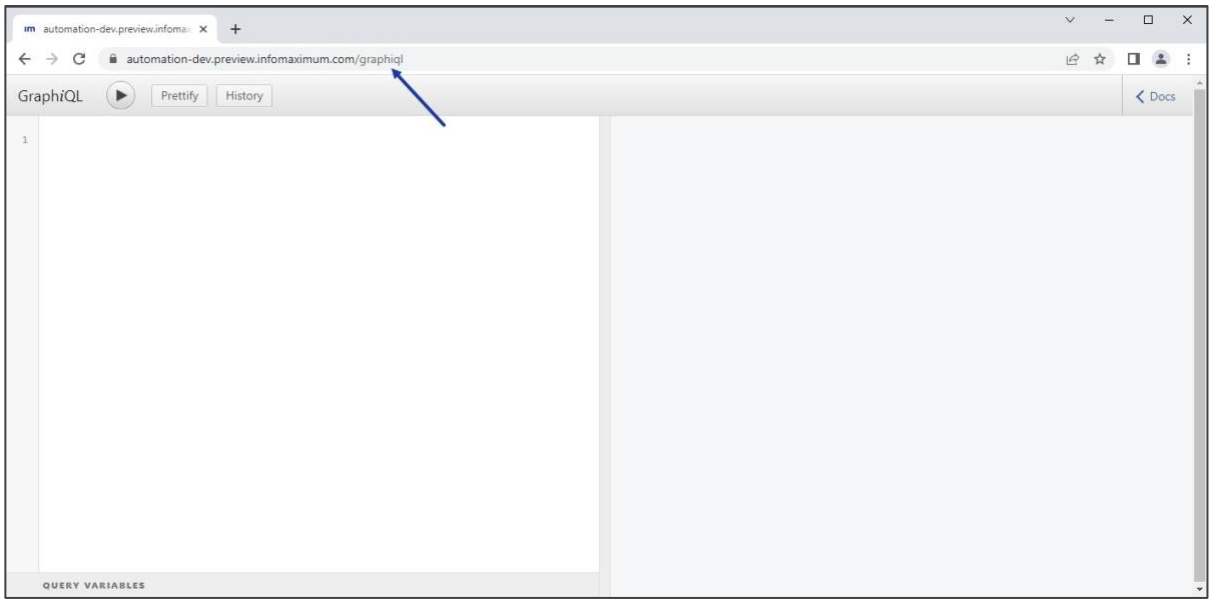


Работа с GraphQL

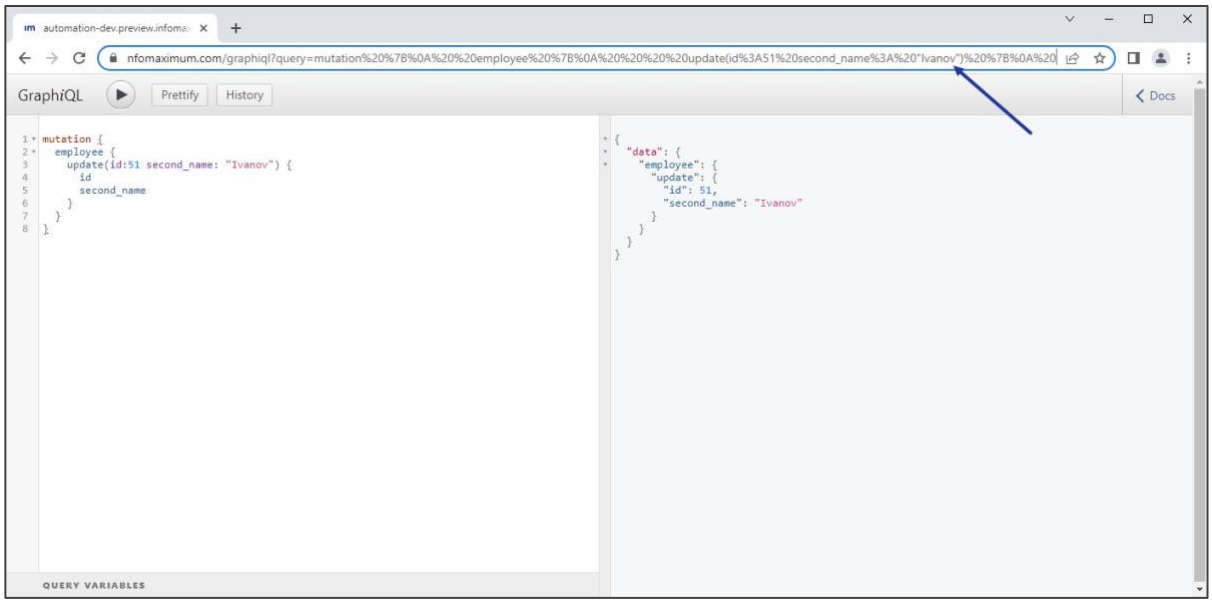
Описание GraphQL

В качестве API система Proceset использует технологию GraphQL. **GraphQL** — это язык запросов для API с открытым исходным кодом, который обеспечивает более эффективную и гибкую альтернативу классическому REST. В то время как типичные API REST требуют загрузки с нескольких URL-адресов, API GraphQL позволяет получить необходимые данные в одном запросе.

Для создания и выполнения GraphQL-запросов можно использовать встроенную браузерную IDE (интегрированную среду разработки) GraphiQL. Чтобы получить к ней доступ, перейдите на страницу системы и в адресной строке введите *адрес_системы/graphiql*.



Когда вы редактируете запрос во встроенной IDE, URL-адрес также обновляется. При этом сохраняются пробелы, комментарии и неверные запросы. Вы можете пересылать URL-ссылки другим коллегам.



GraphQL-запросы являются интерактивными. Это означает, что вы можете изменить запрос и увидеть новый результат.

GraphQL API поддерживает автогенерацию документации. Документация всегда в актуальном состоянии. Для ее просмотра откройте вкладку *Docs* (находится в правом верхнем углу) на странице встроенной IDE. В этом разделе представлены все запросы, которые можно выполнить в Proceset.

GraphQL-запросы в системе разделены по типу на три группы:

- query
- mutation
- subscription

Выберите нужную группу и найдите интересующий запрос.

В описании каждого запроса представлены список полей и их типы данных.

С дополнительной информацией по работе с GraphQL можно ознакомиться на официальном сайте.

Описание способов аутентификации и способов работы с GraphiQL

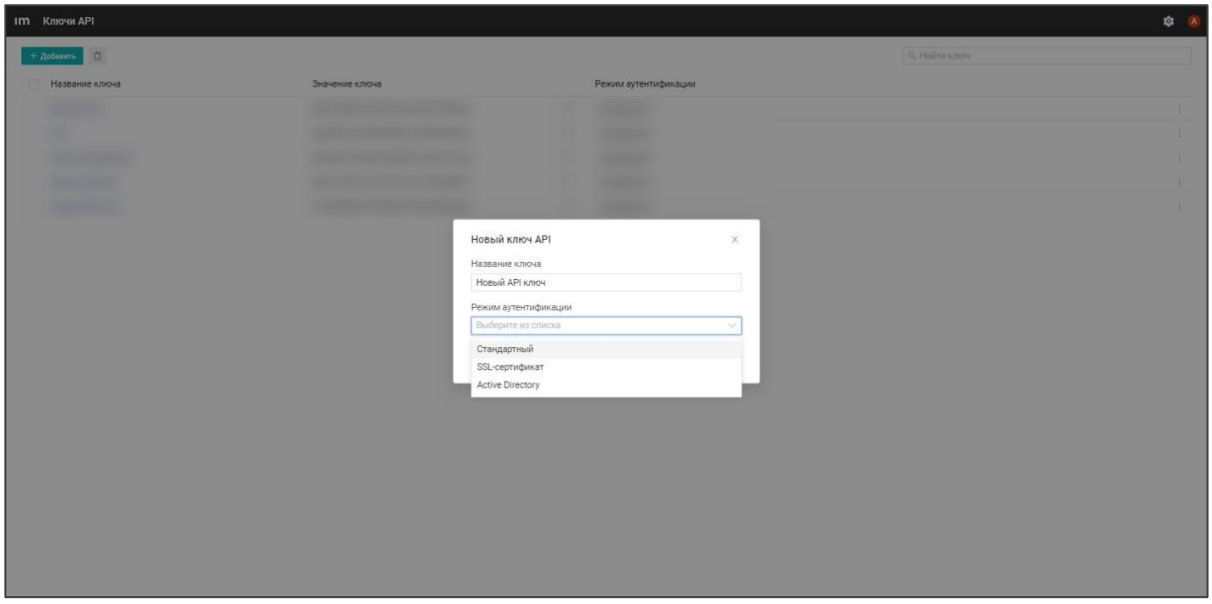
Для более удобного формирования и выполнения запросов в систему включена встроенная IDE GraphiQL. Чтобы ее использовать:

1. Авторизуйтесь в Proceset.

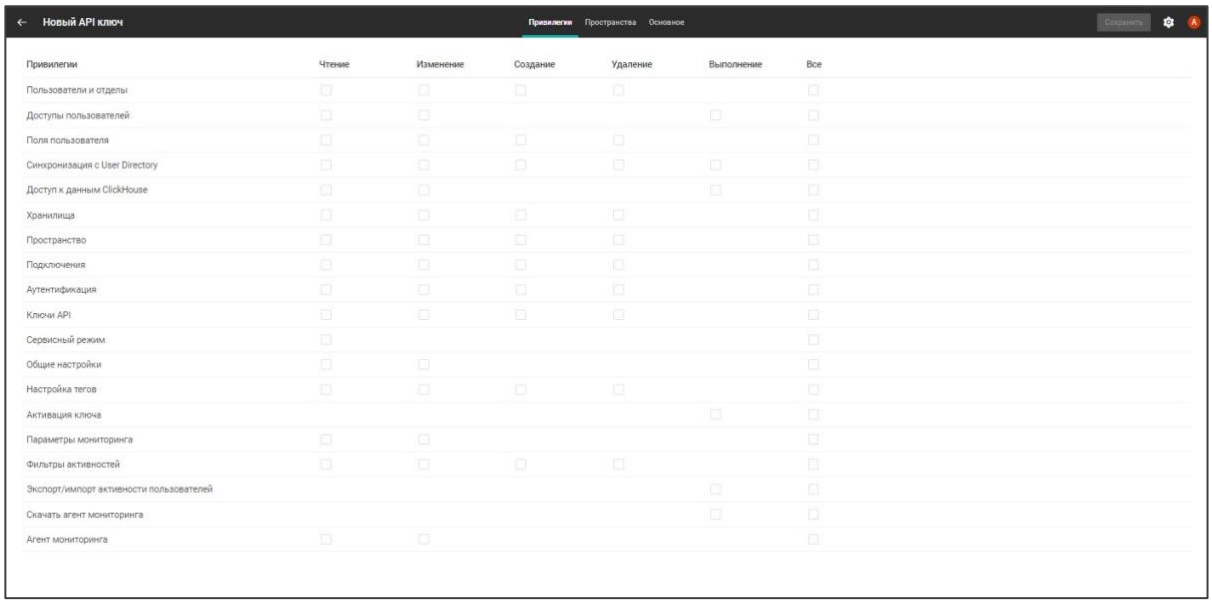
2. Перейдите по адресу: *адрес_системы/graphiql*.

Для выполнения запроса из внешней системы необходимо подготовить ключ API для аутентификации в разделе «Настройки» / «Ключи API». Для подключения внешних систем возможны два режима аутентификации через ключи API:

- стандартный ключ API представляет собой последовательность символов, которую должна будет предъявлять внешняя система для аутентификации
- SSL-сертификат. Внешняя система для аутентификации сначала предоставляет API-ключ, затем происходит взаимная аутентификация систем на основе SSL-сертификатов



Назначьте привилегии созданному ключу API. Подробнее о настройке привилегий ключам API см. Права доступа для ключей API.



Запрос от имени ключа API

Чтобы GraphQL-запрос был от имени ключа API, необходимо в адресной строке ввести ключ. Пример: https://automation-dev.preview.infomaximum.com/graphql?api_key=ключ

Примеры запросов GraphQL

Далее приведены некоторые GraphQL-запросы, которые могут быть использованы в системе.

Получение рабочего пространства

Запрос:

```
{
  workspace {
    workspace (id: 273) {
      id
      name
    }
  }
}
```

Ответ:

```
{
  "data": {
    "workspace": {
      "workspace": {
      "id": 273,
      "name": "test"
      }
    }
  }
}
```

Поле	Определение
workspace	Рабочее пространство
id	Идентификатор рабочего пространства
name	Название рабочего пространства

Получение сотрудников с конкретной ролью доступа

Запрос:

```
mutation {
   employee {
      create (
         login: "test"
         first_name: "Ivan"
         second_name: "Ivanov"
         language: RUSSIAN
         phone_numbers: "12345678901"
      ){
         id
         login
         first_name
         second_name
         language
         phone_numbers
      }
   }
}
```

Ответ:

```
{
   "data": {
      "employee": {
         "create" {
            "id": 244,
            "login": "test",
            "first_name": "Ivan",
            "second_name": "Smirnov",
            "language": "RUSSIAN",
            "phone_numbers": [
               "12345678901"
            ]
         }
      }
   }
}
```

Поле	Определение
access_role	Роль доступа
employees	Сотрудники
items	Элементы
id	Идентификатор сотрудника
first_name	Имя сотрудника
second_name	Фамилия сотрудника

Получение GUID скрипта и ID пространства

Для запуска скрипта требуются его GUID и ID рабочего пространства. Чтобы получить их, выполните запрос:

```
{
  automation {
    script {
      script_general(id: <id_скрипта>) {
        id
        guid
        workspace {
          id
        }
      }
    }
  }
}
```

Ответ:

```
{
  "data": {
    "automation": {
      "script": {
        "script_general": {
          "id": 20001,
          "guid": "1234567a-1234-567b-234c-123456789abc",
          "workspace": {
            "id": 5001
          }
        }
      }
    }
  }
}
```

Поле	Определение
automation	Модуль «Автоматизация»
script	Профиль скрипта
script_general	Основная информация
id	Идентификатор скрипта
guid	GUID скрипта
workspace_id	Идентификатор рабочего пространства

Запуск скрипта

Запрос:

```
mutation {
   automation {
      script {
         execute_by_guid (
            guid:"1234567a-1234-567b-234c-123456789abc"
            workspace_id: 5001
```

```
        ){
            guid
        }
    }
  }
}
```

Ответ:

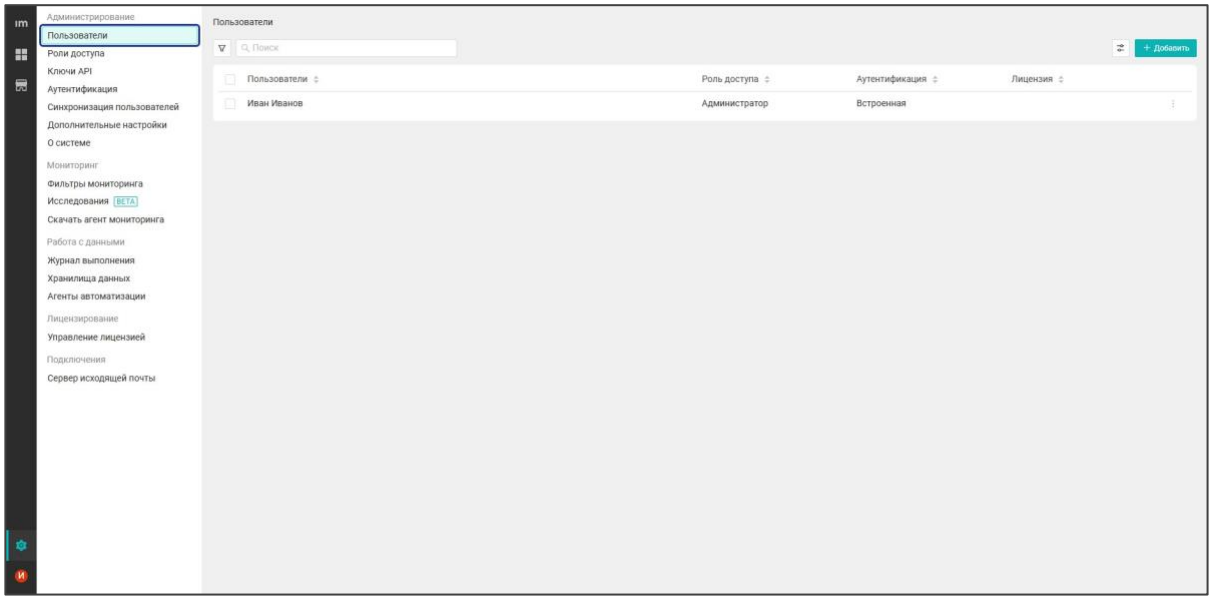
```
{
  "data": {
    "automation": {
      "script": {
        "execute_by_guid": {
          "guid": "1234567a-1234-567b-234c-123456789abc"
        }
      }
    }
  }
}
```

Поле	Определение
automation	Модуль «Автоматизация»
script	Выполнение скрипта
execute_by_guid	Запуск скрипта по GUID
guid	GUID скрипта
workspace_id	Идентификатор рабочего пространства

Управление пользователями системы

Просмотр списка пользователей

Просмотр списка пользователей происходит через веб-интерфейс системы. Список находится в разделе *Администрирование* на странице *Пользователи*, в нем отображаются профили сотрудников, к которым есть доступ.



Профили пользователей можно создавать вручную через веб-интерфейс. Также они могут создаваться автоматически на основе данных, поступивших от агентов мониторинга и/или интеграции с Active Directory или с помощью GraphQL. Основы работы с GraphQL описаны в Работа с GraphQL.

Просмотр списка групп

Просмотр списка групп возможен двумя способами:

- Через веб-интерфейс системы. Список групп находится в разделе Администрирование на странице Пользователи
- Через GraphQL-запрос следующего формата:

```
{
department{
  departments{
    id
    name
   }
  }
}
```

Просмотр изменения групп

Для просмотра изменения групп используйте GraphQL-запрос:

```
{
  department{
    department (id:1) {
```

```
    id
    name
   }
  }
}
```

где id:1 – номер конкретной группы.

Для сравнения необходимо выполнить два запроса или более: первый запрос выполняется в начале анализируемого периода, второй запрос – в конце анализируемого периода. Отличия в ответах указывают на изменение параметров.

Управление ролями доступа системы

Просмотр списка ролей доступа

Существует два варианта просмотра списка РД:

- Через веб-интерфейс системы. Список ролей находится в разделе Настройки/Роли Доступа
- Отправка GraphQL-запроса следующего формата:

```
query{
 access_role{
  access_role_list{
    items{
     element{
      __typename
      id
      name
     }
    }
  }
 }
}
```

Системные письма на почту пользователей

Письмо-приглашение в систему

Чтобы отправить письмо-приглашение в систему, необходим доступ к **профилю сотрудников** и должен быть настроен сервер исходящей почты.

Для отправки письма-приглашения:

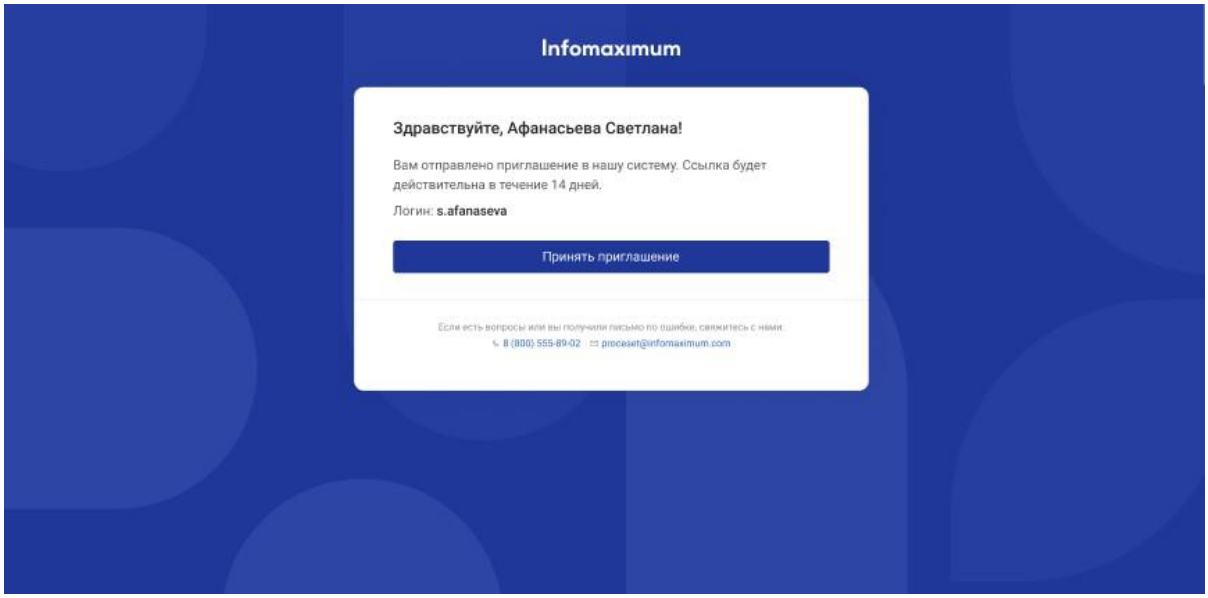
1. Откройте профиль нужного сотрудника.

