Руководство администратора системы Infozorok

ООО «ИБ-ИННОВАЦИИ»
2025

Содержание

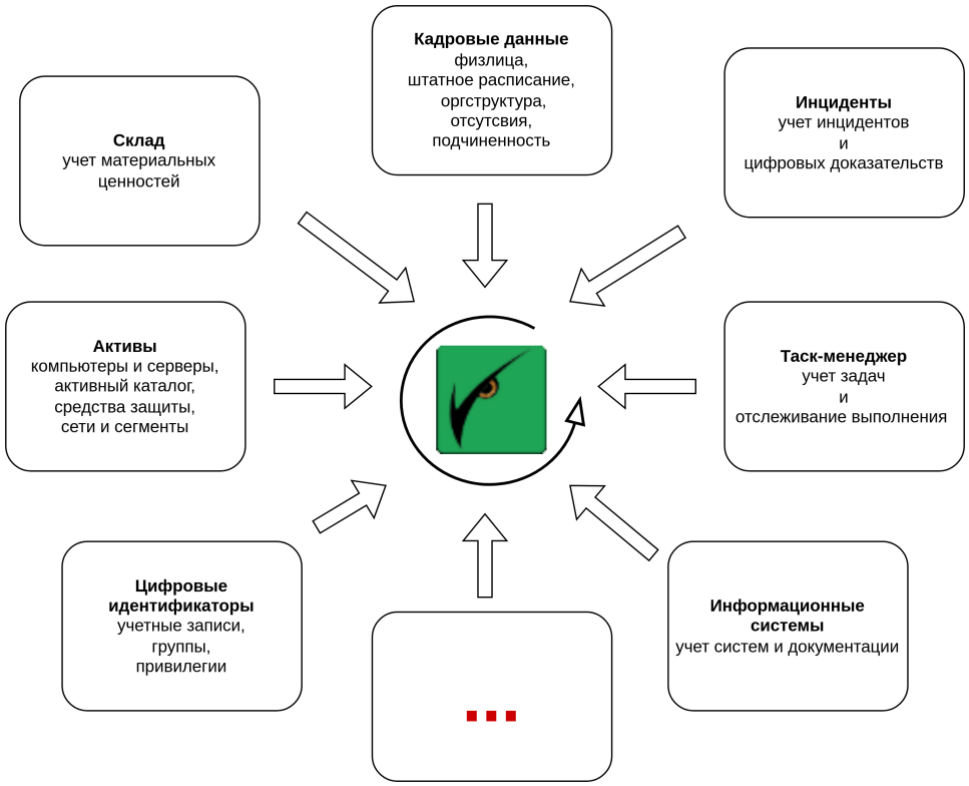
Перечень условных обозначений и сокращений 3
1.Возможности системы 4
2.Установка системы 5
2.1.Системные требования 5
2.2.Подготовка 5
2.3.Генерация ключей шифрования 7
2.4.Установка сервера 7
2.5.Установка агента 7
2.6.Активация агента 8
3.Администрирование системы 9
3.1.Общие принципы 9
3.2.Настройка интеграций 12
3.3.Настройка алертов 16
3.4.Настройка автоматизации 19
3.5.Регистрация пользователей и разграничение прав доступа 21
4.Устранение неисправностей 23

Перечень условных обозначений и сокращений

Модуль	Логическое объединение данных в системе в соответствии с выполняемым функционалом
Коллекция	Единица организации данных в системе, аналог таблицы в базе данных
Объект коллекции	Запись в коллекции системы, аналог строки в таблице базы данных
Объединяющая коллекция	Особый тип коллекции, которая связывает коллекции-источники по общему признаку и отображает информацию в едином интерфейсе, аналог сводной таблицы

1.Возможности системы

Система информационной безопасности InfoZorok - это программное решение, которое интегрирует несколько функций безопасности в единую платформу.
Выполняет такие задачи как инвентаризация и управление ИТ-активами, контроль конфигураций, управление событиями, а так же централизованный сбор и анализ данных за счет интеграции с другими системами информационной безопасности.