2. Перейдите во вкладку Доступ.

3. Нажмите кнопку **Отправить приглашение**.

4. На почтовый адрес, указанный в профиле сотрудника (вкладка *Основное*), придет сообщение с приглашением.

Чтобы принять приглашение, нажмите **Принять приглашение**. Вы перейдёте на страницу входа в систему. Ссылка в письме на приглашение активна в течение **14 дней**.



Важно.

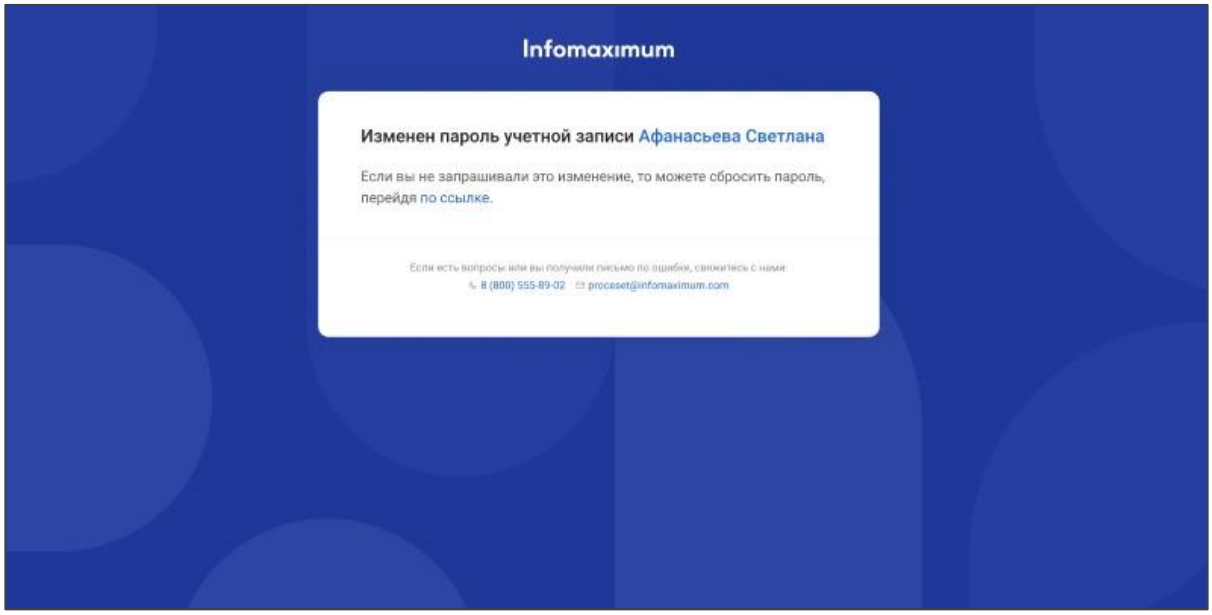
- Функция **Отправить приглашение** доступна, если:
  - указан логин сотрудника
  - указана электронная почта у сотрудника
  - задана роль доступа

Если в системе у вас появилось сообщение «Приглашение отправлено», но сотрудник не получил письмо, проверьте настройку сервера исходящей почты.

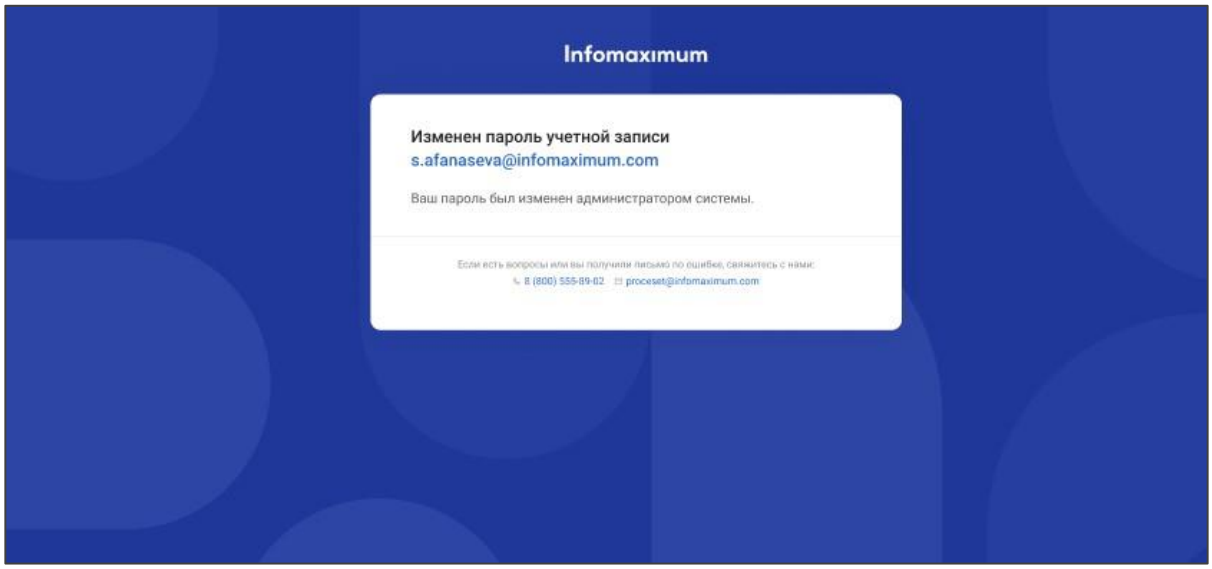
Письмо об изменении пароля

Письмо об изменении пароля придет в двух случаях:

- если вы самостоятельно изменили пароль



- если администратор системы изменил ваш пароль



Пользователь не получает письмо об изменении пароля в следующих случаях:

- если администратор системы задает пароль в профиле сотрудника первый раз, то есть ранее у пользователя пароля не было
- если пользователь самостоятельно восстанавливает пароль через почту
- когда было отправлено письмо-приглашение в систему и пользователь задал пароль.

Письмо для восстановления пароля

Если вы забыли пароль:

1. При входе в систему нажмите **Забыли пароль?**.

2. Введите ваш логин.

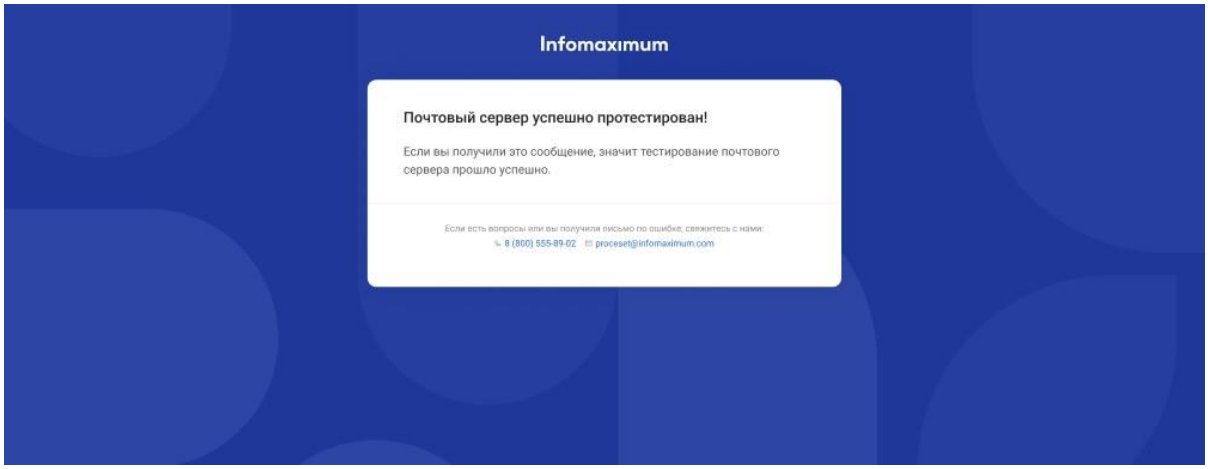
3. На адрес электронной почты, указанный в вашем профиле, будет отправлена инструкция по восстановлению пароля.

Ссылка на восстановление пароля действует в течении **24 часов**. Когда ссылка становится недействительной, появляется ошибка:

«Ссылка устарела, повторите процедуру восстановления пароля еще раз».

Письмо об успешном тестировании сервера исходящей почты

После сохранения параметров, заданных в настройках сервера исходящей почты, происходит тестирование соединения. Если работа сервера настроена правильно, то на указанную почту придёт письмо.

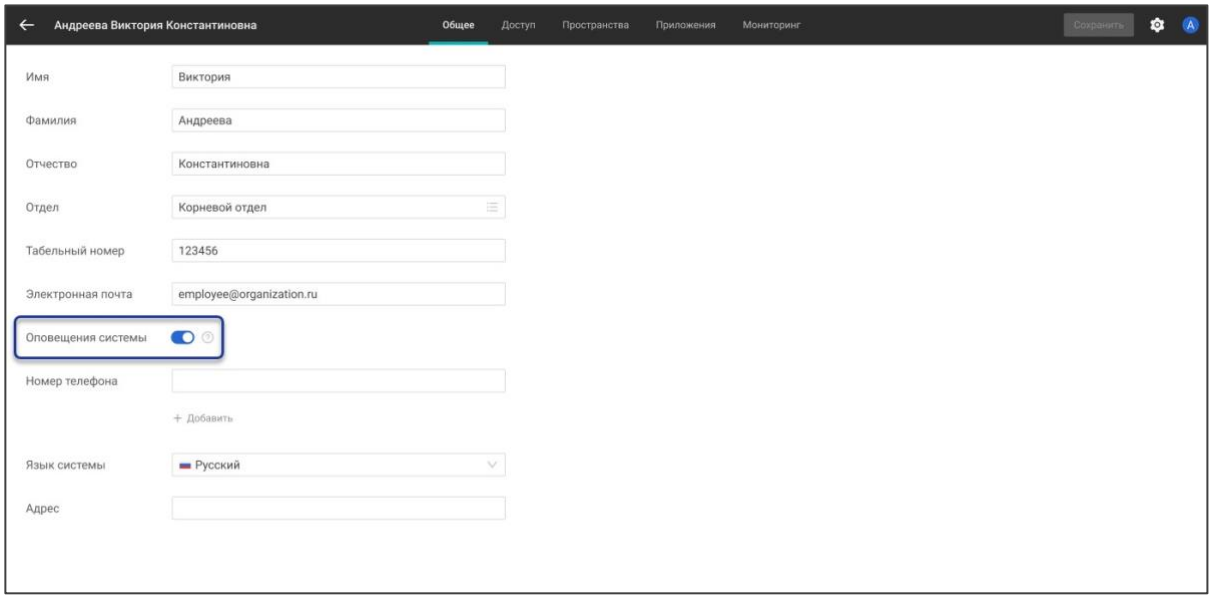


Письмо о диагностических событиях

Диагностические события, создаваемые системой, направляются на почтовый сервер для администратора системы. Примеры событий:

- ошибка в создании бэкапа
- ошибка в последней синхронизации системы с AD
- ошибка отправки писем сервером исходящей почты
- выявление повышенной нагрузки на ЦПУ, где установлен сервер приложений
- выявление увеличивающейся очереди активности
- сбой по синхронизации активности между базами данных
- заканчивается оперативная память, где установлен сервер приложений

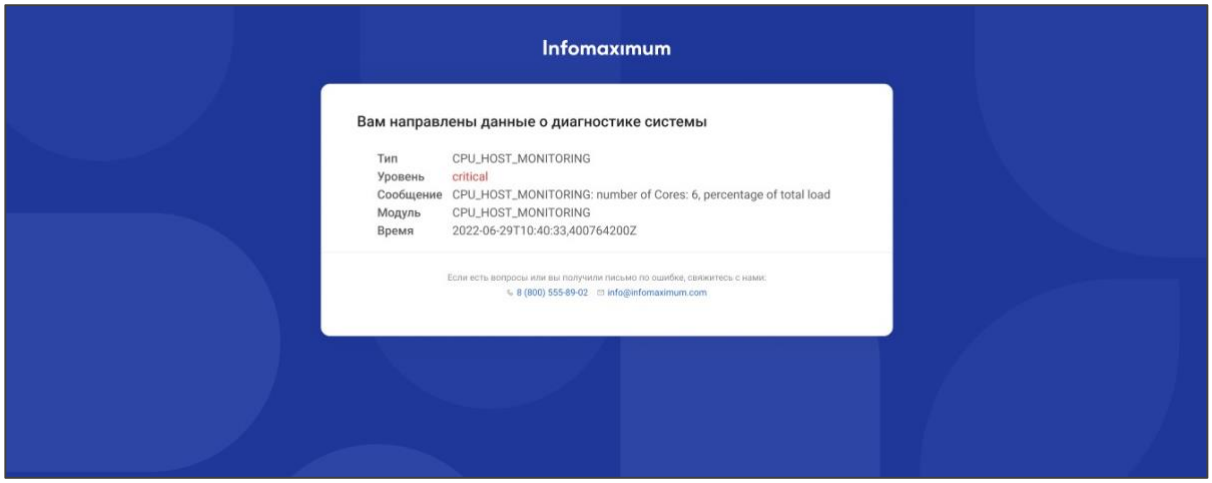
За согласие или отказ от получения уведомлений отвечает переключатель «Оповещения системы» в профиле администратора системы. По умолчанию настройка включена.



Периодичность отправки писем можно изменить в config-файле *com.infomaximum.subsystem.core.json*. По умолчанию отправляется 1 письмо в 1 час.

В письме отображаются:

- тип
- уровень
- сообщение
- модуль
- время



Встраивание дашбордов Proceset

Дашборды Proceset можно встраивать во внешние системы с помощью HTML-элемента <iframe>.

Встраивание через iframe используется, когда нужно показать дашборд внутри другой веб-системы, например, CRM или корпоративного портала, и открыть его сразу с заданными параметрами без перехода в интерфейс Proceset.

Важно. Для просмотра дашбордов требуется активная авторизация пользователя в Proceset. При встраивании через iframe содержимое отображается только при наличии действующей пользовательской сессии. Анонимный доступ к дашбордам не поддерживается.

Включение поддержки iframe

Чтобы дашборды Proceset можно было отображать во внешних системах, включите поддержку <iframe> в конфигурации системы.

Важно. Перед включением iframe согласуйте это решение с отделом информационной безопасности, так как оно может создавать дополнительные риски безопасности:

- Cross-Site Scripting (XSS)
- Clickjacking
- Фишинг через iframe
- Cross-Frame Scripting (XFS)
- Внедрение вредоносного контента
- Загрузка небезопасных ресурсов
- Проблемы с контролем доверия пользователя
- Утечка данных и нарушение приватности

Добавьте параметр support_iframe в файл конфигурации com.infomaximum.subsystem.frontend.json:

```
{
  "support_iframe": true
}
```

После изменения конфигурации необходимо перезапустить сервер.

Встраивание дашборда на страницу

При встраивании дашборда можно передать параметры в URL, которые влияют на его отображение, например:

```
   <iframe

src="https://site.com/im/report/5673/publish?frame=true&filters=%5B%7B%22type%22%3A%22
formula%22%2C%22format%22%3A%22STRING%22%7D%5D"
   width="100%"
   height="600"
   frameborder="0"
   allowfullscreen
   ></iframe>
```

Атрибут	Описание
src	URL дашборда Можно скопировать из адресной строки браузера при его открытии URL должен содержать параметр frame=true
width	Ширина iframe
height	Высота iframe
frameborder="0"	Отключение рамки вокруг встроенного дашборда
allowfullscreen	Включение полноэкранного режима

Параметры отображения дашборда

При встраивании дашборда можно передать параметры в URL, которые влияют на его отображение, например:

```
https://site.com/im/report/123/publish?frame=true&filters=[{"type":"formula","format":"STRING"}]
```

Параметр	Описание
frame	Управляет отображением навигационной панели и других элементов управления при встраивании дашборда Если значение параметра true, панель навигации и дополнительные элементы управления скрываются, если false — отображаются
filters	Позволяет передать готовые фильтры через адресную строку Значение должно быть в формате JSON и корректно закодировано для использования в URL (например, с помощью encodeURIComponent)

Заметка. Дополнительные параметры (например, переменные или входящие значения для BI) поддерживаются в зависимости от конфигурации дашборда.

Требования и ограничения на стороне внешнего сайта

Если на сайте, куда вы встраиваете iframe, настроена политика безопасности Content Security Policy (CSP), необходимо разрешить загрузку iframe с домена, на котором размещен Proceset. В противном случае браузер заблокирует встраивание, даже если на стороне Proceset все настроено корректно.

Пример настройки:

```
Content-Security-Policy: frame-src https://site.com;
```

Где site.com — адрес установки Proceset.

Также рекомендуется использовать атрибут sandbox в <iframe> для ограничения действий встроенного содержимого.

Пример:

```
<iframe

src="https://site.com/im/report/5673/publish?frame=true&filters=%5B%7B%22type%22%3A%22formula%22%2C%22format%22%3A%22STRING%22%7D%5D"
  width="100%"
  height="600"
```

```
  frameborder="0"
  allowfullscreen
  sandbox="allow-same-origin allow-scripts allow-forms"
></iframe>
```

Различия системы Proceset на операционных системах Linux и Windows

Аутентификация

В системе Proceset на Linux невозможно использовать стандартную аутентификацию Windows. Это обусловлено тем, что возможности стандартной аутентификации Windows обеспечивает операционная система Windows, на которой устанавливается Система. Аутентификация через AD на Linux возможна только с помощью механизма Kerberos.

Ключи API

В системе Proceset на Linux невозможно использовать ключи API с аутентификацией через Active Directory. Это связано с тем, что механизм «Встроенной проверки подлинности Windows», который используется для проверки подлинности ключей, доступен только на ОС Windows. Такие ключи API подходят для агентов мониторинга на Windows.

Горячие клавиши в модуле автоматизации

Перетаскивание холста зажатием клавиши «Space» на ОС Linux доступно только при использовании TrackPoint.

Редактирование конфигурационных файлов

Примечание. Редактирование конфигурационных файлов недоступно в SAAS-версии.

Если операционная система Linux используется в качестве сервера, настройки, которые можно изменять в конфигурационных файлах, задаются через переменные окружения при старте Docker-контейнера.

Примечание. Не все настройки могут быть изменены таким образом, поскольку не все они вынесены в переменные окружения.

Параметры, которые вы можете изменить на ОС Linux, перечислены ниже.

Файл com.infomaximum.subsystem.frontend.json

Параметр	Описание	Примечание
FE_URL	Адрес сервера Proceset	Необходимо указать полностью, с протоколом и портом
FE_SERVICE_MODE	Сервисный режим	По умолчанию выключен. Чтобы включить, укажите true (доступно с версии 2308)
FE_SERVICE_MODE_MESSAGE	Сообщение при включенном сервисном режиме	Отображается, если включен сервисный режим (доступно с версии 2308)

Файл com.infomaximum.subsystem.monitoring.json

Параметр	Описание	Примечание
MN_EMAIL_HASHING	Хеширование данных из полей Кому и Копия	По умолчанию default. Для отключения укажите none

Параметр	Описание	Примечание
MN_STORAGE_GUID	Параметр позволяет передать GUID хранилища для синхронизации данных активности при обновлении или перезапуске системы	По умолчанию поле storage_guid не заполнено, для синхронизации данных автоматически берется GUID предыдущего хранилища из встроенной файловой базы знаний

Файл com.infomaximum.subsystem.dashboard.json

Параметр	Описание	Примечание
DB_ROWS_LIMIT	Настройка лимита количества строк при выгрузке таблиц из дашборда	По умолчанию 5000, допустимо любое значение

Файл com.infomaximum.subsystem.automation.json

Параметр	Описание	Примечание
AU_PROD	Параметр стабильной версии	По умолчанию true. Если указано false, появляется доступ к экспериментальному функционалу (доступно с версии 2310)

Файл com.infomaximum.subsystem.core.json

Параметр	Описание	Примечание
CR_NOTIFICATION_MESSAGE	Сообщение для информационного окна	Включается автоматически, когда оставляется сообщение (доступно с версии 2311)

Мониторинг

Мониторинг в Proceset — это сбор и хранение данных о пользовательской активности на рабочих станциях. Эти данные позволяют организациям изучать активность сотрудников, видеть, как используется профильное программное обеспечение, и применять результаты для оптимизации процессов и управления рабочим временем.

Как работает мониторинг

На компьютеры сотрудников устанавливается агент мониторинга. Он регистрирует действия пользователей (работу с окнами и приложениями, ввод текста, использование клавиатуры и мыши) и передает собранные данные на сервер Proceset. После этого данные становятся доступны для анализа.

Агент мониторинга

Агент мониторинга запаковывает и передает ZIP-архив с активностью пользователя на сервер приложения Proceset.

Примечание. В папке /databases/monitoring_raw_data данные хранятся временно и будут удалены при истечении срока 14 дней. Срок можно изменить в конфигурационном файле com.infomaximum.subsystem.monitoring.json.

Архивы передаются на сервер и попадают в папку /databases/monitoring_raw_data во встроенную файловую базу данных. В ней они читаются фоновым процессом, помещаются в очередь на обработку и проверяются на корректность. Архивы, не соответствующие требованиям (например, с ошибками в manifest.json, несуществующим пользователем и т. д.), не попадают в ClickHouse и не отражаются в таблицах monitoring_activity и monitoring_agent_inspector_log. Такие архивы помещаются в специальную очередь corrupted и остаются на сервере.

После проверки корректные архивы перемещаются в отдельные базы данных по годам — постоянное хранилище во встроенной файловой базе данных. Из постоянного хранилища данные синхронизируются с СУБД ClickHouse.

Процесс синхронизации запускается по расписанию, заданному параметром rdb_ch_synchronization в конфигурационном файле com.infomaximum.subsystem.monitoring.json. В ходе синхронизации:

- Данные активности пользователей загружаются в таблицу main.monitoring_activity
- Технические логи агентов мониторинга загружаются в таблицу main.monitoring_agent_inspector_log

Описание структуры таблиц monitoring_activity и monitoring_agent_inspector_log из базы данных main представлено на странице Описание структуры хранения данных в ClickHouse.

Имя базы данных main можно изменить:

- В конфигурационном файле com.infomaximum.subsystem.monitoring.json с помощью параметра monitoring_database_name
- На ОС Linux — с помощью переменной окружения MN_DB_NAME

Требования к имени базы данных:

- Длина не более 255 символов
- Состоит только из латинских букв в верхнем или нижнем регистре, цифр и символов подчеркивания _
- Начинается с буквы

Взаимодействие с сервером приложения

Взаимодействие осуществляется по протоколу HTTPS, порт: 8010. Агент мониторинга передает файлы активности пользователей, а также запрашивает настройки. Взаимодействие автоматическое.

Взаимодействие с сервером Active Directory

Взаимодействие осуществляется по LDAPS. Цель обмена: аутентификация агента мониторинга на АРМ пользователя. Взаимодействие автоматическое.

Как использовать данные мониторинга

Собранные агентом данные можно:

- Анализировать с помощью готовых отчетов, разработанных компанией «Инфомаксимум»:
  - Отчет Task Mining показывает, как выполняются рабочие задачи за компьютером, помогает найти повторяющиеся действия, операции с наибольшими трудозатратами и процессы, которые можно упростить или автоматизировать
  - Отчет Анализ рабочего времени позволяет отслеживать активность сотрудников в течение дня, выявлять переработки и недоработки, анализировать использование корпоративного ПО и контролировать дисциплину
- Визуализировать в собственных дашбордах
- Запрашивать напрямую в скриптах автоматизации, используя переменную activity_table для доступа к таблице main.monitoring_activity независимо от ее физического расположения

Сбор данных агентом мониторинга

Важно.

- Агент мониторинга Proceset Agent использует системные API операционных систем Windows (Microsoft UI Automation и др.) и Linux (AT-SPI и др., если применимо).
- Качество, полнота и корректность сбора аналитических данных, а также стабильность работы агента мониторинга целиком зависят от:
    - Технологической совместимости клиентских приложений с используемыми API и стандартами доступности
    - Особенностей архитектуры и реализации сторонних приложений, включая приложения, использующие проприетарные или устаревшие технологии (legacy-системы)
    - Настроек безопасности и политик корпоративной инфраструктуры, блокировки сторонними средствами защиты, ограничений операционной системы и/или антивирусного программного обеспечения
    - Корректности разметки и экспонирования свойств интерфейса со стороны мониторируемых приложений
    - Правильности активации и поддержки соответствующих технологий доступности, например, Java Access Bridge для Java-приложений
- Компания «Инфомаксимум» принимает все разумные меры для максимальной совместимости с прикладным программным обеспечением, однако не несет ответственности за те ограничения, которые предопределены архитектурой сторонних решений, изменениями системных API, либо политиками безопасности инфраструктуры клиента.

Агент мониторинга

Агент мониторинга — приложение, которое устанавливается на компьютеры сотрудников и регистрирует действия пользователей (работу с окнами и приложениями, ввод текста, использование клавиатуры и мыши).

Агент мониторинга используется как источник данных для анализа пользовательской активности на рабочих станциях. Он играет ключевую роль в построении системы Task Mining, что позволяет точно фиксировать рабочие действия сотрудников, переключения между программами, взаимодействие с интерфейсами и другими элементами ОС. Собранные данные используются для выявления повторяющихся задач, построения моделей бизнес-процессов и анализа загруженности сотрудников.

Примечание. Агент мониторинга не влияет на производительность компьютера и не сохраняет конфиденциальные данные, такие как пароли.

Основные возможности агента мониторинга:

- Фиксирует рабочую активность сотрудников в приложениях, браузерах и других программах
- Помогает определить уровень вовлеченности пользователя, периоды активности и простоя
- Собирает данные, необходимые для построения моделей процессов и выявления повторяющихся задач

В зависимости от уровня детализации, агент может работать в режиме базового или расширенного мониторинга, а также поддерживает сбор данных из Java-приложений и HID-устройств.

Агент мониторинга собирает активность пользователя, которая состоит из событий. *Событие* — это пользовательское действие (нажатие кнопки, переключение вкладки, заполнение поля и т. п.) или программное действие (например, появление диалогового окна, изменение заголовка у окна, изменение активного окна и т. п.). Все события относятся к программе, окно которой активно в текущий момент. Если событие не приводит к каким-то последствиям, например, клики по неактивному элементу управления, такие события не записываются в активность.

Каждое событие содержит в себе информацию о времени совершения действия, названии программы, ее версии, пути к исполняемому файлу и, в зависимости от типа мониторинга, дополнительную информацию. Подробное описание — в разделе Описание структуры хранения данных в ClickHouse.

Примечание. Для тестирования сбора активности может использоваться portable-версия агента мониторинга — *Консольный агент*. Эта версия агента запускается без установки в систему, собирает активность в рамках текущей пользовательской сессии и не передает архивы на сервер. Процесс сбора активности отображается непосредственно в окне терминала.

Базовый мониторинг

При базовом мониторинге агент записывает сведения об открытых окнах приложений и переключениях между ними:

- В случае использования браузера — заголовок активной вкладки и URL-адрес страницы
- В случае использования десктопных программ — заголовок окна
- В случае 1С 8.3 возможен сбор заголовка активной вкладки внутри программы
- В случае MS Excel — заголовок активного листа

К базовому мониторингу относятся события оконной иерархии и HID-активность.

При локальном (без отправки на сервер) хранении активности в течение месяца объем занимаемого дискового пространства на одного пользователя составляет 25 МБ.

Расширенный мониторинг

Расширенный мониторинг собирает данные форм (parameters), если они есть, и события. Parameters — это значения редактируемых полей (с типом Edit и ComboBox). Их поиск происходит при фокусе и при потере фокуса редактируемым полем. После того как parameters-элементы найдены, они будут добавлены во все события в окне.

Агент мониторинга не собирает и не сохраняет пароли пользователей. Поля, в которых пользователю необходимо ввести пароль, не фиксируются агентом мониторинга — сохраняется только факт редактирования поля *Пароль*.

Мониторинг Java-приложений

Для расширенного мониторинга доступны Java-приложения, поддерживающие технологию Java Access Bridge. Это позволяет агенту мониторинга расширить список программ, активность в которых необходимо отслеживать. Примерами таких программ могут быть Oracle, SweetHome 3D и другие.

Для фиксации данных о работе пользователя в Java-приложениях требуется использование модуля Javамониторинга. Java-мониторинг обеспечивает более детализированную информацию о работе Javaприложений, что помогает в их анализе, оптимизации и способствует более эффективному управлению. С этим модулем в приложениях, работающих с технологией Java Access Bridge, будет фиксироваться вся расширенная информация, без него — только информация о заголовках окон.

Сбор и передача данных пользовательской активности

Примечание. Если по каким-то причинам агенту мониторинга не удается определить URL текущей страницы в браузере, активность на этой странице не будет зафиксирована.

Для корректного учета активности по каждому пользователю дополнительно агент собирает:

- Имя и логин учетной записи пользователя, ее домен или рабочую группу
- Имя компьютера, его домен или рабочую группу
- Часовой пояс пользовательской сессии

Сервер Proceset получает от агента мониторинга данные об активности сотрудников и сохраняет их в промежуточную базу данных, затем данные синхронизируются с таблицей monitoring_activity в СУБД ClickHouse. Структура таблицы monitoring_activity, включая информацию о колонках, их типах данных и содержимом подробно описана в разделе Описание структуры хранения данных в ClickHouse.

Виды событий

Примеры событий оконной иерархии:

- WindowOpen — всегда, когда открывается что-то новое (окно, вкладка, файл, url, даже если id окна не менялся)
- DocumentOpen — когда открывается новый документ
- DocumentSave — когда сохраняется документ
- SheetOpen — открывается или переключается лист в Excel
- WindowSwitch — всегда, когда происходит переключение между старыми вкладками или окнами
- WindowUpdate — всегда, когда меняется заголовок вкладки или окна

События, которые не записываются

- Действия, которые ничего не меняют: клики по окну, прокрутка, движение мыши по элементам выпадающего списка
- Смена фокуса при нажатии стрелок
- Клик по редактируемому полю

- Переключение на рабочий стол и действия на нем
- Переключение на панель задач и действия в ней
- Переключение на меню **Пуск** и действия в нем
- Переключение на панель поиска и действия в ней

Особенности фиксации активности

Объекты UI Automation, связанные с неактивной вкладкой браузера, постоянно остаются в памяти агента.

HID-активность

Агент мониторинга дополнительно фиксирует активность от HID-устройств, таких как мышь и клавиатура. С помощью HID-активности можно определить промежутки активности и неактивности пользователя за ПК.

HID-активность от действий пользователя определяется как аппаратное событие — *hardware*. Роботизированное или программное событие определяется как *injected*.

HID-активность фиксируется при следующих действиях:

- Клик мыши: программный (**InjectedMouseClick**) и аппаратный (**HardwareMouseClick**)
- Прокрутка мышью: программная (**InjectedMouseScroll**) и аппаратная (**HardwareMouseScroll**)
- Нажатие кнопок клавиатуры: программное (**InjectedKeyboardEvent**) и аппаратное (**HardwareKeyboardEvent**)

У каждого события используется свой тип, представленный в виде числового кода. С подробной информацией о типах событий вы можете ознакомиться на странице Информация о логировании.

Данные для HID-события берутся из предыдущего события, но с заменой на соответствующий тип HIDактивности (**HardwareMouseClick**, **HardwareKeyboardEvent** и т. п.). HID-активность фиксируется только для полей, в которых возможна длительная работа, например, поля редактирования.

HID-событие фиксируется относительно последнего зафиксированного события, когда интервал между ним и предыдущим событием равен 1 минуте или больше. В то же время, от HID-события до события пользовательского действия может быть меньше 1 минуты.

Если пользователь работает так, что каждое нажатие с интервалом меньше 1 минуты приводит к событию, например, переключению окон, HID-события вовсе могут быть не зафиксированы.

Алгоритм работы

Сохраняется время последнего HID-события, а также время последнего записанного в активность HIDсобытия. Когда происходит новое событие:

- Если время события больше или равно времени последнего события + 1 минута, то в активность добавляются последнее HID-событие и новое
- Если время события больше или равно времени последнего записанного в активность + 1 минута, то в активность добавляется новое событие

В остальных случаях запоминается время.

Пример 1:

Пользователь печатает в текстовом документе и последнее HID-событие зафиксировалось в 10:00:00. Он печатал до 10:00:35, после чего отошел от рабочего места на 5 минут и в 10:05:20 снова кликнул мышью по текстовому документу и начал печатать. В этом случае агент зафиксирует HID-события в 10:00:00, 10:00:35 и 10:05:20.

Анализируя время событий можно сделать вывод что пользователь не работал за ПК с 10:00:35 по 10:05:20, так как разница во времени между 10:00:00 и 10:00:35 меньше 1 минуты.

Пример 2:

Пользователь работает в браузере и последнее HID-событие зафиксировалось в 15:00:30. Он совершил клик в браузере в 15:00:55, сделал паузу и кликнул по уже другой программе в 15:01:20. Агент зафиксирует только одно HID-событие в 15:01:20, так как интервал между 15:00:30 и 15:01:20 меньше минуты и нет необходимости фиксировать в качестве HID-события 15:00:55.

Информация о логировании

Логи агента мониторинга содержат сведения о его работе и действиях пользователей. Они используются для диагностики сбоев, проверки корректности работы агента и анализа активности сотрудников.

Агент передает данные, которые собираются ММАП и РММАП на АРМ пользователей, в виде архивов. Имена архивов формируются по следующему шаблону: computer_name+_+user_login@user_domain+_+время создания архива в UTC+.+генерация случайных чисел, обеспечивающих уникальность имени каждого файла.

Архивы могут содержать следующие файлы:

- manifest.json
- activity.json
- inspector_log.json
- service_log.json
- timetracking_log.json

Разные типы архивов содержат разный набор файлов.

Архив инспектора агента	**Архив службы агента**	**Архив модуля Таймтрекинг**
manifest.json	manifest.json	manifest.json
activity.json	service_log.json	timetracking_log.json
inspector_log.json	—	—

Информация об активности пользователей хранится в таблице main.monitoring_activity.

Логи агента мониторинга, передаваемые с рабочих мест на сервер приложения, хранятся в таблице main.monitoring_agent_inspector_log.

manifest.json

В *manifest.json* фиксируется основная информация о пользователе, а также информация о типе логов. В поле **type** определяется от кого пришел архив — от инспектора (**inspector**), службы (**service**) или от трекинга (**timetracking**).

Пример:


```
{
  "version": "1.0.4",
  "machine_guid": "3c830f47-9fcd-4a48-a568-750d8b179987",
  "user": {
    "name": "Иванов Иван",
    "login": "ivanov",
    "domain": "CORP.BUSINESSPROJECT.COM",
    "ad_guid": "d59396ef-c493-4215-8dd5-1c478c846191",
    "timezone": "Europe/Moscow",
    "timezone_sec": 10800,
    "employee_id": 2
  },
  "computer": {
    "name": "c-066-im.corp.infomaximum.com",
    "domain": "domain.com",
    "workgroup": ""
```

```
  },
  "agent": {
    "version": "2.14.3"
  },
  "type": "inspector"
}
```

Где:

- version — версия протокола (формат данных для сервера)
- machine_guid — уникальный идентификатор устройства, с которого был собран архив
- user:name — имя пользователя (учетная запись ОС)
- user:login — логин пользователя (учетная запись ОС)
- user:domain — домен пользователя (при наличии)
- user:ad_guid — уникальный идентификатор пользователя в Active Directory
- user:timezone — часовой пояс
- user: timezone_sec — смещение часового пояса относительно UTC в секундах
- user: employee_id — внутренний id пользователя
- computer:name — имя АРМ
- computer:domain — домен АРМ
- computer: workgroup — рабочая группа АРМ
- agent:version — версия агента
- type — тип источника данных, от которого получен архив

activity.json

Активность сотрудников отправляется в JSON-формате в файле *activity.json* и представляет из себя массив записей.

Каждая запись обязательно содержит следующие поля:

```
{"version":"1.0.9"}
{
"time":1692694112,
  "time_ms":469,
  "cpu_loading":12,
  "memory_loading":30,
  "window_activity":{
    "app_info":{
      "program_name":"Google Chrome",
      "version":"115.0.5790",
    "executable_path":"%ProgramFiles(x86)%\\Google\\Chrome\\Application\\chrome.exe"},
    "type":43,
    "input_type":1
    "main_window":"",
    "domain":"",
    "url_path":"",
    "tab":"",
    "file_name":"index (3).html",
    "file_path":"%USERPROFILE%\\Desktop\\",
    "location":[],
    "element":null,
    "parameters":[]
```

```
}
}
```

Где:

Примечание. В зависимости от события некоторые поля могут быть пустыми.

- version — версия манифеста
- time — время события по UTC
- time_ms — время в миллисекундах
- cpu_loading — количество нагрузки на ЦП
- memory_loading — значение нагрузки на память
- window_activity — контейнер с информацией о событии
- app_info — информация о приложении
- program_name — название приложения из секции Description
- version — версия приложения
- executable_path — полный путь до исполняемого файла
- type — тип события
- input_type — параметр, отвечающий за определение типа активности: программная или аппаратная. Если непосредственно до совершенного события была аппаратная HID-активность, то параметр заполняется как input_type=1, если было программное событие, то input_type=0
- main_window — заголовок главного окна
- domain — домен окна
- url_path — URL окна
- file_name — имя открытого файла
- file_path — полный путь к открытому файлу
- location — иерархия окон со значением типа строго меньше 10:
    - name — название поля
    - type — код события
- element — элемент (например, кнопка, чекбокс)
- parameters — контейнер, содержащий следующую информацию:
    - name — название поля
    - value — хешированное значение
    - type — код события
    - is_current — информация о выполнении действия из текущего поля. Если параметр принимает значение 1 — действие выполнено в текущем поле, 0 — действие не было совершено в данном поле

Типы событий

Поле **type** представляет собой числовой код события. События сгруппированы по источнику в диапазоны.

Код события	Описание
1	Start — старт агента мониторинга, при нормальной работе агента (без сбоев и завершений из диспетчера задач) совпадает со стартом или разблокировкой пользовательской сессии
2	Stop — стоп агента мониторинга, при нормальной работе агента совпадает с блокировкой или завершением пользовательской сессии
3	ProcessCrashed — остановка работы приложения
	Оконная иерархия

Код события	Описание
40	WindowSwitch — переключение окон, переключение вкладок в браузере, а также любое переключение между ранее открытыми окнами программ
41	WindowUpdate — изменение заголовка окна
42	WindowOpen — открытие нового окна: браузера, страницы в новой вкладке браузера, вкладки в программе и т. п.
43	DocumentOpen — открытие нового документа
44	DocumentSave — сохранение документа под другим именем
45	SheetOpen — изменение листа в Excel (переключение на другой лист, переименование или открытие нового)
	Оконные элементы (только расширенный мониторинг)
52	Invoke — клик по UI-элементам (кнопки, пункты меню и др.)
53	FieldEdit — редактирование элементов Edit и Document. Фиксируется после завершения редактирования и смены фокуса на другой элемент. Элемент Document фиксируется только для отдельных приложений. Название и значение поля фиксируется в parameters
56	FileSelect — выбор файла, папки через соответствующее окно открытия. Путь к файлу и его название фиксируются в parameters в поле select_from
58	FileSave — сохранение файла в диалоговом окне, а также скачивание файла в браузере Google Chrome. Путь к файлу и его название фиксируются в parameters в поле save_to
60	Copy — копирование или вырезание текста. Текст отображается в parameters в поле copy_value
61	Paste — вставка текста. Текст отображается в parameters в поле paste_value
62	ValueSelect — выбор значения в комбобоксе, чекбоксе или радиокнопке (с радиогруппой или без таковой), если в этом значении нет извлекаемых данных
63	Respond — кнопка или гиперссылка, по которой было событие Invoke, стала невидимой
64	FormRespond — событие, записываемое вместо Respond, если в этом окне было изменение parameters полей
65	Select — выбор элементов в листе, таблице и дереве
66	Print — отправка документа на печать
	HID-активность
90	InjectedMouseClick — клик мышкой программный
91	InjectedMouseScroll — прокрутка мышкой программная
92	InjectedKeyboardEvent — нажатие кнопки клавиатуры программное
95	HardwareMouseClick — клик мышкой аппаратный
96	HardwareMouseScroll — прокрутка мышкой аппаратная
97	HardwareKeyboardEvent — нажатие кнопки клавиатуры аппаратное

Control-элементы

В базовом мониторинге могут быть только Window и ExcelSheet, в расширенном— все.

Значение	Код	Описание
Window	1	Окно
Button, Splitbutton	10	Кнопки
CheckBox	12	Чекбокс. Также CheckBox заменяет RadioButton, если Radiogroup не была найдена
ComboBox	13	Комбобокс. Также ComboBox заменяет RadioButton, если для него была найдена Radiogroup
Edit	14	Текстовые поля
Hyperlink	15	Гиперссылки
ListItem	17	Списки
TreeItem	34	Древовидные списки
MenuItem	21	Пункты меню
TabItem	29	Вкладка

Значение	Код	Описание
DataItem	39	Элемент таблицы
Document	40	Текстовые поля в Outlook
HotKey	49	Горячие клавиши
Location	50	Отображается только в колонке parameters_type для полей, указывающих на место (save_from, save_to, excel_sheet и т. п.)
Clipboard	51	Отображается только в колонке parameters_type для полей copy_value и paste_value
UrlParam	52	Отображается только в колонке parameters_type для полей, в которых содержится параметр, извлеченный из URL
NameParam	53	Отображается только в колонке parameters_type для полей, в которых содержится параметр, извлеченный из названия окна либо названия элемента, кроме поля Номер в 1С, для которого указывается тип Edit
Id	54	Отображается только в колонке parameters_type для полей, в которых содержится идентификатор окна
Message	55	Отображается только в колонке parameters_type для поля message от окна подтверждения
ExitCode	56	Отображается только в колонке parameters_type для поля exit_code от события аварийного прекращения процесса
ExcelSheet	57	Название листа в Excel

Горячие клавиши

В событии Invoke фиксируются комбинации с модификаторами Ctrl, Alt, Shift, а также функциональные F1—F12 . Не фиксируются сочетания Shift+A, где A — любая односимвольная клавиша. Например, такие сочетания используются при написании заглавных букв.