Гибкая структура решения позволяет оперативно добавлять новый функционал. Имеет единую консоль, что обеспечивает централизованной контроль различных аспектов безопасности.
Система предоставляет возможность улучшить защиту за счет комплексного подхода к безопасности. Это помогает снизить сложность управления безопасностью и повысить эффективность обнаружения и реагирования на угрозы.

2.Установка системы

Администратор системы должен обладать базовыми знаниями:
- технологии контейнеризации
- программирования на языке python
- язык запросов sql (postgresql)

Перед началом работы рекомендуется ознакомиться с Инструкцией пользователя.

Система состоит из трех компонентов:
1.Сервер - содержит веб-сервер и базу данных
2.Агент - содержит службу с помощью которой осуществляется подключение к источникам информации, сбор этой информации и ее обработка в базе данных
3.Сервер лицензий (на стороне вендора) - подтверждение запросов на выпуск контрольного значения лицензии

2.1.Системные требования

Infozorok Server:
OS - Ubuntu 24.10 Server
RAM - 4Gb
Storage - 50Gb
Infozorok Agent:
OS - Ubuntu 24.10 Server
RAM - 4Gb
Storage - 50Gb
Для корректной установки необходимо чтобы оба сервера имели записи в DNS, корректно настроенную службу времени NTP и установленную последнюю версию Docker Swarm и PSQL.

Допускается установка сервера и агента на одном хосте.

2.2.Подготовка

Для работы системы на серверах необходимо установить Docker Swarm и PSQL. Для этого выполните следующие команды:
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh get-docker.sh
sudo usermod -aG docker $USER
перезагрузите сервер - sudo reboot now
docker swarm init --advertise-addr <ip-адрес сервера в локальной сети>

sudo apt install -y postgresql-client

Распакуйте полученный от вендора дистрибутив в домашний каталог с помощью команды от sudo:
sudo tar -xzf iz.tar.gz

Каталог дистрибутива имеет следующую структуру:

```
iz
├── izagent
│   ├── Dockerfile
│   ├── izagent
│   ├── requirements.txt
│   ├── izagent_install.sh
│   ├── cert
│   ├── lib
│   └── keys
│       ├── izagent_encrypt_private.pem
│       ├── izagent_encrypt_public.pem
│       ├── izagent_ssl.crt
│       ├── izagent_ssl.key
│       └── izca_ssl.crt
├── izca
│   └── keys
│       ├── izca_ssl.crt
│       ├── izca_ssl.key
│       └── izca_ssl.srl
├── izserver
│   ├── data
│   │   ├── iz-extensions
│   │   ├── iz-uploads
│   │   └── postgresql-data
│   ├── izserver_install.sh
│   └── keys
│       ├── izca_ssl.crt
│       ├── izserver_site.crt
│       ├── izserver_site.key
│       ├── izserver_ssl.crt
│       └── izserver_ssl.key
└── keys_generate.sh
```

Каждый корневой каталог относится к компоненту системы:
1.izserver - сервер Infozorok (веб-сервер и база данных)
2.izagent - агент InfoZorok
3.izca - ключи CA, на основе которых сформированы ключи для веб-сервера (TLS)
В корне находится скрипт для генерации всех необходимых ключей — keys_generate.sh.

Установка состоит из четырех этапов:
1. Генерация ключей (этот этап можно пропустить и воспользоваться для ознакомления с системой уже сгенерированными тестовыми ключами)
2. Установка сервера

3. Установка агента
4. Активация продукта

2.3.Генерация ключей шифрования

1.Перейдите в каталог iz и выполните команду
cd ~/iz
./keys_generate.sh
введите FQDN имя сервера, на который будет установлена система InfoZorok
В результате будут сгенерированы криптографические ключи агента, сервера, локального центра сертификации и помещены в каталоги keys соответствующего дистрибутива (izagent, izserver, izca).