Регистр фиксации важен: модификаторы Ctrl, Alt, Shift должны быть именно в таком виде, F1—F12 заглавными буквами. Клавиша в сочетании с модификатором должна быть в верхнем регистре, например, Ctrl+X.

Если в названии элемента есть горячие клавиши, то в качестве имени в element_name записываются только они в нужном регистре, а остальная часть названия фиксируется в колонке element_hotkey_name.

inspector_log.json

В *inspector_log.json* передаются логи инспектора агента. В логах содержится информация о старте инспектора, периодичность запросов на обновление, тип мониторинга сотрудника.

Пример логов:

```
{"version":"1.0.0"}
PID=9212 TID=23400 important Start Inspector started, version = 2.14.3, user: nnesterova@CORP.INFOMAXIMUM.COM, settings: {"monitoring_status":2,"timetracking_status":false,"logger":{"level":"LEVEL_ERROR"}}, loggerSetting: LEVEL_ERROR
PID=9212 TID=23400 important system_info::GetUserAdGuid user sid = S-1-5-21-580362855-3340923925-1044332590-1417
PID=9212 TID=23400 important ActivitySender::ActivitySender reservedSpace file size = 1048576b
PID=9212 TID=23400 important ModuleManager::LoadModule module 'C:\Program
```

Files\ProcesetAgent\mod_extended_monitoring.dll' loaded
PID=9212 TID=23400 important Controller::Start ActivityCollectorExtended started
PID=9212 TID=23400 important ModuleManager::LoadModule module 'C:\Program Files\ProcesetAgent\mod_crash_watcher.dll' loaded
PID=9212 TID=23400 important Controller::Start ProcessCrashWatcher started
PID=9212 TID=9116 important SettingsManager::RefreshUserId EnsureUser success, employeeId = 2
PID=9212 TID=9116 important SettingsManager::RefreshServerComponents GetServerComponents success, components: com.infomaximum.subsystem.activedirectory com.infomaximum.subsystem.clickhouse.core com.infomaximum.subsystem.clickhouse.standalone com.infomaximum.subsystem.core com.infomaximum.subsystem.dashboard com.infomaximum.subsystem.dashboardext com.infomaximum.subsystem.frontend com.infomaximum.subsystem.monitoring com.infomaximum.subsystem.workspaces
PID=9212 TID=9116 important SettingsManager::RefreshSettings GetSettings success, employeeInfo: employeeId = 2, displayName = 'Нестерова Наталья', logLevel = 'LEVEL_ERROR', monitoringType = Extended, trackingEnabled = false
PID=9212 TID=1484 important UserActivity::ExtractWindowActivity activityEventCount = 3, utcTime = [2021.12.06_04:41:42 - 2021.12.06_04:41:47]
PID=9212 TID=1484 important `anonymous-namespace'::PackActivity Timezone init id = Europe/Moscow, offsetInSec = 10800

service_log.json

В *service_log.json* собирается информация о старте службы, блокировке системы под конкретным пользователем, отправленных сбоях.

Пример логов:

{"version":"1.0.0"}
PID=6556 TID=17352 important main Service started, version = 2.14.3
PID=6556 TID=1116 important SessionController::StartInspector Inspector started, sessionId = 14, UserPrincipalName = 'nnesterova@CORP.INFOMAXIMUM.COM'
PID=6556 TID=17352 important RunAsService::<lambda_6f040d3ff28f03704b9026c4883d74c7>::operator () Session event WTS_SESSION_LOCK, sessionId = 14
PID=6556 TID=17352 important SessionController::OnSessionDeactivated Session deactivated, sessionId = 14, UserPrincipalName = 'nnesterova@CORP.INFOMAXIMUM.COM'
PID=6556 TID=17352 important RunAsService::<lambda_6f040d3ff28f03704b9026c4883d74c7>::operator () Session event WTS_SESSION_UNLOCK, sessionId = 14
PID=6556 TID=1116 important SessionController::StartInspector Inspector started, sessionId = 14, UserPrincipalName = 'nnesterova@CORP.INFOMAXIMUM.COM'

timetracking_log.json

В *timetracking_log.json* собираются логи модуля таймтрекинга (время старта модуля, информация о пользователе и т. д.).

Пример логов:

{"version":"1.0.0"}
PID=660 TID=5172 important main Timetracking started, version = 2.14.3, user: client02@TEST.INFOMAXIMUM.RU, loggerSetting: LEVEL_ERROR

PID=660 TID=5172 important IdleManager::ProcessResumeInfo Save on server user activity = Пт ноя 26 11:33:32 2021
PID=660 TID=5172 important IdleManager::ProcessResumeInfo NotifyActiveTrackIdleResume

Уровень логирования

Изменяя уровень логирования, можно настраивать степень детализации технических логов, собираемых агентом мониторинга. Уровень логирования задается в конфигурационном файле com.infomaximum.subsystem.monitoring.json.

Отличия в уровнях логирования

Примечание. По умолчанию для всех пользователей установлен уровень логирования ERROR.

Доступны следующие уровни:

- INFO
- WARNING
- ERROR
- CRITICAL
- OFF

Выбор уровня определяет, какие сообщения будут сохраняться в логах. Разделение по уровням отражает значимость записей:

- CRITICAL — критические ошибки, при которых агент не может продолжить работу. Например:
  - Затруднения со сбором активности
- ERROR — ошибки, не критичные для продолжения работы, но требующие внимания. Например:
  - Отсутствие подключения к серверу при отправке активности
  - Ошибки при попытке получить доступ к буферу обмена (Clipboard) в операционной системе
- WARNING — потенциальные ошибки или важная диагностическая информация. Например:
  - Сообщения о создании скриншота для элемента
  - Записи об обновлении списка полей в событии после сканирования элементов на форме
- INFO — вспомогательная информация, не логируемая по умолчанию. Например:
  - Запросы при отправке архивов с активностью
  - Сообщения о завершении работы программ, фиксирование PID процессов

Иерархия уровней логирования

Уровни логирования образуют иерархию: каждый последующий уровень включает сообщения всех предыдущих. Порядок уровней: CRITICAL → ERROR → WARNING → INFO.

Например:

- При уровне CRITICAL в лог записываются только сообщения уровня CRITICAL
- При уровне ERROR записываются сообщения уровней CRITICAL и ERROR
- При уровне WARNING — CRITICAL, ERROR и WARNING, и т. д.

Кроме того, при использовании уровня ERROR (и выше) дополнительно фиксируются записи с условным уровнем IMPORTANT, содержащие ключевую информацию о работе агента и пользователе:

- Сведения об агенте мониторинга: версия приложения, текущий уровень логирования, тип мониторинга, включены ли скриншоты
- Данные о пользователе: отображаемое имя (displayName), идентификатор в системе (employeeId), SID доменного пользователя
- Информация об операционной системе (в событии старта инспектора после метки OsInfo)
- Часовой пояс пользователя
- Запуск и остановка модулей сбора активности (базового или расширенного)
- Объем памяти, используемой агентом
- Количество переключений между окнами и список заголовков окон за период сбора активности (используется для диагностики проблем)
- Записи об отключении подписки на события фокуса
- Сообщения о сбоях приложений

Описание структуры хранения данных в ClickHouse

Все описанные таблицы созданы на основе движка ReplacingMergeTree.

Предупреждение. Не вносите изменения в таблицы с данными мониторинга main.monitoring_activity и main.monitoring_agent_inspector_log. Это может привести к серьезным сбоям в работе системы и потере данных.

Таблица «monitoring_activity»

В таблице хранится информация об активности пользователей. Путь для обращения к таблице по умолчанию: main.monitoring_activity.

Знаками ✓ и ✕ отмечены наличие или отсутствие колонки в зависимости от типа мониторинга.

Имя колонки	Тип	Описание	Базовый мониторинг	Расширенный мониторинг
agent_version	String	Версия агента мониторинга	✓	✓
computer_name	String	Имя компьютера	✓	✓
cpu_loading	UInt32	Значение загрузки процессора в процентах (от 0 до 100)	✕	✓
domain	String	Домен	✓	✓
element_ctrl	UInt8	Контрол элемента element_name	✕	✓
element_hotkey_name	String	Название пункта меню	✕	✓
element_name	String	Имя элемента (например, кнопка или чекбокс) с типом 10 и более	✕	✓
employee_account_id	UInt64	Идентификатор сущности EmployeeAccount во встроенной файловой базе данных модуля мониторинга	✓	✓
executable_path	String	Полный путь до исполняемого файла	✓	✓
file_name	String	Имя открытого файла, может быть пустым	✓	✓
file_path	String	Полный путь к открытому файлу, может быть пустым	✓	✓
id	UInt64	Идентификатор доменного элемента во встроенной файловой базе данных	✓	✓
input_type	UInt8	Если активность является аппаратной, то параметр заполняется как 1, если программной, то как 0	✓	✓
location_name	Array(String)	Иерархия окон	✓	✓
location_type	Array(UInt32)	Тип окна	✓	✓
main_window	String	Заголовок окна	✓	✓

Имя колонки	Тип	Описание	Базовый мониторинг	Расширенный мониторинг
memory_loading	UInt32	Значение использования оперативной памяти в процентах (от 0 до 100)	✕	✓
parameters_name	Array (String)	Название элемента, из которого получено значение	✓	✓
parameters_type	Array(UInt32)	Тип элемента	✓	✓
parameters_value	Array (String)	Значение поля может отображаться в явном виде, если это специальное поле (например, Организация в 1С) или если выключено хеширование, а также в виде хеша в формате crc32:<32-битное_число>	✓	✓
parameters_is_current	Array(UInt8)	Значение 1 для текущего редактируемого поля и всех извлеченных из него параметров Значение 0 для поля, которое не редактировалось	✓	✓
program	String	Название запущенной программы	✓	✓
research_id	UInt64	Идентификатор исследования, в рамках которого сделан скриншот	✕	✓
screenshot_id	String	Идентификатор скриншота	✕	✓
tab	String	Вкладка главного окна, может быть пустым	✓	✓
time_offset	Int8	Часовой пояс (сдвиг в часах от времени в UTC)	✓	✓
time	DateTime64	Время события, инициированного пользователем, представлено как календарная дата и время, с заданной до миллисекунд и наносекунд точностью	✓	✓
type	Int16	Тип события	✓	✓
url_path	String	Путь из URL после домена	✓	✓
version	String	Версия запущенной программы	✓	✓

Колонки, которые присутствуют в базовом мониторинге, не всегда могут быть заполнены одновременно. Для каждого события (кроме 1 — старт, 2 — стоп и 3 — падение), заполняются следующие колонки:

- id

- time
- employee_account_id
- time_offset
- program
- version
- executable_path
- type
- input_type
- computer_name
- agent_version

В зависимости от типа программы в базовом мониторинге могут быть дополнительно заполнены следующие колонки:

- Браузеры: main_window, domain и url_path
- Программы для работы с документами (например, Microsoft Office): file_path и file_name
- Другие настольные программы: main_window

Если агент мониторинга зафиксировал идентификатор окна или название листа Excel в колонке parameters_name, в базовом мониторинге также могут быть заполнены следующие колонки:

- parameters_name
- parameters_value
- parameters_is_current
- parameters_type

Помимо стандартного пути main.monitoring_activity, в скриптах автоматизации можно использовать переменную activity_table, которая хранит путь к таблице monitoring_activity. Это позволяет автоматически формировать корректный путь к таблице при изменении инфраструктуры и минимизирует риск ошибок. В скриптах можно обратиться к таблице через подстановку переменной в запрос или код.

Значение переменной activity_table задается в конфигурационном файле com.infomaximum.subsystem.monitoring.json через параметры monitoring_database_name и storage_guid.

Таблица «monitoring_agent_inspector_log»

В таблице хранятся технические логи агентов мониторинга, передаваемые с рабочих мест на сервер приложения. Данные используются при первичной диагностике проблем с агентами мониторинга. Путь для обращения к таблице по умолчанию: main.monitoring_agent_inspector_log.

Поле	Тип	Описание
time_offset	Int8	Смещение (UTC)
time	DateTime64(3, UTC)	UTC время в секундах
level	String	Уровень логирования записи в колонке message. Может принимать значения: CRITICAL, ERROR, IMPORTANT, WARNING, INFO

Поле	Тип	Описание
pid	Int32	id процесса, 32-битное целое число
tid	Int32	id потока, 32-битное целое число
function	String	Название функции в коде, которой соответствует запись в колонке message
message	String	Текст лога: - Версия агента, тип мониторинга - Имя пользователя, его id в системе - Версия операционной системы - Периодичность отправки архивов на сервер и т. д. В зависимости от уровня логирования записей меняется объем записываемых данных
machine_guid	String	id компьютера (зависит от ОС), позволяет узнать, с какого именно АРМ пришла активность
agent_version	String	Версия агента
remote_address	String	Заполняется строковым представлением ip-адреса клиента пришедшим на сервер
computer_name	String	Берётся наименование компьютера из файла manifest.json из архива активности
employee_account_id	UInt64	Идентификатор элемента EmployeeAccount во встроенной файловой базе данных модуля мониторинга
type_log	String	Тип записи в колонке message: - inspector – логи инспектора - service – логи службы SYSTEM - timetracking – логи модуля timetracking
row_number	UInt64	Порядковый номер записи в таблице

Срок хранения информации по умолчанию составляет 14 дней. Его можно изменить через параметр partition_life_circle_time в конфигурационном файле com.infomaximum.subsystem.monitoring.json.

Хранение логов агента мониторинга

Логи агента мониторинга используются для диагностики его работы, выявления сбоев и проверки корректности настроек. Эта информация может потребоваться в следующих случаях:

- При обращении в техническую поддержку
- При анализе причин недоступности сервера
- При необходимости проверить, корректно ли работает автообновление, сбор активности или отправка данных
- Для отслеживания ошибок установки или конфигурации агента

Расположение логов агента мониторинга на Windows

Каталоги, которые создаются и используются агентом мониторинга на ОС Windows:

- C:\Program Files\ProcesetAgent — исполняемые файлы и необходимые dll-модули программы
- C:\ProgramData\ProcesetAgent — общие для компьютера файлы работы программы
- C:\ProgramData\ProcesetAgent\dumps — аварийные дампы агента. Обычно этот каталог пустой, так как дампы передаются на сервер приложения

Файлы конфигурации:

- C:\ProgramData\ProcesetAgent\settings.cfg – файл общей конфигурации агента мониторинга для компьютера. В этом файле можно изменить параметры автообновления, адрес сервера, настройки прокси, API ключ. Изменения в этом файле применяются после перезапуска службы агента мониторинга или перезагрузки компьютера
- %AppData%\ProcesetAgent\settings.cfg – файл конфигурации агента мониторинга по пользователю. В этом файле есть возможность изменить статус мониторинга за пользователем (0 – выключен, 1 – базовый мониторинг, 2- расширенный мониторинг), проверить статус таймтрекинга, указать уровень логирования агента мониторинга. Изменения в этом файле вступают в силу после LogOff\LogOn пользователя, перезапуска службы агента мониторинга, перезагрузки компьютера или через час после изменения файла

Файлы логов:

- C:\ProgramData\ProcesetAgent\logs\Service.log – лог-файл службы агента мониторинга
- %AppData%\ProcesetAgent\logs\sessionInspector.log – лог-файл работы агента мониторинга по пользователю

Другие исполняемые файлы:

- C:\Program Files\ProcesetAgent\agent_configurator.exe – утилита agent_configurator
- C:\Program Files\ProcesetAgent\agent_setup.exe – файл-инсталлятор, может служить для удаления программы с компьютера

Расположение логов агента мониторинга на Linux

Ниже представлен список каталогов с логами агента мониторинга на ОС семейства Linux.

Каталоги, которые создаются и используются агентом мониторинга в Linux:

- ~/.ProcesetAgent/screenshots– скриншоты, сделанные в рамках мониторинга активности пользователя
- /var/lib/ProcesetAgent/dumps – аварийные дампы агента
- /var/lib/ProcesetAgent/data – файлы со статистикой работы агента

Файлы конфигурации:

- /var/lib/ProcesetAgent/[username] – папка с файлом, в которой содержится путь к каталогу конкретного пользователя ProcesetAgent
- /var/lib/ProcesetAgent/settings.cfg — глобальный конфигурационный файл агента (настройки подключения к серверу, прокси, API-ключ, автообновление и др.)
- ~/.ProcesetAgent/settings.cfg — конфигурационный файл с настройками мониторинга для конкретного пользователя (уровень мониторинга, включен ли таймтрекинг, уровень логирования и др.)

Файлы логов:

- /var/lib/ProcesetAgent/logs – лог-файлы системной службы Service
- ~/.ProcesetAgent/logs– каталог с логами компонента SessionInspector

Директория с исполняемыми файлами агента:

- /usr/libexec/ProcesetAgent – каталог установки, содержит исполняемые файлы и компоненты агента

Сбор скриншотов агентом мониторинга [!BETA]

Примечание. Эти функции доступны в бета-версии. Попробуйте их в работе и поделитесь своим мнением — обратная связь помогает нам развивать продукт.

С помощью запуска исследований агент мониторинга собирает скриншоты при определенных действиях пользователей. Полученные скриншоты можно хранить, обрабатывать и анализировать. Скриншоты позволяют ускорить время запуска проектов Task Mining, а также повышают качество разметки операций. Сбор скриншотов доступен только для расширенного мониторинга.

Агент мониторинга делает снимок экрана при клике пользователем на экранные формы. Скриншот создается для следующих событий:

- Клик по кнопке, ссылке и другим элементам. Код в активности — 52
- Редактирование поля. Код в активности — 53
- Копирование текста. Код активности — 60
- Вставка текста. Код активности — 61
- Выбор значения в комбобоксе, клик по радиокнопке или чекбоксу. Код активности — 62
- Выбор значения в списке или таблице. Код в активности — 65
- Текущий активный элемент выделяется:
- Прямоугольной рамкой, если агент получил у этого элемента координаты и они по ширине и высоте больше пяти пикселей
- Кругом с центром в месте клика мышкой, если координаты не удалось получить или прямоугольник по ширине или высоте меньше пяти пикселей

Идентификатор скриншота фиксируется в таблице «monitoring_activity» в колонке screenshot_id. Сами скриншоты хранятся во встроенной файловой базе данных Proceset и удаляются из нее после завершения исследования. Средний размер одного скриншота — 50 Кб.

Примечание. Размер архивов со скриншотами может в несколько раз превышать объем данных активности без изображений. Не рекомендуем включать сбор скриншотов для всех пользователей.

Агент мониторинга передает собранные скриншоты вместе с активностью на сервер Proceset. Архив, передаваемый агентом, содержит 10 минут активности сотрудника. Рассчитать точный размер архива невозможно: количество скриншотов зависит от количества событий, которые индивидуальны для каждого бизнес-процесса. Примерное значение архива с данными активности за день можно получить по формуле (количество событий x средний размер скриншота) x 6 x 8, где:

- количество событий x средний размер скриншота — активность сотрудника за 10 минут
- количество событий x средний размер скриншота x 6 — активность сотрудника за 1 час
- 8 — количество рабочих часов

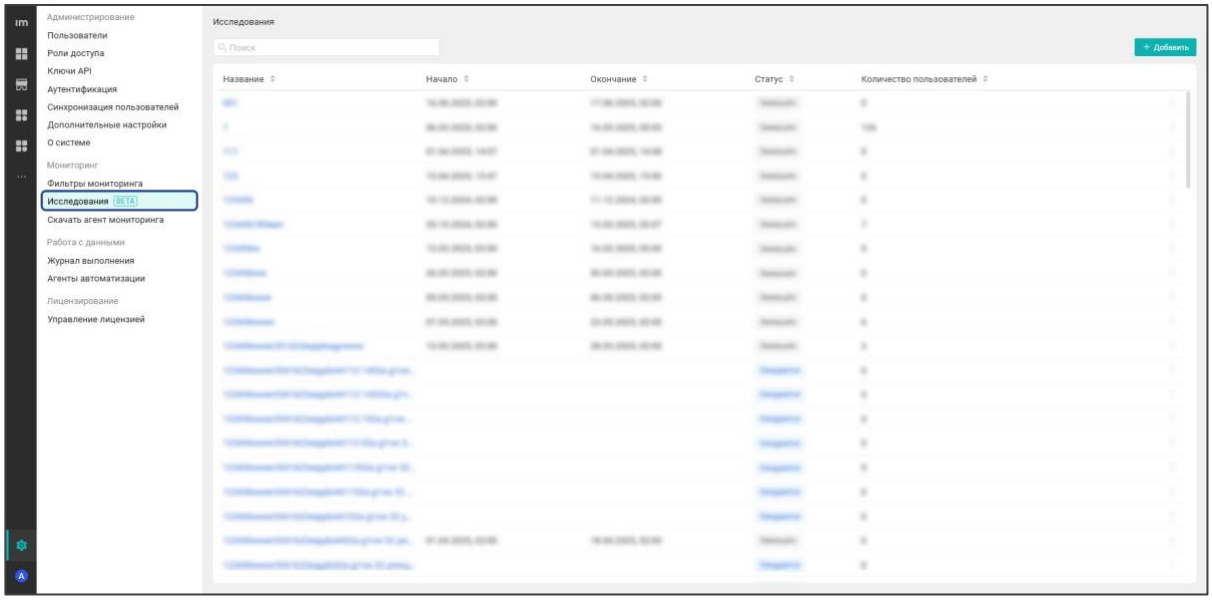
Ограничения по сбору скриншотов агентом мониторинга

При сборе скриншотов агентом мониторинга существуют следующие ограничения:

- При клике по кнопке, ссылке или другим элементам:
  - При использовании горячих клавиш события фиксируются, но скриншоты не создаются
  - При нажатии на элемент в контекстном меню такое меню пропадает. Вместо него на скриншоте отображается выделенная область, где находился элемент меню
- При редактировании поля: если у элемента нет координат, то скриншот не создается. Координаты мыши не используются, так как для такого события они неактуальны
- При выборе значения: если выбран ComboBox-элемент, у которого нет координат, то скриншот не создается. Координаты мыши не используются, так как для такого события они неактуальны
- При копировании текста: если копируется элемент, который не относится к parameters-элементам, либо элемент без координат, то скриншот не создается
- При вставке текста: если вставка происходит в элемент, который не относится к parameters-элементам, либо в элемент без координат, то скриншот не создается
- События Copy и Paste фиксируются при изменениях в буфере обмена. Эти события могут фиксироваться после того, как произошло следующее событие после копирования или вставки. В этом случае скриншот для таких событий не создается, поскольку изображение на экране уже изменилось. Если события Copy или Paste происходят через контекстное меню, то элемент, который копируется или вставляется, определяется при постобработке. В таком случае скриншот наследуется от предыдущего события
- В главном окне программ MS Excel и Word отключен фокус. При работе с главным окном расширенный мониторинг не регистрирует события и не создает скриншоты

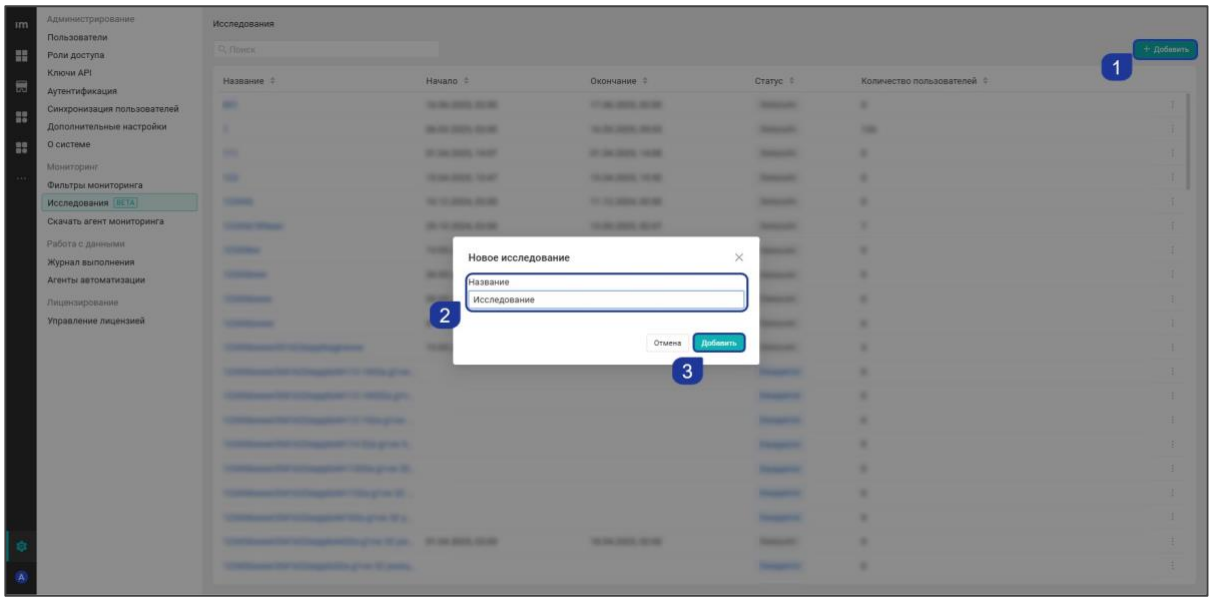
Настройка сбора скриншотов

Чтобы настроить сбор скриншотов, перейдите на страницу *Исследования* из меню настроек системы.



Примечание. За доступ к разделу *Исследования* отвечает привилегия Исследования.

Чтобы создать исследование, нажмите кнопку + **Добавить** в правом верхнем углу, укажите его название в открывшемся окне и кликните по **Добавить**.



После создания исследования открываются его настройки:

- Название — уникальное название исследования
- Срок проведения — время начала и окончания сбора скриншотов • Размытие — включение/выключение размытия скриншотов
- Статус — статус исследования:
    - *Ожидает запуска* — исследование еще не запущено, необходима настройка
    - *В процессе* — исследование собирает скриншоты. Если исследование остановить, запуск исследования будет доступен, пока не наступит дата завершения
    - *Завершено* — исследование не собирает скриншоты, наступила дата завершения или исследование остановлено

Если включено размытие, то применяется размытие скриншота, кроме области текущего элемента. Если текущий элемент в круге или значение должно хешироваться, то к текущему элементу так же применяется размытие.

Степень размытия задается в конфигурационном файле мониторинга, по умолчанию установлено значение 50. Можно указать значение от 1 до 100. Если указать 0, размытие не будет применяться.

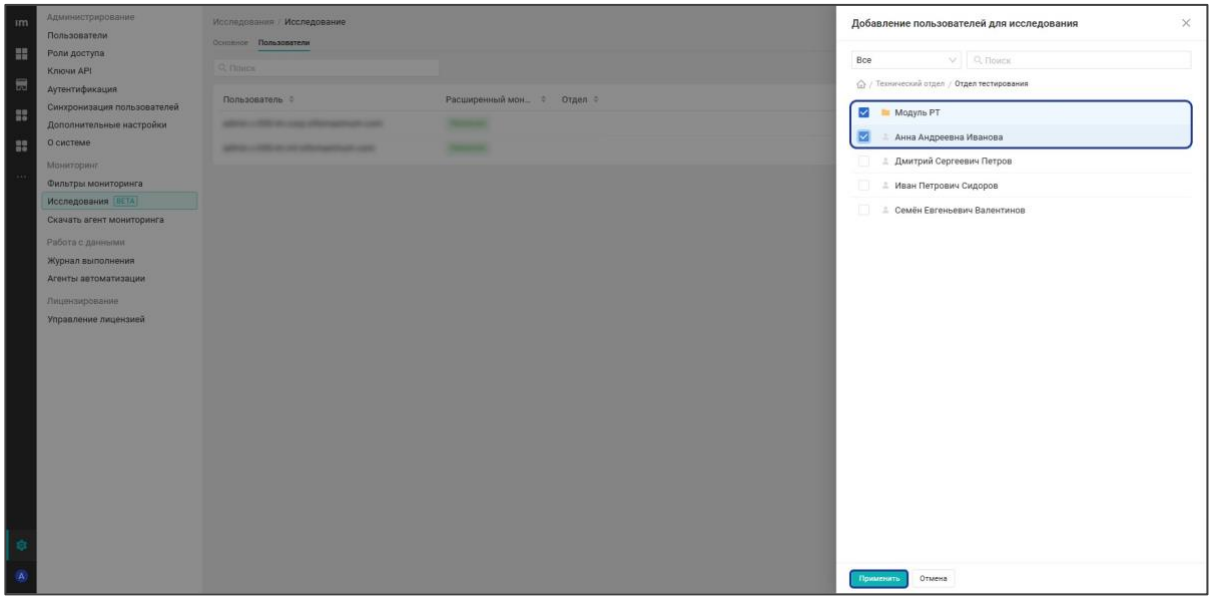
Чтобы добавить пользователей для исследования:

1. Перейдите во вкладку *Пользователи*.

2. В правом верхнем углу экрана нажмите **Добавить**.

3. Отметьте галочкой пользователей и/или отделы. Если отмечен отдел, в исследование будут добавлены все сотрудники отдела. Чтобы добавить только определенных пользователей, кликните по отделу и отметьте галочкой нужных сотрудников.

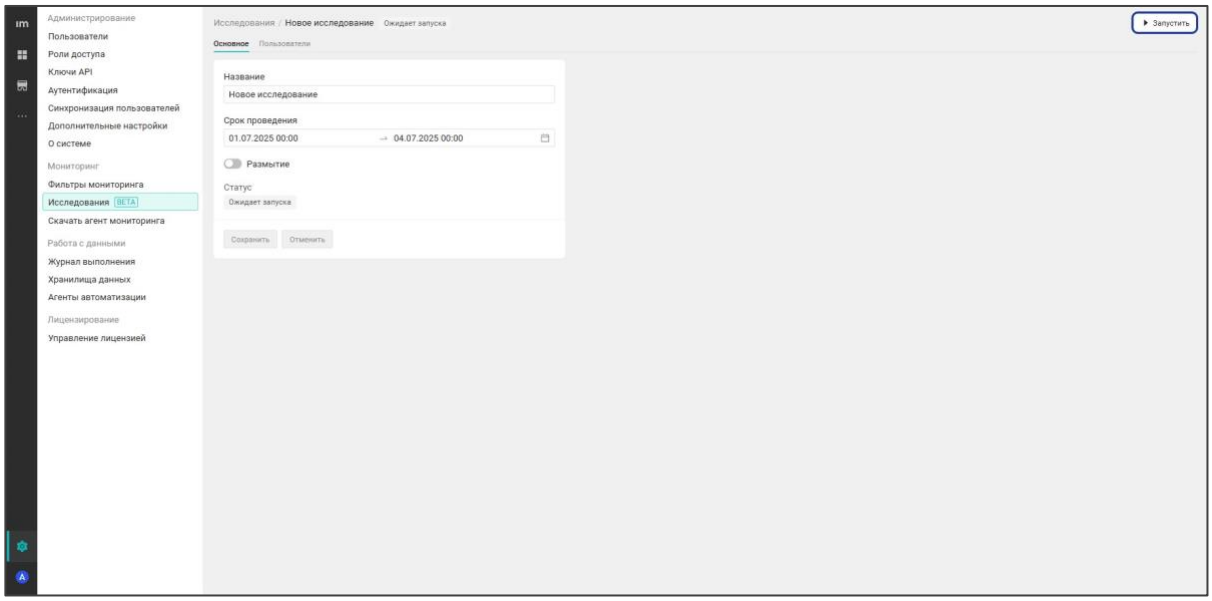
4. Нажмите **Применить**.



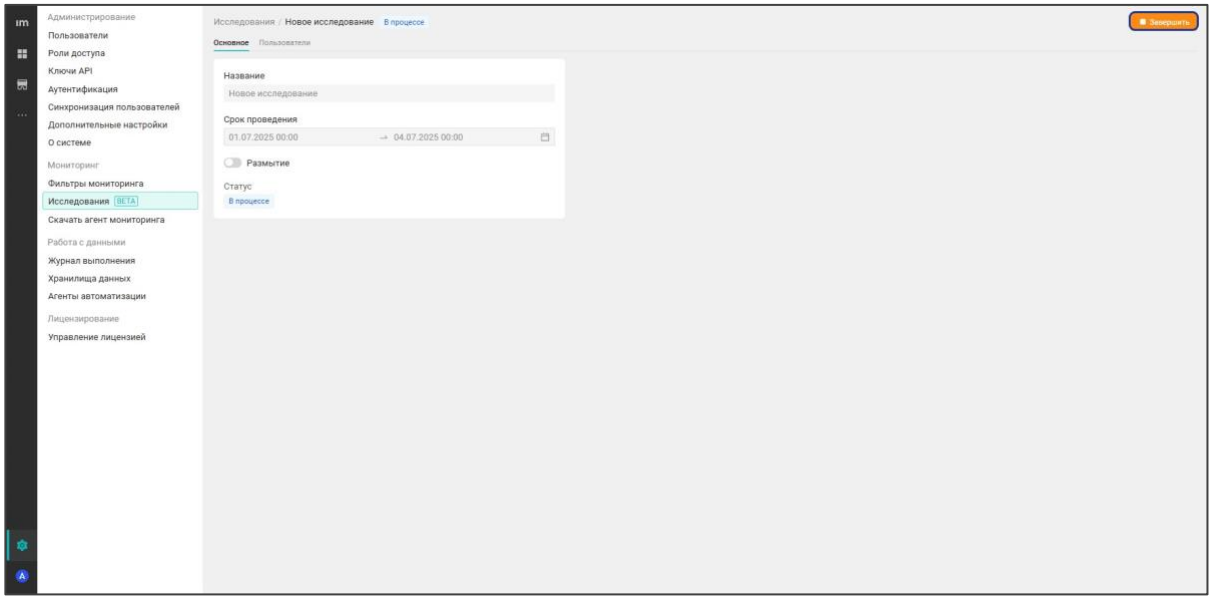
Примечание.

- Для исследования можно добавить пользователей, которым назначена любая лицензия по мониторингу.
- Для исследования можно добавлять пользователей, к которым нет доступа.
- Агент собирает скриншоты только от пользователей с лицензией **Мониторинг расширенный**.

Чтобы запустить исследование, нажмите **Запустить** на странице его настроек.



Если исследование запущено, то агент мониторинга собирает скриншоты в соответствии с заданными настройками. Настройки и список пользователей запущенного исследования недоступны для редактирования. После запуска исследования кнопка **Запустить** меняется на **Завершить**.



Если исследование завершено, то агент перестает собирать скриншоты. Для исследования, которое остановлено раньше даты завершения, можно изменить все параметры, кроме даты начала и списка пользователей. После остановки такого исследования кнопка **Завершить** заменяется на **Запустить**. Исследования, оканчивающиеся в соответствии с датой завершения, невозможно редактировать.

Идентификатор исследования, в рамках которого сделан скриншот, фиксируется в таблице «monitoring_activity» в колонке research_id.

Примечание. Собранные агентом скриншоты можно вывести в дашборд, используя виджет Текст. Подробное описание представлено в разделе Отображение скриншотов мониторинга в дашборде.

Экспорт и импорт активности пользователей

Экспорт и импорт активности пользователей предназначены для передачи данных мониторинга между серверами, создания резервных копий или восстановления данных в случае аварийного завершения работы. Эти функции используются:

- При миграции на другой сервер или кластер
- Для резервного копирования архивов активности
- При передаче активности между тестовым и продуктивным контурами
- Для интеграции данных мониторинга с другими системами анализа

В системе предусмотрен экспорт и импорт активности пользователей. Для выгрузки и загрузки активности необходимо подключить привилегию Экспорт/Импорт активности пользователей с операцией выполнения **E**.

Также у конфигурационного файла системы *com.infomaximum.subsystem.monitoring.json* должен быть выключен параметр по разбору и синхронизации архивов активности, принятой с агентов, и последующей передачей в СН. У параметра parsing_activity_enabled должно быть значение false.

Экспорт активности

Чтобы выгрузить данные активности в указанный каталог на диске, введите GraphQL-запрос:

```
mutation{
  activity_exchange{
    export_activity_queue(directory_path:"C:\\work\\Business-
Projects\\activity\\monitoring\\data\\backup", limit_per_file:25000){
      time_ms
      total_size
      count
      files
    }
  }
}
```

Где:

- directory_path — путь, куда экспортируются данные активности
- limit_per_file — лимит файлов в одном контейнере экспорта. Если лимит файлов не задан, то выгружаемые данные будут делиться по 25000 архивов активности

Поля запроса описаны в таблице ниже.

Поле	Тип данных	Определение
time_ms	Long	время выполнения операции в миллисекундах
total_size	Long	размер необработанных байтов, прочитанных из встроенной файловой базы данных с учетом наименования, но без учета внешних метаданных архива контейнера
count	Int	общее количество экспортируемых архивов с активностью из встроенной файловой базы данных

Поле	Тип данных	Определение
files	String	список имен файлов, сформированных в процессе экспорта в указанном каталоге

Во время экспорта запись новых архивов во встроенную файловую базу данных блокируется. Файл выгружается с названием *activity_queue({C}-{CN}){yyyy_MM_dd_HH_mm_ss_SSS}.zip*, где:

- {C} — первый ключ
- {CN} — последний ключ
- {yyyy_MM_dd_HH_mm_ss_SSS} — время формирования архива

Пример названия архива: *activity_queue(1-3)20221221162246000.zip*.

Важно.

- При экспорте данных из встроенной файловой базы данных не удаляйте соответствующие записи из ClickHouse.
- Данные во встроенной базе синхронизируются с ClickHouse, и при последующем импорте система использует ClickHouse для восстановления информации.
- Если записи в ClickHouse отсутствуют, процесс импорта запускает длительную процедуру обратной синхронизации или завершается с ошибками согласованности.

Данные активности не экспортируются, если:

- В GraphQL-запросе не указан путь, куда выгружаются данные
- На диске не существует путь или указан файл, а не каталог
- Не удалось блокировать запись активности в очередь в течении 800 мс. В этом случае ожидается повторный запрос

Импорт активности

Чтобы загрузить данные активности существующих пользователей и источников активности из указанного каталога на диске, введите GraphQL-запрос:

```
mutation{
  activity_exchange{
    import_activity_queue(directory_path:"C:\\work\\Business-
Projects\\activity\\monitoring\\data\\backup", is_create_by_login:true){
      time_ms
      total_size
      count
      files
    }
  }
}
```

Где:

- directory_path — путь на сервере, откуда импортируются данные активности
- is_create_by_login: true/false — параметр, который отвечает за проверку на наличие источников у пользователя и его создание при необходимости:
    - Если источник по login и domain есть в базе, считается, что источник существует и дальнейшая работа по нему не требуется

- Если источник по login существует, но нет соответствия по domain, из полученного источника извлекается идентификатор пользователя, и этому пользователю создается новый источник
- Если не найдено соответствие по login и domain, источник создается автоматически

Поля запроса описаны в таблице ниже.

Поле	Тип данных	Определение
time_ms	Long	Время выполнения операции MS
total_size	Long	Размер необработанных байтов, прочитанных из контейнера архива с учетом наименования
count	Int	Общее количество импортируемых архивов с активностью из встроенной файловой базы данных
files	String	Список имен файлов прочитанных операцией импорта

Предупреждение. Если источник активности и пользователь отсутствуют, необходимо выполнить GraphQL-запрос с использованием ключа API и из ссылки убрать букву i в слове «graphiql». Пример: *automation-dev.preview.infomaximum.com/graphql?&api_key=ключ*. После изменения ссылки необходимо обновить страницу.

Для импорта активности поместите архивы агента мониторинга в один архив с названием *activity_queue({C}{CN}){yyyy_MM_dd_HH_mm_ss_SSS}.zip*, где:

- {C} — первый ключ
- {CN} — последний ключ
- {yyyy_MM_dd_HH_mm_ss_SSS} — время формирования архива

Примечание. Можно указать приблизительное время формирования архива, необходимо только соблюсти обозначенный формат записи.

Данные активности не импортируются, если:

- В GraphQL-запросе не указан путь на сервере, откуда выгружаются данные
- На диске не существует путь или указан файл, а не каталог

Примечание:

- Ошибки, связанные с системой ввода-вывода, будут отображены в журнале событий
- Чтобы активность пользователя с одного сервера при импорте соответствовала этому же пользователю на другом сервере, ему необходимо создать источник с логином и доменом
- После импорта файлы с диска удаляются, данные активности записываются во встроенную файловую базу данных
- После загрузки активности на сервер у параметра parsing_activity_enabled необходимо указать значение true

Архивация активности

В системе предусмотрена архивация данных активности. Это позволяет уменьшить нагрузку на основную встроенную файловую базу данных, уменьшить размер базы и сократить количество фоновых процессов, связанных с активностью.

Важно. Архивация активности не затрагивает БД ClickHouse.