2.Для обеспечения безопасности переместите сгенерированные ключи локального удостоверяющего центра (каталог iz/izca) во внешнее защищенное хранилище.

2.4.Установка сервера

1.Перейдите в каталог iz/izserver и выполните команду
cd ~/iz/izserver
./izserver_install.sh
введите FQDN имя сервера InfoZorok. В ходе установки необходимо будет придумать пароль для сервисной учетной записи базы данных системы.
2.Проверьте корректность работы стека Infozorok командой
docker service ls
все сервисы должны быть запущены (статус "1/1")
3.Выполните вход в интерфейс сервера InfoZorok (https://<FQDN сервера, который вы указывали при установке>). Начальный логин - admin@admin.com пароль - d@faultp#ssword
4.Смените логин и пароль администратора в профиле (в левом нижнем углу), поля Email и Password.
5.Перейдите в меню System->Settings и заполните полученные от вендора лицензионный ключ и срок его действия в двойных кавычках (“license_key” и “license_expiry” в формате ГГГГ-ММ-ДД).

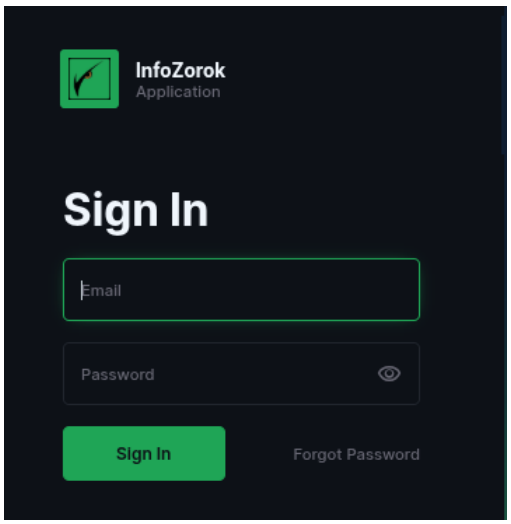
2.5.Установка агента

1.Перенесите каталог с дистрибутивом агента (iz/izagent) на нужный сервер и выполните команду
cd ~/iz/izagent
./izagent_install.sh
введите FQDN имя сервера InfoZorok, уровень логирования (нажмите ENTER для установки рекомендуемого значения) и интервал работы агента (нажмите ENTER для установки рекомендуемого значения).
2.Проверьте корректность работы стека агента командой
docker service ls

сервис должны быть запущен (статус "1/1")

2.6.Активация агента

1.Выполните вход в интерфейс сервера InfoZorok
Для доступа к интерфейсу системы необходимо перейти по адресу, где опубликован web-сервер системы (https://<адрес сервера>/admin/login). Начальный логин - admin@admin.com пароль - d@faultp#ssword



2.В меню Настройки->Settings, скопируйте сертификат и запрос на контрольное значение ("license_certificate" и "license_control_value_request") и отправьте их вендору.