Создание экземпляра базы данных

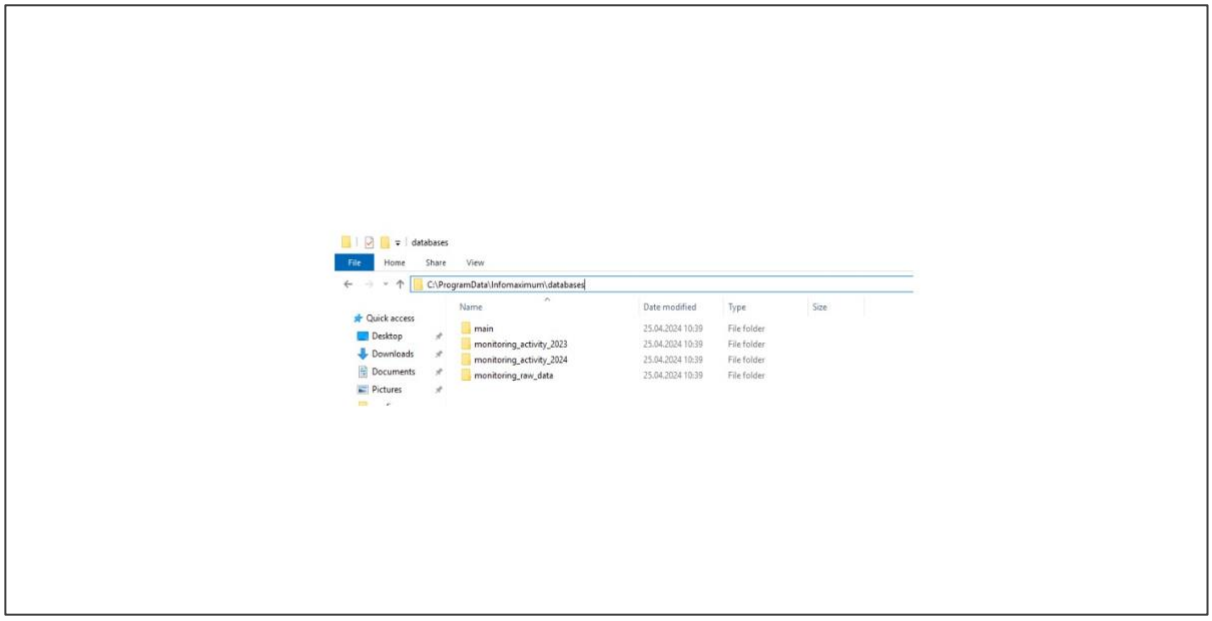
База данных за конкретный год создается в двух случаях:

- При миграции данных из текущей структуры на основании метаинформации о существующих в системе месячных колонках
- В случае поступления в систему активности, где год отличается от существующих в системе годичных баз данных, а также если отсутствует информация о ее отсоединении от системы

Сервер Proceset, получая пакет активности от агента мониторинга, записывает принятые данные в промежуточную встроенную файловую базу данных. Промежуточная БД расположена по пути {dataDir}databases\monitoring_raw_data. После этого выполняется сбор и обработка принятой информации и запись в другую встроенную БД, соответствующую году, за который поступила информация.

Новая база данных по умолчанию создается в каталоге {dataDir}, «рабочей директории» платформы — {dataDir}databases\monitoring_activity_{yyyy}, где:

- {dataDir} — рабочая директория
- monitoring_activity_ — префикс наименования базы данных
- {yyyy} — год колонки



Данные активности индексируются по порядку поступления на сервер. Первая строка с активностью, поступившей на сервер Proceset, получит id = 1, а поступившая десятой — id = 10.

Если в activity_database запись за {yyyy} год отсутствует, запускается процесс создания базы за указанный год. В каталоге {dataDir}\databases\ создается папка monitoring_activity_{yyyy}. Если в каталоге существует папка, то приложение записывает в лог соответствующее исключение. Инициализируется DBProvider — экземпляр подключения к встроенной файловой базе данных. В базе monitoring_activity_{yyyy} создаются колонки:

- activity_idx_{yyyy}
- activity_{yyyy}
- default

В ColumnFamily default заносятся данные:

- CREATE_DATE — дата создания экземпляра базы данных
- YEAR — год, к которому относится база данных
- GUID — уникальный идентификатор

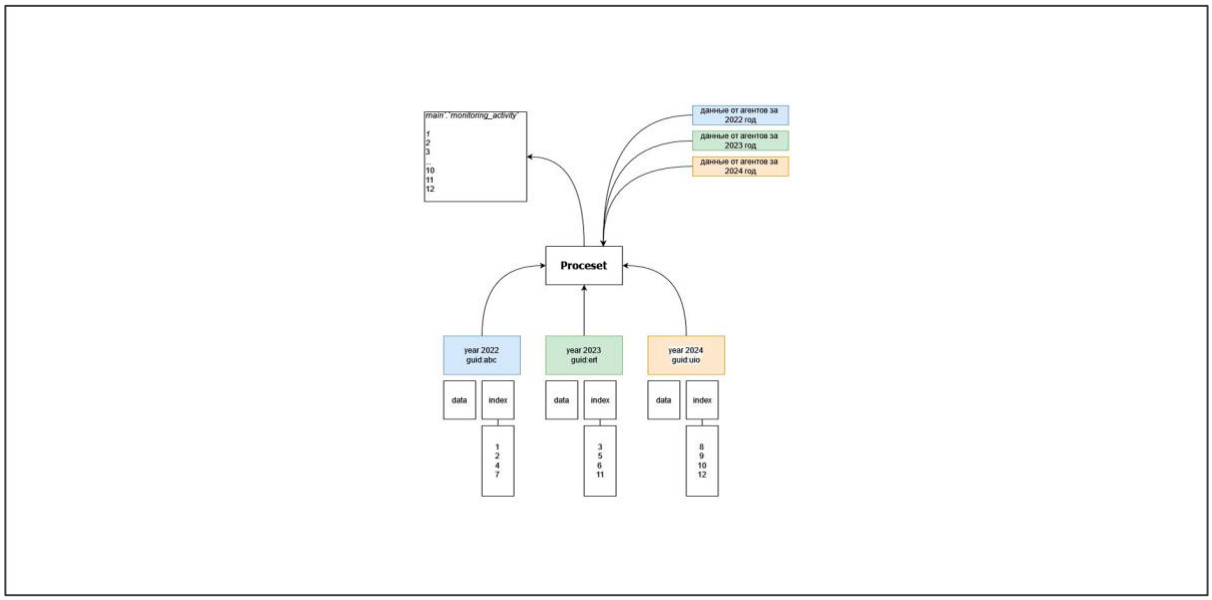
Если в процессе инициализации БД происходит ошибка, то приложение записывает в лог соответствующее исключение. В доменную сущность activity_database в базе мониторинга заносятся данные созданной БД:

- guid
- year

Новый экземпляр подключения к встроенной файловой базе данных регистрируется в сервисе записи активности.

Между встроенной файловой базой данных сервера Proceset и СУБД ClickHouse, которая используется как хранилище данных, происходит периодическая синхронизация данных активности.

На стороне СУБД ClickHouse данные записываются в таблицу main.monitoring_activity. В процессе синхронизации сервер Proceset определяет последний индекс активности, которая записана в указанной таблице ClickHouse, после записывает в нее недостающие данные относительно встроенной файловой СУБД.



Данные активности от агентов мониторинга хранятся одновременно и на стороне сервера Proceset, и на стороне сервера системной СУБД ClickHouse. Это сделано для надежности, так как данные от агентов мониторинга генерирует сама система Proceset, и их невозможно восстановить из других источников.

Удаление и восстановление данных активности на сервере СУБД ClickHouse

Сервер Proceset синхронизирует данные по последнему индексу активности, который есть в таблице на стороне СУБД ClickHouse. Если удалить строки до этого индекса, то они не будут заново синхронизированы.

Данными активности в таблице main.monitoring_activity можно оперировать до строки с самым последним индексом любым способом, который доступен в СУБД ClickHouse.

Важно. Будьте осторожны при выполнении операций **копировать**, **удалить**, **вставить**, так как они влияют на корректность данных в дашбордах, которые построены на данных активности.

Очистив таблицу (TRUNCATE) main.monitoring_activity, можно инициировать полную синхронизацию всех данных активности из встроенной СУБД Proceset в СУБД ClickHouse.

Удаление и восстановление данных активности на сервере Proceset

С сервера Proceset можно удалять данные активности по годам, отключая соответствующие внутренние БД. Для этого:

1. Остановите службу сервера Proceset (Infomaximum или Infomaximum-app в зависимости от ОС).

2. Вырежьте из каталога {dataDir}databases\monitoring_activity_{yyyy} каталог с данными за нужный год.

3. Запустите службу сервера Proceset.

После этого сервер Proceset не сможет обращаться к данным за этот год.

Если от агентов мониторинга придут данные за удаленный год, то они будут помещены в категорию поврежденных (corrupted). Поврежденные данные можно будет корректно обработать, если подключить обратно БД за нужный год.

Для того, чтобы подключить к серверу Proceset ранее удаленную БД с данными за год, выполните действия в обратном порядке:

1. Отключите службу сервера Proceset.

2. Поместите в каталог `{dataDir}databases` ранее вырезанную папку с БД.

3. Запустите службу.

Важно. Каждой встроенной БД с активностью система присваивает идентификатор GUID. Если GUID базы данных, расположенной в {dataDir}\databases\, неизвестен системе, то

эта БД не будет подключена. В связи с этим возможно подключение и отключение БД с активностью только в рамках одного экземпляра сервера Proceset.

Перенос данных активности на сервере Proceset в другую директорию

При необходимости файловые БД с данными активности на сервере Proceset можно перенести в другую директорию. Для этого:

1. Остановите службу сервера Proceset (Infomaximum или Infomaximum-app в зависимости от ОС).

2. Переместите из каталога {dataDir}\databases\monitoring_activity_{yyyy} каталог с данными за нужный год в нужную директорию.

3. Внесите в конфигурационный файл com.infomaximum.subsystem.monitoring.json новый блок по примеру ниже:

```
external_activity_databases:["C:\\activity_data_database\2021"]
```

Если таких БД несколько, укажите их через запятую.

4. Запустите службу сервера Proceset.

Работа с поврежденными архивами мониторинга

В системе Proceset сбор пользовательской активности осуществляется агентом мониторинга, который ежедневно отправляет на сервер архивы с данными. После приема сервер выполняет первичную валидацию архивов. Те, которые не соответствуют требованиям (например, поврежденный manifest.json, несуществующий пользователь, ошибки структуры), не попадают в ClickHouse и помещаются в специальную очередь corrupted.

Диагностика поврежденных архивов

Для работы с поврежденными архивами используется GraphQL API.

Получение статистики по очереди corrupted

Чтобы вывести количество архивов в очереди, выполните запрос:

```
{
 monitoring_diagnostics {
  corrupted_file_query {
   corrupted_file_statistic {
    all_size
    node_id
   }
  }
 }
}
```

Где:

- all_size — общее количество архивов в очереди corrupted
- node_id — идентификатор узла, на котором накоплены поврежденные файлы

Список поврежденных архивов с детализацией

Чтобы получить список поврежденных архивов, выполните запрос:

```
{
 monitoring_diagnostics {
  corrupted_file_query {
   corrupted_file_column_families {
    min_id
    max_id
    column_family_name
    node_id
    size
    corrupted_file_data_list {
     id
     source_file_name
    }
   }
  }
 }
}
```

Где:

- min_id, max_id — диапазон идентификаторов архивов в группе
- column_family_name — название группы
- node_id — идентификатор сервера
- size — количество архивов в группе
- corrupted_file_data_list — список файлов:
    - id — уникальный ID архива
    - source_file_name — исходное имя файла (например, archive_20251114_012345.zip)

Общая статистика очереди обработки архивов

Чтобы отличить проблемы с corrupted от общей нагрузки на систему, выполните запрос:

```
{
 monitoring_diagnostics {
  agent_file_query {
   agent_file_queue_statistic {
    all_queue_size
    wait_processing_size
    processed_size
   }
  }
 }
}
```

Где:

- all_queue_size — всего архивов в очереди
- wait_processing_size — архивов, ожидающих обработки
- processed_size — успешно обработанных и загруженных в ClickHouse

Выгрузка архива

Для выгрузки архива из очереди corrupted необходимы идентификаторы, полученные из запроса на получение списка поврежденных архивов:

- id архива
- node_id сервера

Выполните запрос:

```
{
 monitoring_diagnostics {
  corrupted_file_query {
   corrupted_file_by_id(id: $id, runtimeNodeId: "$node_id")
  }
 }
}
```

После выполнения запроса:

1. Удалите букву i в адресной строке браузера из graphiql.

2. Нажмите **Enter**.

3. Начнется автоматическая загрузка архива в браузере. Файл скачается с именем corrupted_<id>.zip.

Повторная обработка поврежденных архивов

Если вы уверены, что архивы попали в corrupted ошибочно, можно запустить массовую повторную обработку.

Важно. Перед выполнением запроса убедитесь, что в очереди corrupted находится не менее 100 000 архивов.

```
mutation {
  computer_activity {
    reload_corrupted_activities(is_delete_on_fail: true)
  }
}
```

Где у параметра is_delete_on_fail может быть установлено два значения:

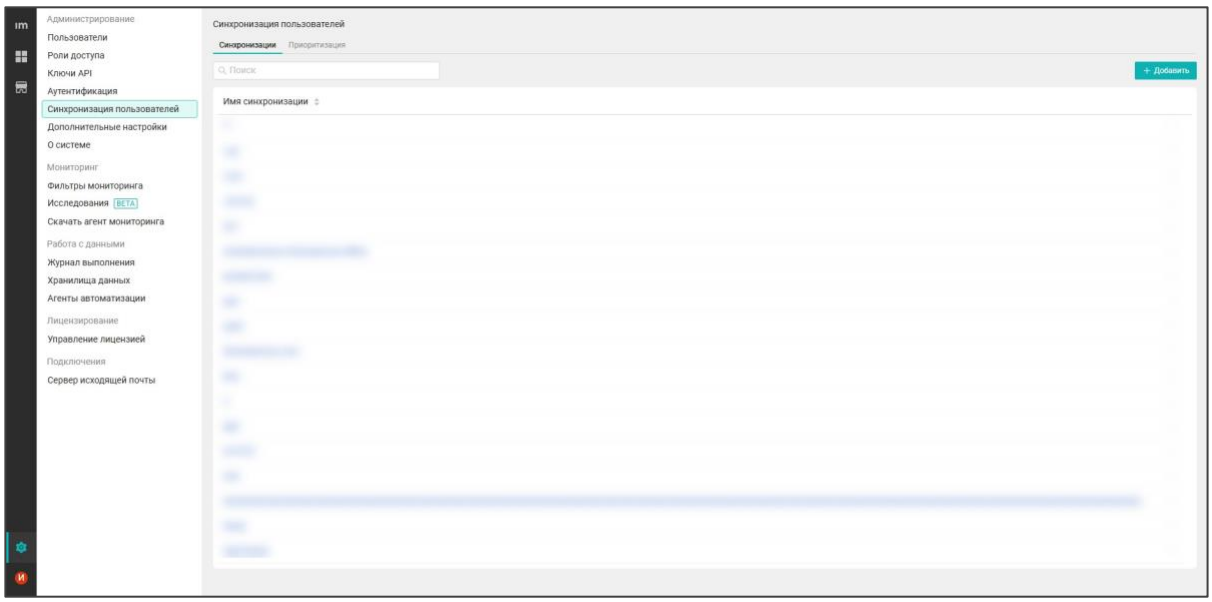
- true — если архив не удается разобрать, то он безвозвратно удаляется
- false — если архив не удается разобрать, то он возвращается в очередь corrupted

Синхронизация пользователей с Active Directory

Настройки синхронизации пользователей с Active Directory в веб-интерфейсе представлены в разделе *Настройки/Администрирование/Синхронизация пользователей*.

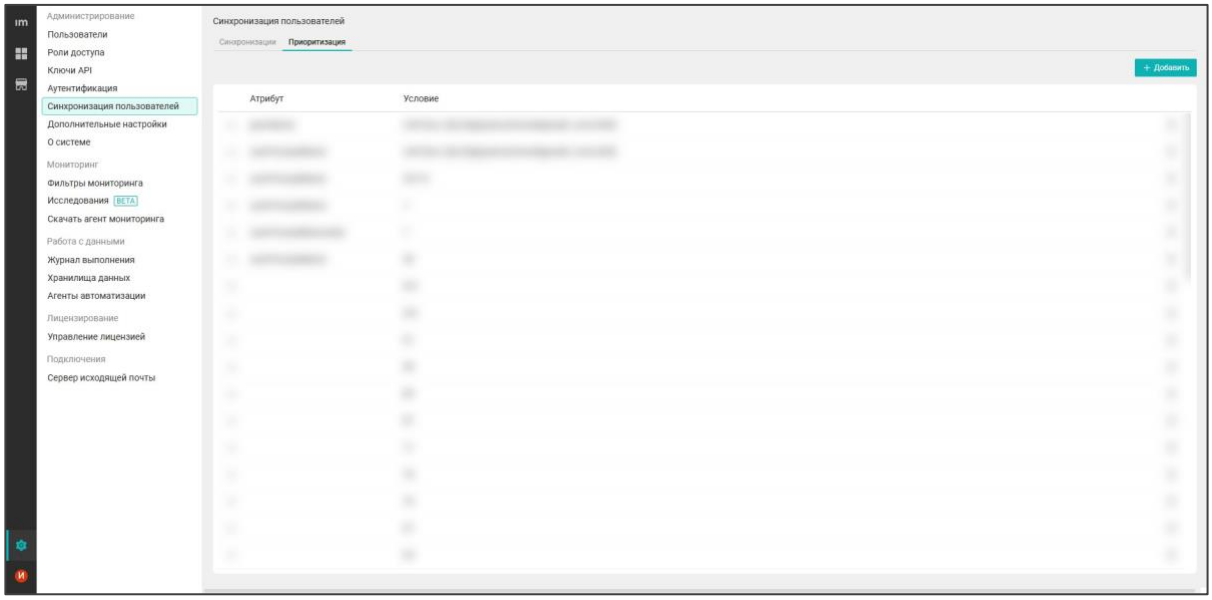
Раздел по умолчанию не содержит настроенных синхронизаций.

При наличии настроенных синхронизаций они отображаются во вкладке *Синхронизации* в следующем порядке: знаки препинания (например, кавычки или восклицательный знак), 0-9, A-Z, a-z, А-Я, а-я. Пользователь может сортировать значения по названию синхронизации.

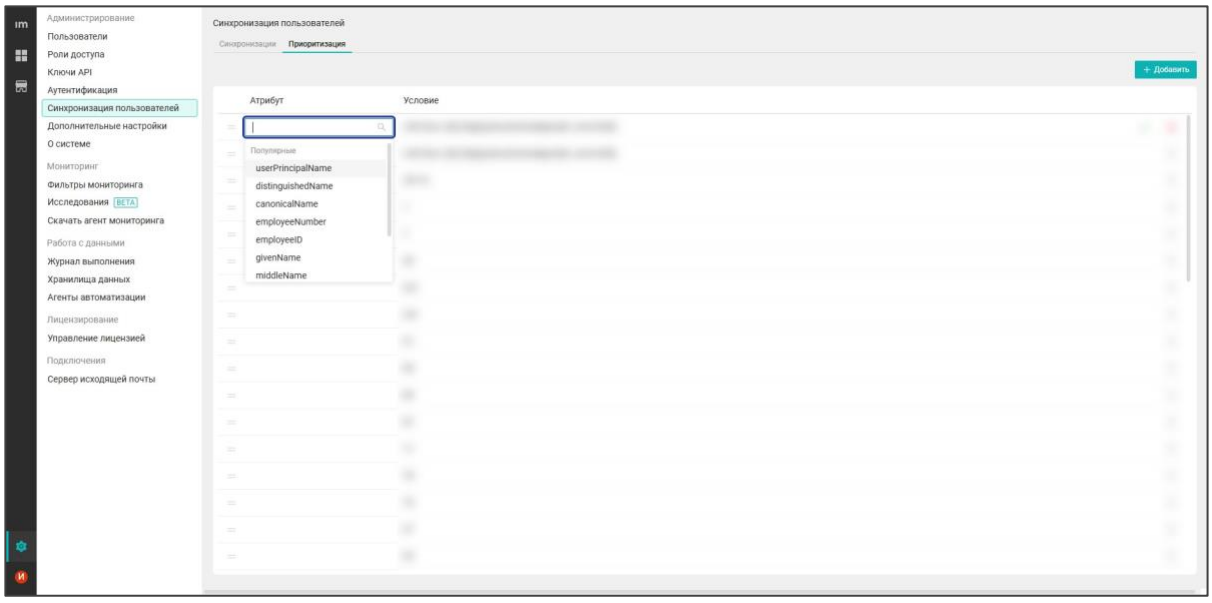


Примечание. Если к одному пользователю привязано несколько учетных записей Active Directory, данные пользователя синхронизируются только из источника с высшим приоритетом.

Если источников синхронизации несколько, возможно задать их приоритетность. Для этого во вкладке *Приоритизация* необходимо добавить условия в формате регулярного выражения. Порядок регулируется перетаскиванием условий: чем выражение выше, тем приоритетнее его выполнение.



Вы также можете установить приоритет по желаемому атрибуту. Выберите его из списка или введите вручную в соответствующем поле. Это поле необязательно для заполнения, не участвует в синхронизации полей пользователей и не отображается в их профилях.