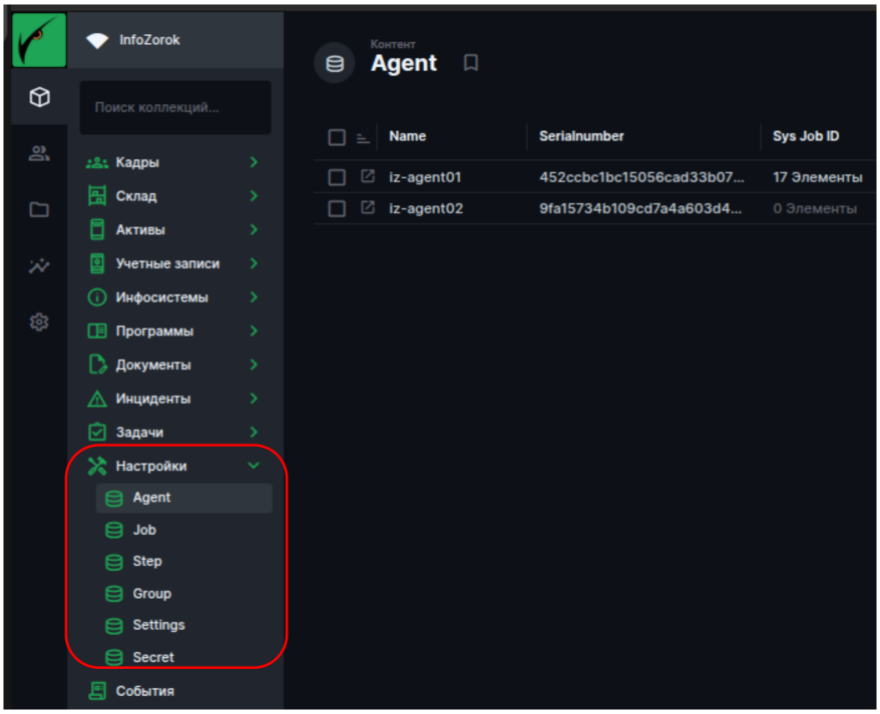
Примечание. На стороне вендора. После получения от клиента запроса на контрольное значение и сгенерированный им сертификат, администратор на стороне вендора запускает утилиту infozorok_distr/vendor/izlserver/generate-cv.
Далее последовательно вводит, без двойных кавычек:
- пароль секретного ключа (secret key password)
- control value request
- license certificate
- license key value
Проверяет содержимое запроса, генерирует контрольное значение (Control value) и отправляет клиенту

3.После получения от вендора контрольного значения, сохраните его в "license_control_value".
4.Перейдите в меню Настройки->Agent, найдите вновь созданного агента (в поле Sys Job Id будет 0 items). Назначьте новому агенту тестовую задачу, например test_job1, которая будет записывать строку "agent1" в коллекцию test1.
5.Перейдите в главном меню в Test->test1, подождите период обновления (30 секунд) и убедитесь, что новые значения добавляются.
6.Логи работы системы и возможные ошибки можно посмотреть в журнале, пункт Event главного меню.