Вкладка «Основное»

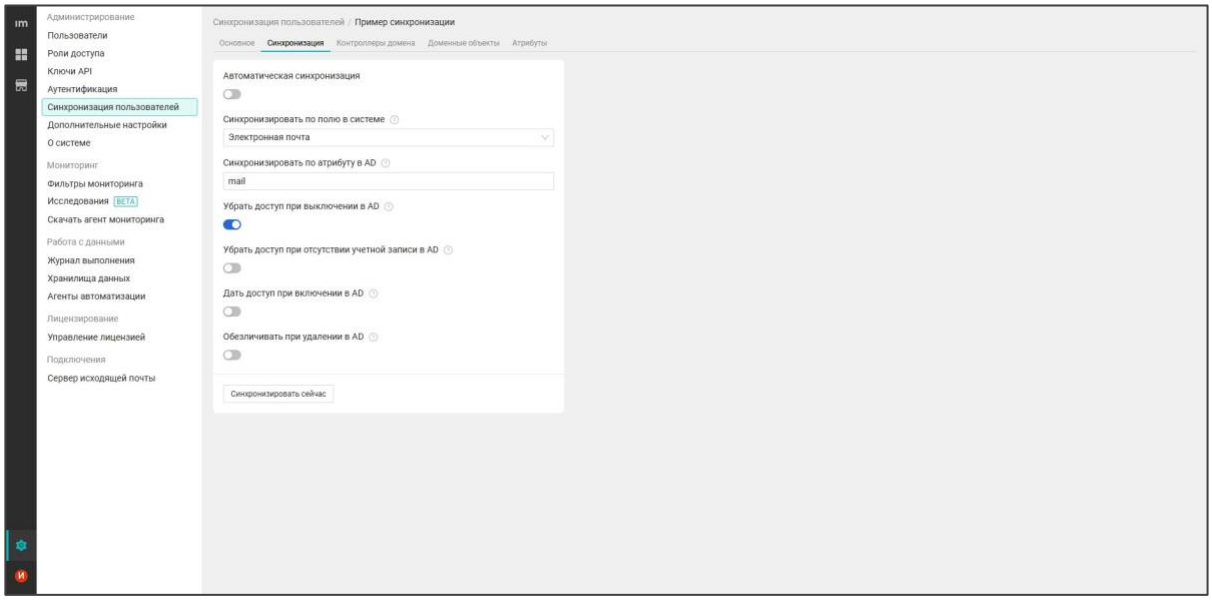
Во вкладке *Основное* можно изменить название синхронизации.

Вкладка «Синхронизация»

На странице представлены следующие параметры:

- Автоматическая синхронизация (включение/отключение периодической синхронизации данных из AD. Периодичность — 15 минут)
- Синхронизировать по полю в системе
- Синхронизировать по атрибуту в AD
- Убрать доступ при выключении в AD (позволяет автоматически выключать пользователя в системе)

- Убрать доступ при отсутствии учетной записи в AD (отключает пользователя из системы, если учетная запись удалена из AD)
- Дать доступ при включении в AD (позволяет автоматически включать пользователя при включении его в AD. Данная опция работает только при включенном параметре Убрать доступ при выключении в AD)
- Обезличивать при удалении в AD (включение данной опции позволяет обезличить данные пользователей, учетные записи которых были удалены из Active Directory)
- Время синхронизации (скрыто, если синхронизация данных из AD не осуществлялась)
- Синхронизировать сейчас (отправка запроса на моментальную синхронизацию данных)



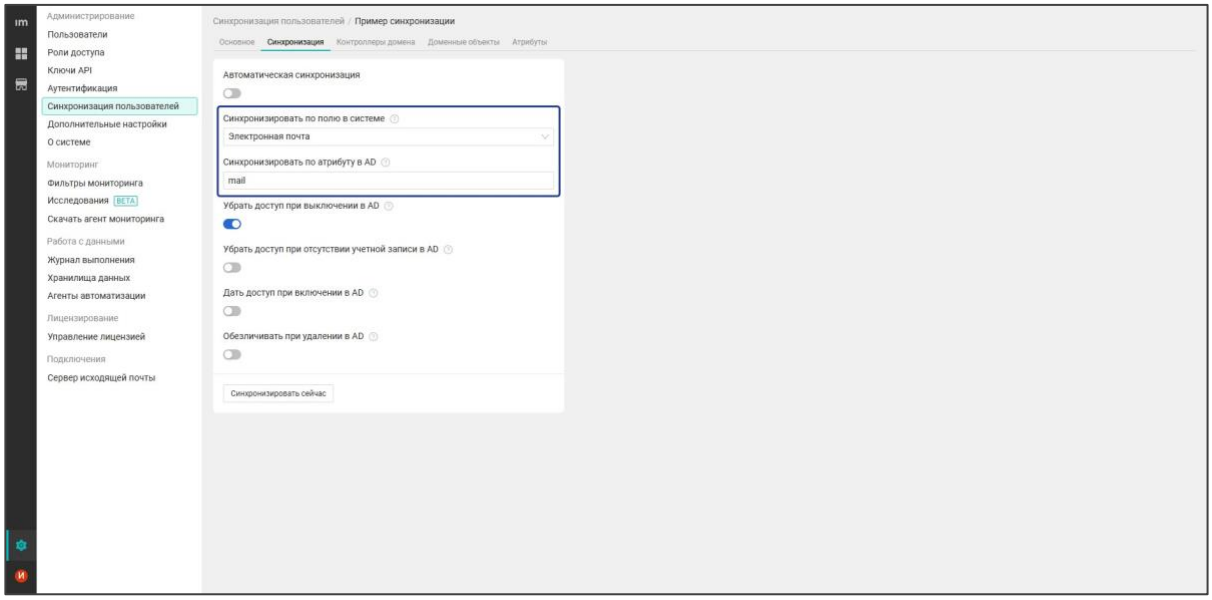
Время автоматической синхронизации с AD можно изменить через GraphQL-запрос:

```
mutation {
  app_config {
    active_directory {
      update_config(sync_period: 65000) {
        sync_period
      }
    }
  }
}
```

sync_period — период синхронизации, указывается в миллисекундах.

Первичное сопоставление атрибута

В профиле синхронизации доступна настройка первичного сопоставления атрибута (поля **Синхронизировать по полю** и **Синхронизировать по атрибуту в AD**). Первичное сопоставление атрибутов позволяет решить проблему, когда пользователь может синхронизироваться в систему из других систем.

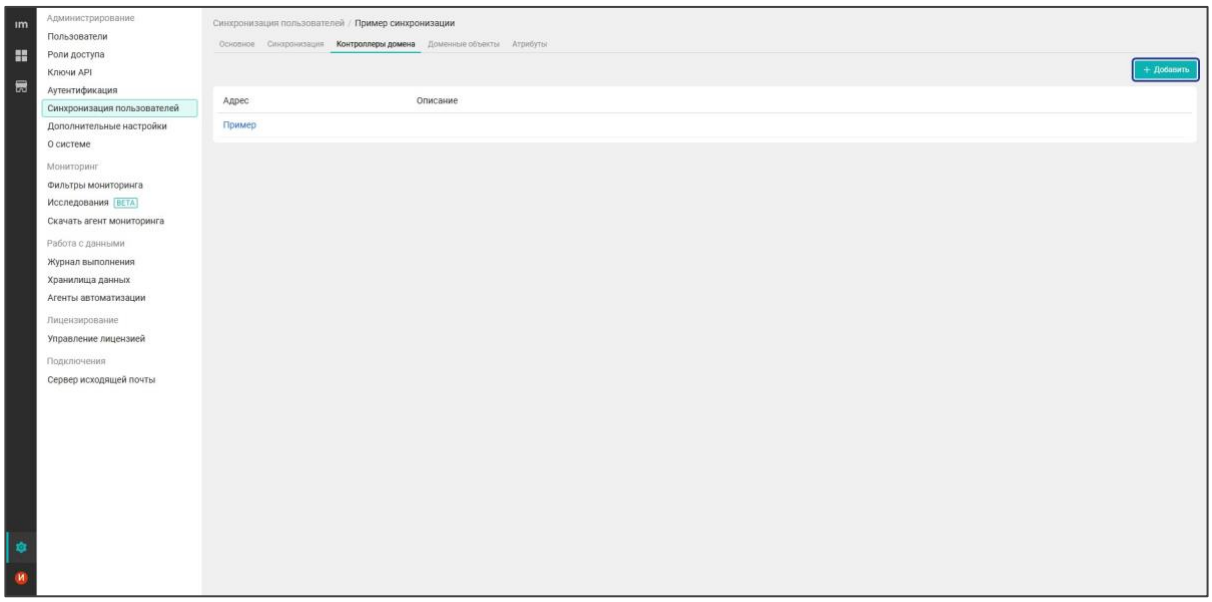


При первичной синхронизации значение атрибута из AD сопоставляется со значением поля в профиле пользователя. Если значение атрибута из AD совпадает со значением в поле пользователя, то новый пользователь при синхронизации из AD не создается, а синхронизируется в уже созданного.

Пример: Пользователь синхронизировался из «Босс-кадровика», где в поле *Табельный номер* в профиле пользователя указано значение 1001. Функционалом первичной синхронизации сопоставляется значение атрибута из AD со значением поля в профиле пользователя. Если значения совпадают, в данном примере равны 1001, то новый пользователь не создаётся.

Вкладка «Контроллеры домена»

Для добавления контроллера домена нажмите кнопку **+ Добавить**.



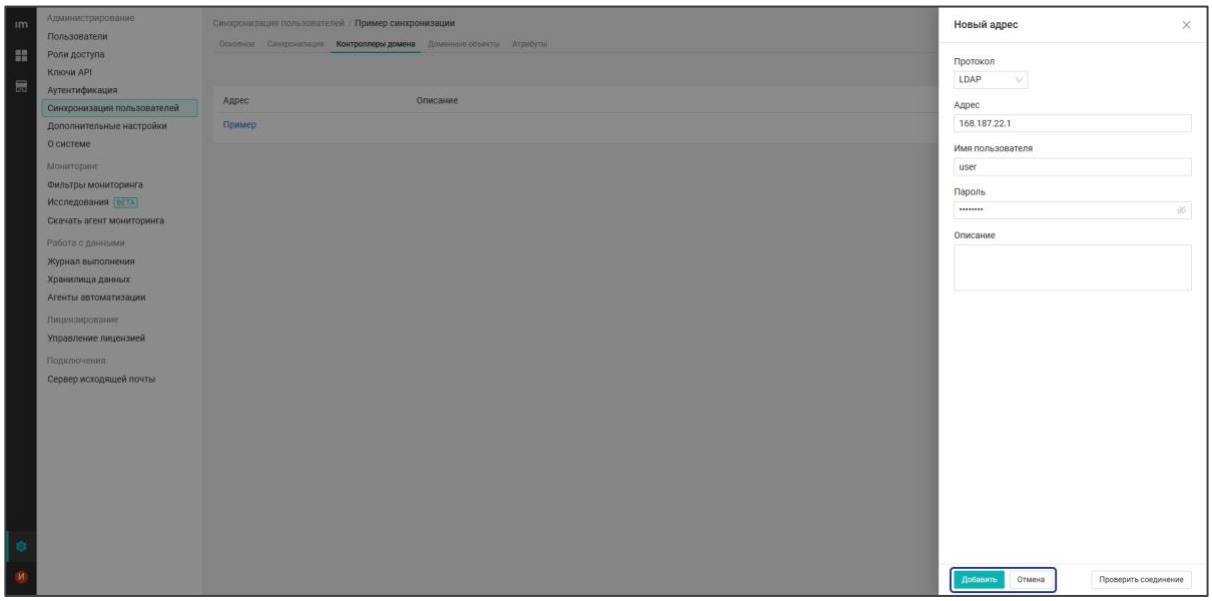
Справа открывается панель *Новый адрес*, где отображаются параметры:

- Протокол: при выборе LDAPS в интерфейсе появляется пункт Сертификат
- Сертификат:

- Интеграция системы с AD поддерживает работу с хранилищем сертификатов ОС Windows. При наличии необходимых сертификатов в хранилище компьютера, где установлена система, загрузка файла сертификата на странице настройки не потребуется
- В ином случае загрузите на странице файл сертификата промежуточного или корневого удостоверяющего центра (CA), которым подписаны сертификаты, используемые контроллерами домена для LDAPS

- Адрес (домен контроллера или имя домена)
- Имя пользователя (логин AD с правами на чтение каталога)
- Пароль (если пароль ранее не был задан)
- Кнопка Изменить пароль (если пароль ранее уже был задан)
- Описание
- Кнопка Проверить соединение (выполняется пробное подключение к домену после сохранения изменений)

Нажмите кнопку **Добавить** (при добавлении нового контроллера) или **Сохранить** (при редактировании существующего) для подтверждения. Для отмены нажмите кнопку **Отменить**.



Вкладка «Доменные объекты»

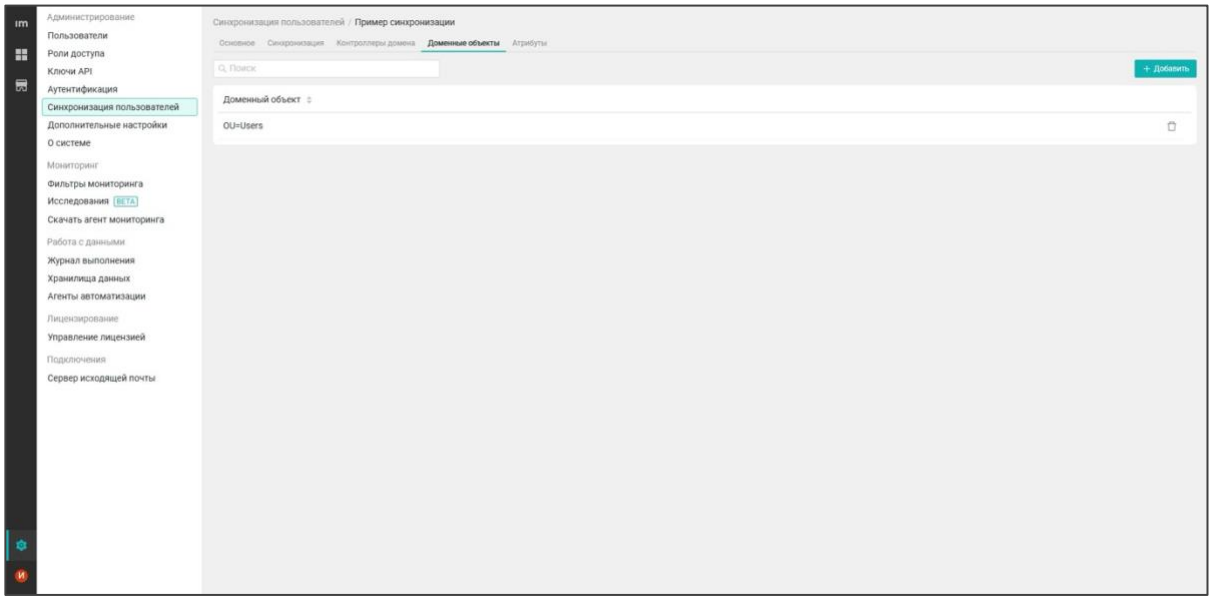
Во вкладке *Доменные объекты* можно добавить объекты, которые будут синхронизироваться из AD в систему. Они могут содержать в себе несколько отделов или конечных пользователей.

Важно. При настройке синхронизации доменных объектов возможно использование подразделений (**OU**) или универсальных групп безопасности (**Universal Security Groups**). Глобальные группы безопасности (**Global Security Groups**) нельзя добавить в список синхронизируемых объектов.

Доменный объект вводится в формате Distinguished Name (пример: OU=UnitName,OU=Users,OU=Root,DC=some,DC=domain,DC=com). Чтобы использовать Universal Security Group, включите их синхронизацию. Введите GraphQL-запрос:

```
mutation {
  app_config {
    active_directory {
      update_config(sync_employee_group_membership_enabled: true,
sync_nested_group_membership_enabled: true) {
        sync_employee_group_member_enabled
        sync_nested_group_membership_enabled
      }
    }
  }
}
```

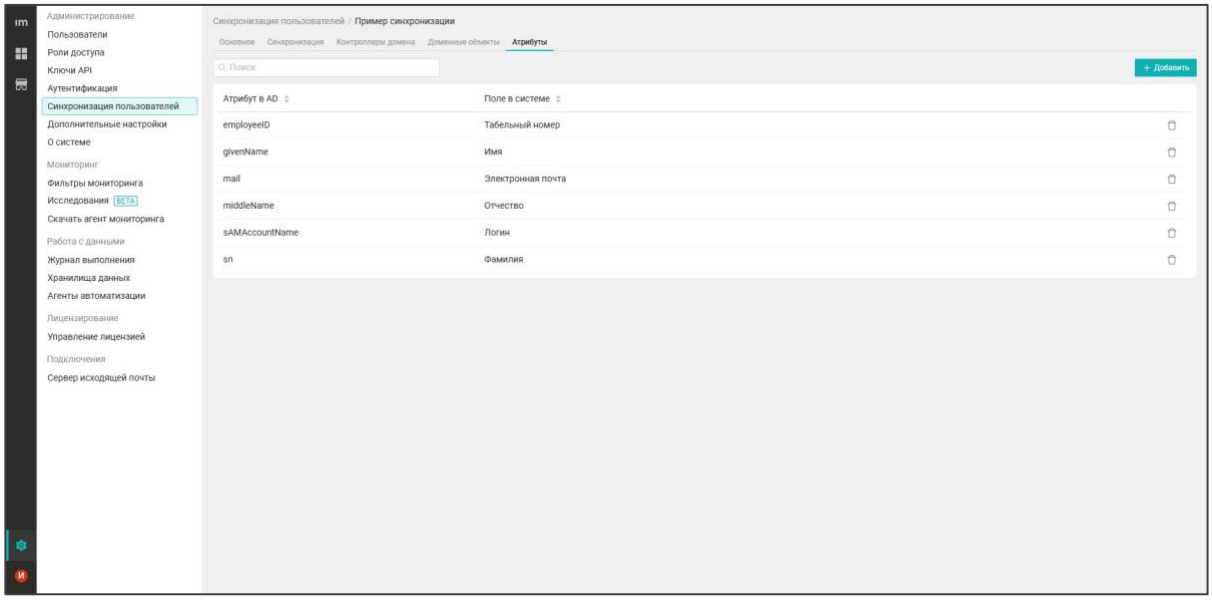
Запрос синхронизации осуществляется в том случае, если присутствуют синхронизируемые объекты.



Вкладка «Атрибуты»

Во вкладке *Атрибуты* предустановлен список базовых атрибутов пользователей, которые будут синхронизироваться из AD в систему. Представлены следующие параметры:

- Атрибут в AD
- Поле в системе



Добавление нового атрибута

Можно добавить и настроить собственные атрибуты, которые будут синхронизироваться в систему. При добавлении нового атрибута необходимо ввести следующие данные:

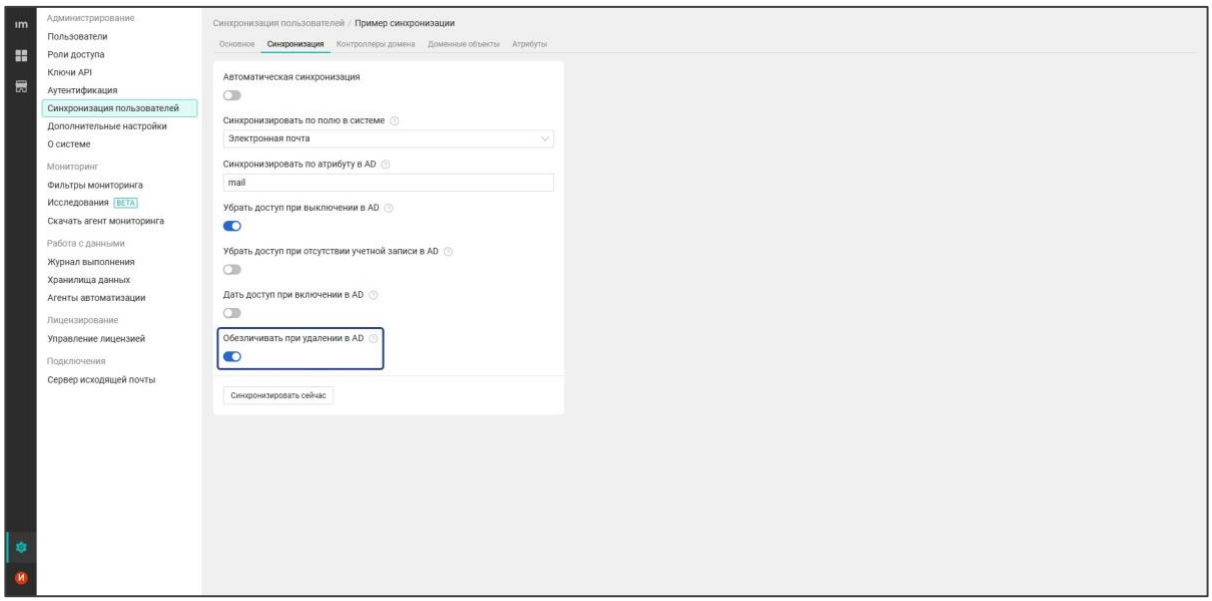
- Атрибут в AD — наименование нового атрибута для синхронизации
- Поле в системе

Настройка обезличивания данных пользователей при удалении в AD

Чтобы обезличить данные пользователей, учетные записи которых удалены из Active Directory, нужно зайти в настройки обезличивания в веб-интерфейсе системы на странице *Настройки/Администрирование/Синхронизация пользователей*, вкладка *Синхронизация*.

Обезличиваются данные всех пользователей, которые были ранее синхронизированы из Active Directory, а теперь отсутствуют в синхронизируемых объектах.