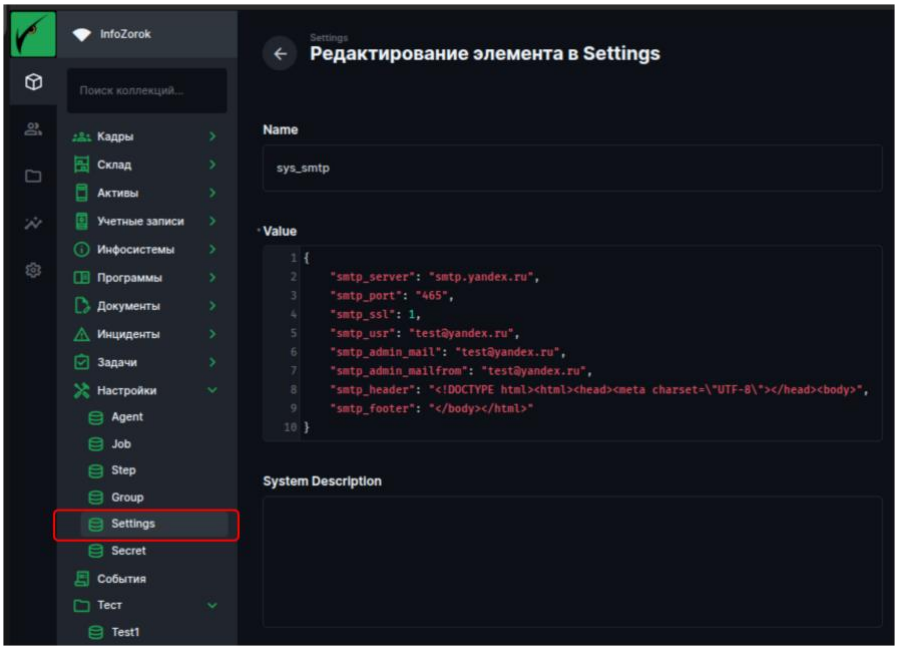
3.Администрирование системы

3.1.Общие принципы

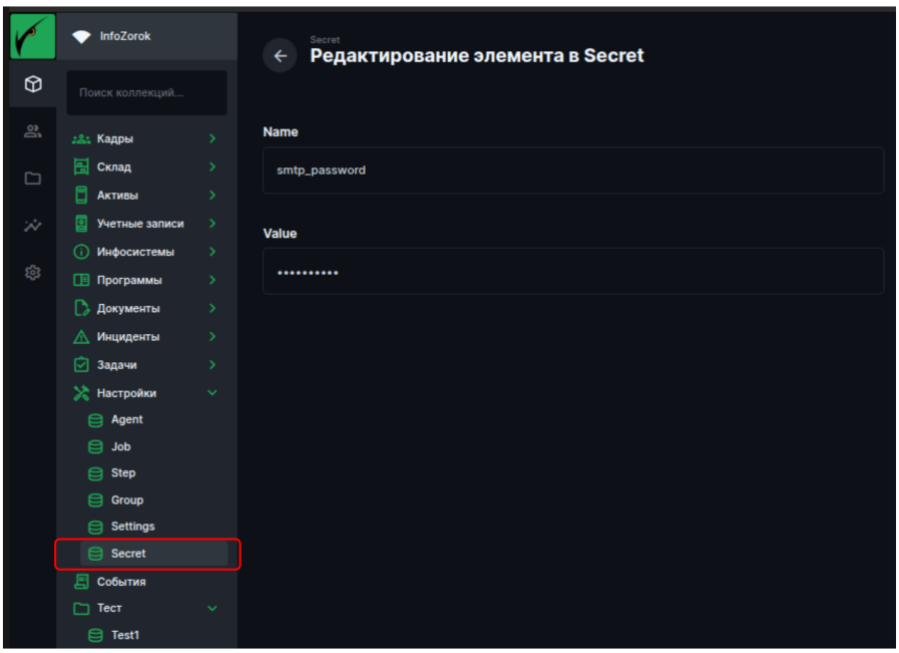
Администрирование системы осуществляется в модуле Настройки.



Настройки системы задаются в коллекции Setting, например SMTP-сервер для отправки алертов, параметры подключения к серверам HR-систем, LDAP-каталогу и др. Для добавления или изменения настройки необходимо выбрать значение с соответствующим именем и изменить значение в виде JSON-строки.



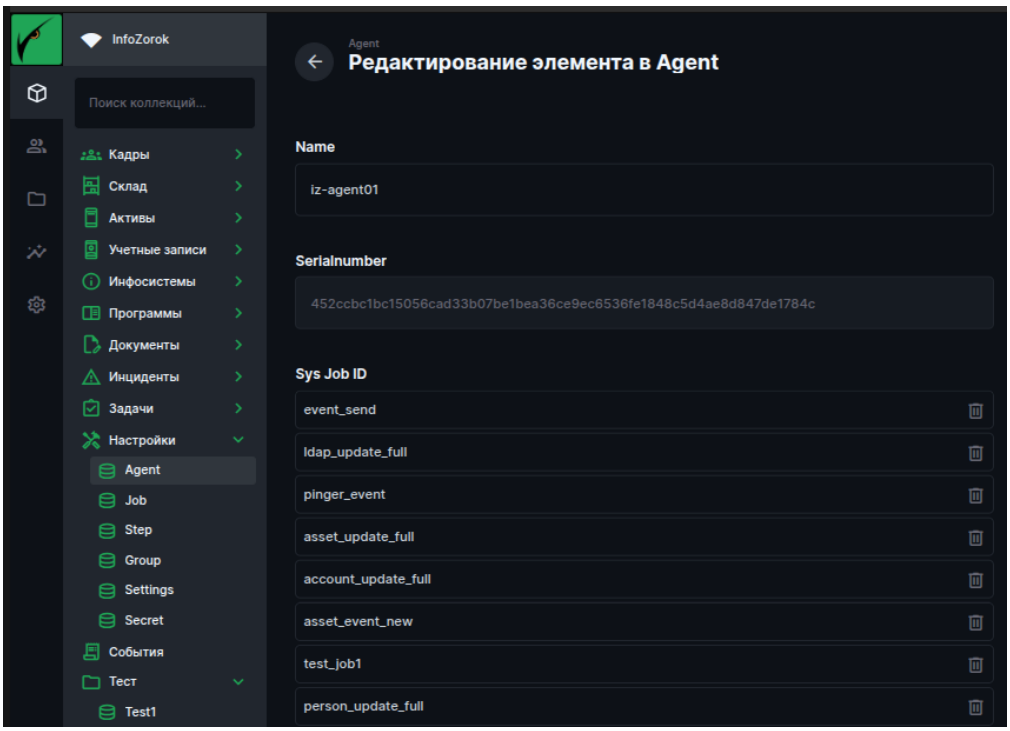
Пароли, API-ключи и другая секретная информация хранится в системе в зашифрованном на ключах агента виде, в коллекции Secret. Чтобы добавить секрет, необходимо выбрать значение с соответствующим именем и изменить значение.



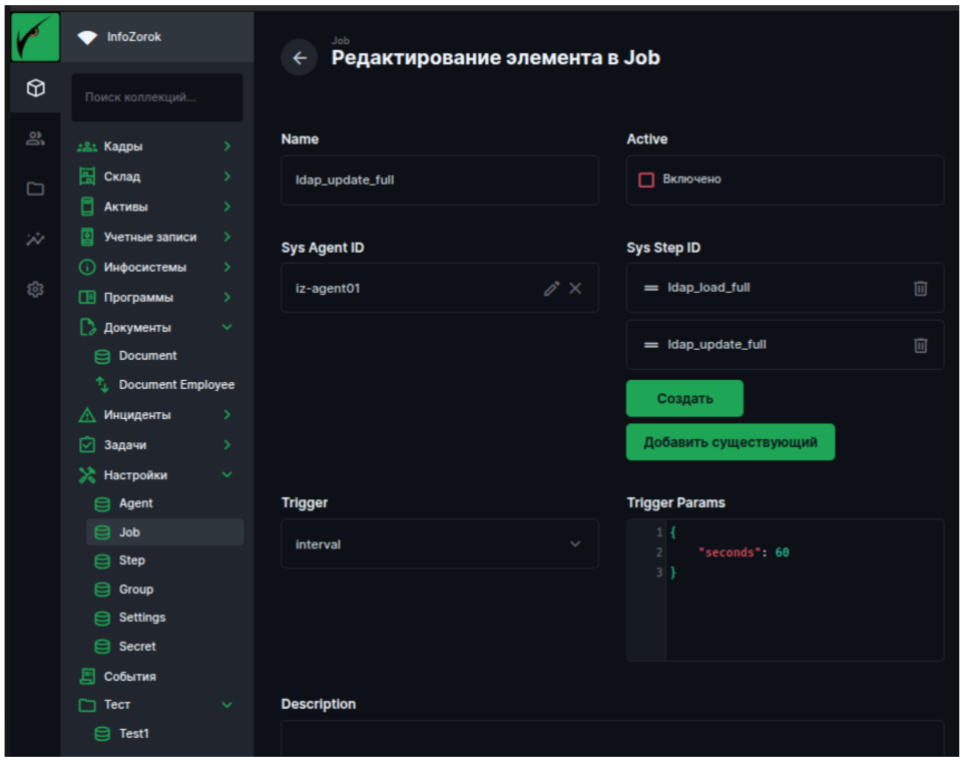
Сбор и обработка информации из внешних систем выполняется агентами, на которых настраиваются соответствующие инструкции и расписание их выполнения.

Принцип работы агентов следующий. Создается работа (Job), которая назначается агенту. В работе указывается расписание ее запуска. В состав работы входит один или несколько шагов (Step), в параметрах которых указываются:
- тип шага (update — интеграция с внешним источником и загрузка данных в систему, event — мониторинг на предмет выявления несоответствия и отправка алерта)
- инструкции и что именно необходимо сделать.