Если у сотрудника было включено обезличивание, то в случае удаления учетной записи из Active Directory в системе у пользователя разрывается связь с Active Directory, выключается доступ, обезличиваются данные. Возможно редактировать профиль сотрудника и использовать учетную запись для локального входа в систему.



Обезличивание происходит посредством замены пользовательских данных на уникальный порядковый номер сотрудника в системе. В профиле сотрудника с обезличенными персональными данными поля, ранее заполненные данными, проставляется значение [employee_id] Без имени.

Активация LDAPS

Протокол LDAP

Протокол LDAP используется для чтения и записи данных в Active Directory. По умолчанию при установке синхронизации трафик LDAP не защищен. Чтобы активировать LDAP через SSL (LDAPS), установите сертификат центра сертификации Майкрософт (CA) или сертификат независимого центра сертификации.

Для установки соединения LDAPS выполните предварительную настройку.

Требования к сертификату для активации LDAPS

Для активации LDAPS установите сертификат, соответствующий следующим требованиям:

1. Закрытый ключ, подходящий для сертификата, находится в хранилище локального компьютера и соответствует данному сертификату. Усиленная защита закрытого ключа не должна быть активирована.

2. Расширение улучшенного ключа включает идентификатор объекта (также известный как OID) проверки подлинности сервера (1.3.6.1.5.5.7.3.1).

3. Полное доменное имя контроллера Active Directory (например, DC01.DOMAIN.COM) должно отображаться в одном из следующих мест:

- «Общее имя» (CN) в поле «Тема»
- DNS-запись в расширении «Дополнительное имя субъекта»

4. Сертификат был выдан центром сертификации, которому доверяют и контроллер домена, и клиенты LDAPS. Отношения устанавливаются через настройку доверия клиентских компьютеров и сервера корневому центру сертификации, к которому привязывается выпускающий центр сертификации.

Сертификат сохраняется на локальную машину и загружается в базу данных системы. Загрузка сертификата осуществляется через веб-интерфейс в параметры «Сертификат» на странице синхронизации или с помощью GraphQL-запроса.

Синхронизируемые атрибуты между системой и AD

Интеграция системы и AD позволяет выполнять настройку атрибутов, которые необходимо синхронизировать. Список всех атрибутов представлен в таблице. Для синхронизации доступны также любые произвольные атрибуты.

Атрибут	Значение	Тип
sAMAccountName	Имя учетной записи пользователя (pre-Windows 2000)	Базовый
userAccountControl	Статус учетной записи	Базовый
employeeID	Табельный номер	Базовый
sn	Фамилия	Базовый
givenName	Имя	Базовый
middleName	Отчество	Базовый
mail	Электронная почта	Базовый
objectGUID	Уникальный идентификатор	Обязательный
userPrincipalName	Имя учетной записи пользователя	Обязательный
distinguishedName	Отличительное имя	Обязательный
objectClass	Позволяет задать тип (класс) искомого объекта, может принимать несколько значений	Обязательный
objectCategory	Позволяет задать тип (класс) искомого объекта, имеет только одно значение	Обязательный

Базовые синхронизируемые атрибуты — это атрибуты, которые могут быть отключены для синхронизации, при этом интеграция будет находиться в стабильном состоянии. Если отключены все базовые атрибуты, то синхронизация не выполняется.

После запуска подсистемы начинается проверка схемы базовых атрибутов в базе данных.

Обязательные синхронизируемые атрибуты — это атрибуты, которые при отключении могут привести к нестабильному состоянию интеграции. Они не хранятся в базе данных, их нельзя отключить. Обязательные синхронизируемые атрибуты всегда запрашиваются вместе с базовыми атрибутами у Active Directory. Настройка атрибутов происходит в веб-интерфейсе.

Получение информации о группах безопасности AD с помощью GraphQL-запроса

Вы можете получить информацию о группах безопасности AD с помощью GraphQL-запроса в GraphiQL.

Важно. Перед тем, как выполнить GraphQL-запрос на получение информации о группах безопасности, необходимо настроить интеграцию AD на синхронизацию информации по группе.

Запрос для групп безопасности AD можно выполнить только от имени ключа API с назначенной привилегией «Пользователи и отделы» с операцией доступа **R** (чтение). Подробную информацию о том, как создать ключ API и назначить привилегии созданному ключу, вы можете найти в следующих разделах:

- Описание способов аутентификации и способов работы с GraphiQL
- Права доступа для Ключей API

Настройте интеграцию AD с помощью следующего запроса:

```
mutation {
  app_config {
    active_directory {
      update_config(sync_employee_group_membership_enabled:true) {
        sync_employee_group_member_enabled
      }
    }
  }
}
```

Примечание. Синхронизация осуществляется по универсальной группе безопасности. Глобальная группа безопасности не поддерживается.

Варианты запроса для групп безопасности:

1. Запрос без фильтра:

```
{
  active_directory {
    ad_group {
      employee_ad_groups {
        employee_id
        canonical_names
        distinguished_names
        names
      }
    }
  }
}
```

2. Запрос с фильтром по пользователям:

```
{
  active_directory {
```

```
  ad_group {
   employee_ad_groups(ids:[<id_пользователя_1>, <id_пользователя_2>]) {
    employee_id
    canonical_names
    distinguished_names
    names
   }
  }
 }
}
```

Пример ответа:

```
{
 "data": {
  "active_directory": {
   "ad_group": {
    "employee_ad_groups": [
     {
      "employee_id": 11,
      "canonical_names": [
       "dmitry.local/root/users/Users"
      ],
      "distinguished_names": [
       "CN=Users,OU=users,OU=root,DC=dmitry,DC=local"
      ],
      "names": [
       "Users"
      ]
     }
    ]
   }
  }
 }
}
```

Поле	Определение
employee_ad_groups	Пользователь с группами безопасности AD
employee_id	Идентификатор пользователя
canonical_names	Имя пользователя в каноническом формате
distinguished_names	Уникальное имя пользователя
names	Название группы безопасности

Запрос также можно выполнить через блок автоматизации «HTTP-запрос». Подробная информация о работе этого блока представлена в разделе Работа с сервисами.

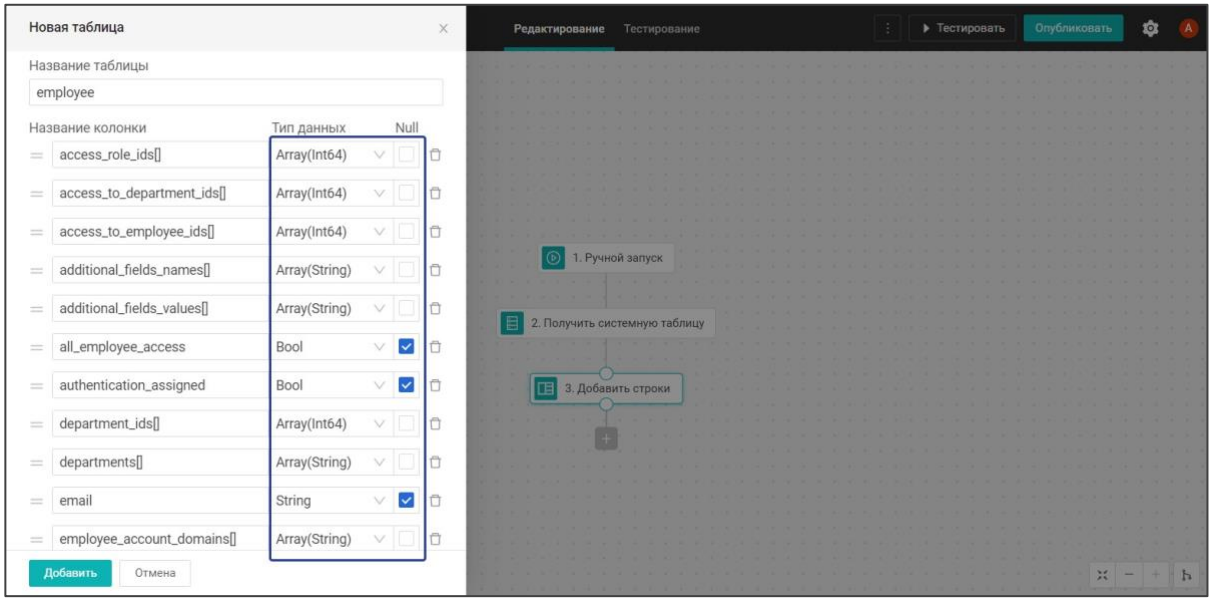
Системные таблицы

Система предоставляет возможность получать актуальную информацию о пользователях, правах доступа, дашбордах, скриптах и других элементах. Необходимые данные можно извлечь и передать в ClickHouse с помощью блока автоматизации Получить системную таблицу.

Ниже представлено описание системных таблиц, а также их полей, которые могут быть получены с использованием этого блока.

Важно.

- При создании новой таблицы из выходных данных блока **Получить системную таблицу** рекомендуется указывать в колонке **Тип данных** соответствующий тип ClickHouse.
- Выбор подходящего типа данных для числовых значений поля id зависит от размера добавляемой таблицы. В большинстве случаев предпочтительными являются типы *Int64*/*UInt64*, но вы можете выбрать нужный тип из следующих вариантов: *UInt32*, *UInt64*, *UInt128*, *UInt256*, *Int64*, *Int128* или *Int256*. Рекомендуется указать тот тип данных, который соответствует вашим требованиям. Ознакомиться с описанием этих типов вы можете на официальном сайте ClickHouse.
- Тип данных ClickHouse *Array(Tun)* не может быть Nullable. При вводе типа колонки *Array(Tun)*, снимите галочку у опции **Null** в соответствующих полях.



Связи системных таблиц

Представленная ниже схема поможет понять, каким образом системные таблицы Proceset связаны между собой.

На схеме отображены:

- Все основные системные таблицы Proceset
- Первичные и внешние ключи системных таблиц
- Связи между таблицами

Таблица «access_role»

В таблицу собирается информация о ролях доступа.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор роли доступа
is_admin	Bool	Роль «Прикладной администратор»
name	String	Имя роли доступа

Таблица «access_role_privilege»

В таблицу собирается информация о привилегиях ролей доступа.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор привилегии
access_role_id	Integer	Идентификатор роли доступа
is_create	Bool	Операция Create
is_delete	Bool	Операция Delete
is_execute	Bool	Операция Execute
is_read	Bool	Операция Read
is_write	Bool	Операция Write
name_ru	String	Имя привилегии
name	String	Код привилегии с указанием модуля

Таблица «ad_attribute»

В таблицу собираются сведения о синхронизируемых атрибутах.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор значения атрибута
additional_field_name	String	Название дополнительного поля
attribute_name	String	Название атрибута
data_type	String	Тип данных
employee_id	Integer	Идентификатор пользователя, к которому относится значение атрибута

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
index	Integer	Индекс, если атрибут хранит массив значений
is_long_value_null	Bool	Пустое или числовое значение
is_string_value_null	Bool	Пустое или строковое значение
long_value	Integer	Числовое значение
string_value	String	Строковое значение

Таблица «dashboard»

В таблицу собирается информация о дашбордах.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
dashboard_author_id	Integer	Идентификатор автора дашборда
dashboard_creation_time	DateTime DateTime64	Время создания дашборда
dashboard_name	String	Наименование дашборда
guid	String	guid дашборда
workspace_id	Integer	Идентификатор пространства

Таблица «dashboard_access»

В таблицу собирается информация о доступах к отчетам.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
dashboard_id	Integer	Идентификатор дашборда
employee_id	Integer	Идентификатор пользователя
operation	String	Операции доступа (Read/Write)

Таблица «department»

В таблицу собирается информация об отделах.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор отдела
name	String	Название отдела
parent_department_id	Integer	Идентификатор родительского отдела
parent_department_ids	Array(Int64)	Массив идентификаторов родительских отделов

Таблица «employee»

В таблицу собирается информация о пользователях.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор пользователя (также ключ сортировки)
access_role_ids	Array(Int64)	Массив идентификаторов ролей доступа пользователя
access_to_department_ids	Array(Int64)	Идентификаторы отделов, к которым у данного пользователя есть доступ

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
access_to_employee_ids	Array(Int64)	Массив идентификаторов сотрудников, к которым у данного пользователя есть доступ
additional_fields_names	Array(String)	Массив имен дополнительных полей пользователя
additional_fields_values	Array(String)	Массив значений дополнительных полей пользователя
all_employee_access	Bool	Наличие доступа ко всем пользователям
authentication_assigned	Bool	Проверка возможности аутентификации у пользователя
authentication_types_assigned	Array(String)	Массив аутентификаций пользователя
department_ids	Array(Int64)	Массив идентификаторов отделов (корневой, промежуточный, конкретный отдел)
departments	Array(String)	Массив наименований отделов (корневой, промежуточный, конкретный отдел)
email	String	Электронная почта
employee_account_domains	Array(String)	Список доменов всех аккаунтов пользователя (пустая строка, если нет)
employee_account_ids	Array(Int64)	Массив идентификаторов источников активности
employee_account_logins	Array(String)	Список логинов всех аккаунтов пользователя (пустая строка, если нет)
first_name	String	Имя
license_roles	Array(String)	Лицензии, назначенные пользователю
login	String	Логин
monitoring_type	String	Сбор активности для конкретного пользователя (DISABLED/SIMPLE/EXTENDED)
name	String	Полное отображаемое имя
patronymic	String	Отчество
personnel_number	String	Табельный номер
phones	Array(String)	Телефон пользователя
second_name	String	Фамилия

Таблица «employee_account»

В таблицу собирается информация об аккаунтах пользователей.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор аккаунта
employee_id	Integer	Идентификатор пользователя

Таблица «employee_ad_group»

В таблицу собирается информация о группах пользователей в Active Directory.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
ad_account_distinguished_name	String	Путь до аккаунта AD
ad_account_guid	String	guid аккаунта AD
ad_account_id	Integer	Идентификатор аккаунта AD
ad_group_canonical_name	String	Название группы AD в каноническом формате
ad_group_distinguished_name	String	Путь до группы AD
ad_group_guid	String	guid группы AD

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
ad_group_id	Integer	Идентификатор группы AD
ad_group_name	String	Название группы AD
display_name	String	Отображаемое имя пользователя в системе
employee_id	Integer	Идентификатор пользователя

Таблица «employee_favourite_workspace»

В таблицу собирается информация о пространствах, добавленных в раздел *Избранное.*

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
employee_id	Integer	Идентификатор сотрудника
workspace_id	Integer	Идентификатор пространства

Таблица «employee_workspace_access»

В таблицу собирается информация о доступах пользователя к дашбордам в пространстве.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
employee_id	Integer	Идентификатор пользователя
operation	String	Операция доступа
workspace_id	Integer	Идентификатор пространства

Таблица «employee_workspace_main_page_group»

В таблицу собирается информация о пространствах, добавленных в закладки.

Важно. Запись в эту таблицу создается, только если в закладке есть хотя бы одно пространство. Пустые закладки не отображаются в системной таблице.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
employee_id	Integer	Идентификатор сотрудника
main_page_group_id	Integer	Идентификатор закладки
main_page_group_name	String	Имя закладки
workspace_id	Integer	Идентификатор пространства

Таблица «employee_license_role_log»

В таблицу собирается информация об изменении лицензионных ролей пользователей.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
api_key_description	String	Описание источника изменения (формат ID~message, где ID — идентификатор API-ключа, message — часть API-ключа) Заполняется, если изменение выполнено через API-ключ. Если изменение выполнено не через API-ключ — поле пустое

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
employee_description	String	Описание источника изменения (формат ID~message, где ID — идентификатор пользователя, message — display_name) Заполняется, если изменение выполнено сотрудником. Если изменение выполнено не сотрудником — поле пустое
employee_id	Integer	Идентификатор пользователя
event_date	DateTime DateTime64	Дата и время события изменения роли
license_role	String	Лицензионная роль
operation	String	Операция добавления или удаления лицензионной роли Возможные значения: - ADD - REMOVE
source_type	String	Тип инициатора изменения Возможные значения: - EMPLOYEE - API_KEY - SYSTEM

Таблица «link_workspace_employee»

В таблицу собирается информация о доступах пользователя в пространстве.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор объекта доступа
employee_id	Integer	Идентификатор пользователя
operation	String	Тип операции (Read/Write)
workspace_id	Integer	Идентификатор рабочего пространства

Таблица «monitoring_screenshot»

В таблицу собирается информация об исследованиях.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор исследования
blur	Bool	Размытие
employees	Array(Int64)	Сотрудники, которые добавлены в исследование
end	DateTime	Дата завершения исследования
name	String	Название исследования
start	DateTime	Дата начала исследования
status	String	Статус исследования

Таблица «monitoring_employee_log_type»

В таблицу собирается информация о типе мониторинга пользователя, а также время и источник его изменения.

События, при которых происходит запись переключения значения мониторинга:

- При автоматическом создании нового пользователя в системе (интеграция из AD, поступление данных от агента) источником изменений будет SYSTEM. Значения будут установлены в DISABLED
- В случаях изменения типа мониторинга пользователем, администратором или через API будет указан источник изменения EMPLOYEE или API_KEY, прежнее значение до изменения и новое установленное значение

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор доменного объекта
api_description	String	Описание источника изменения (формат ID~message, где ID — идентификатор api_key, message — api*****key)
employee_description	String	Описание источника изменения (формат ID~message, где ID — идентификатор пользователя, message — display_name)
employee_id	Integer	Идентификатор пользователя, которому изменили тип мониторинга
event_date	DateTime DateTime64	Дата переключения типов мониторинга
event_source	String	Источник изменения типа мониторинга (SYSTEM, EMPLOYEE, API_KEY)
new_value	String	Новое значение типа мониторинга
old_value	String	Прежнее значение типа мониторинга

Таблица «resource_monitor»

В таблицу собираются данные о текущем состоянии системы. Таблица не содержит исторических записей и отражает только информацию, доступную в памяти на момент запроса.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор монитора ресурсов (timestamp)
cpu	Float32	Нагрузка на ЦПУ в момент запроса в процентах
disk	Float32	Используемое пространство на диске в момент запроса (в Мб)
node_id	String	Идентификатор ноды в кластере
node_name	String	Имя ноды в кластере
ram	Float32	Используемая оперативная память в момент запроса (в Мб)

Таблица «script_event_history»

В таблицу собирается информация о действиях пользователей со скриптами.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
api_key_id	Integer	Идентификатор ключа API, который запускает скрипт
author_event_name	String	ФИО сотрудника или имя ключа API, который запускает скрипт

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
employee_id	Integer	Идентификатор пользователя
event_time	DateTime DateTime64	Дата события
event_type	String	Тип события: - Создание - Удаление - Публикация - Восстановление старой версии
script_general_id	Integer	Идентификатор скрипта
script_version	Integer	Версия скрипта

Таблица «script_execution»

В таблицу собираются данные о выполнении скриптов за последние 24 часа.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор выполнения скрипта
duration	Integer	Длительность выполнения скрипта в миллисекундах
error	String	Ошибка
execution_status	String	Статус выполнения скрипта
node_name	String	Имя агента автоматизации, на котором был выполнен скрипт
script_id	Integer	Идентификатор скрипта
script_name	String	Имя скрипта
script_version	Integer	Версия скрипта
start_time	DateTime	Время начала выполнения скрипта
workspace_id	Integer	Идентификатор пространства
workspace_name	String	Имя пространства

Таблица «system_event»

В таблицу собираются данные о текущих системных событиях. Таблица не содержит исторических записей и отображает только события, находящиеся в памяти в момент запроса.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
event_type	String	Тип события, определяется соответствующим монитором
level	Integer	Уровень оповещения. Соответствие значений в таблице: - 1 — CRITICAL - 2 — ERROR - 3 — WARNING - 4 — INFO
message	String	Сообщение, генерируемое монитором, как реакция на наступление события
subsystem_uuid	String	Уникальный идентификатор модуля
time	DateTime DateTime64	Время, когда произошло событие
ttl	Integer	Период отображения события в миллисекундах

Таблица «tag»

В таблицу собирается информация о тегах.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
colour	String	Цвет тега
read_only	Bool	Параметр, отвечающий за возможность редактирования тега
name	String	Название тега

Таблица «workspace»

В таблицу собирается информация о пространствах.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор пространства
description	String	Описание
folder_id	Integer	Идентификатор папки, в которой находится пространство
is_in_waste_bin	Bool	Индикатор наличия пространства в корзине
name	String	Имя пространства

Таблица «workspace_database»

В таблицу собирается информация о базах данных пространств.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
database_name	String	Имя базы данных
storage_guid	String	Идентификатор хранилища данных
workspace_id	Integer	Идентификатор пространства

Таблица «workspace_folder»

В таблицу собирается информация о папках WorkspaceFolder.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор папки
name	String	Имя папки
parent_id	Integer	Идентификатор родительской папки

Таблица «workspace_tag»

В таблицу собирается информация о тегах и пространствах, которым они назначены.

Поле в поставщике данных	Рекомендуемый тип ClickHouse	Описание
id	Integer	Идентификатор записи
tag_id	Integer	Идентификатор тега
workspace_id	Integer	Идентификатор пространства