Информация о зарегистрированных агента находится в коллекции Agent и содержит:
- имя агента
- серийный номер
- ссылки на работы, выполняемые агентом

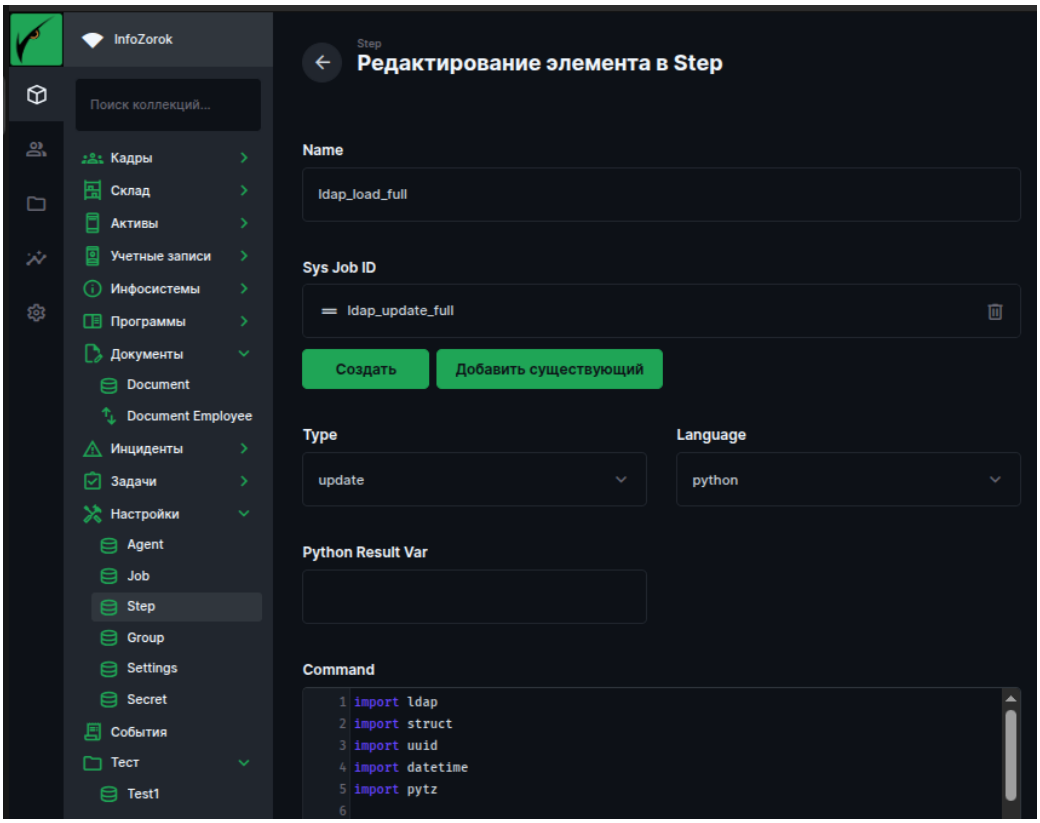


Работы настраиваются в коллекции Job. Каждая работа имеет несколько параметров:
- имя
- тип запуска (interval — раз в период, date - по наступлению даты, once — однократно, on_start — при запуске агента)
- параметры запуска (если interval — указываются часы/минуты/секунды, если дата — в формате «ГГГГ-ММ-ДД ЧЧ:ММ:СС»)



Шаги с инструкциями выполнения указываются в коллекции Step (Шаг).

Step имеет следующие параметры и инструкции:
- имя
- sys job id - работа, в рамках которой выполняется шаг
- тип (update — загрузка данных/обновление, event — задача мониторинга)
- language - язык, на котором будут написаны инструкции (python/sql)
- python result var — имя коллекции, которая будет содержать данные по выявленным событиям, если выбран тип event и язык python
- command — инструкции для выполнения

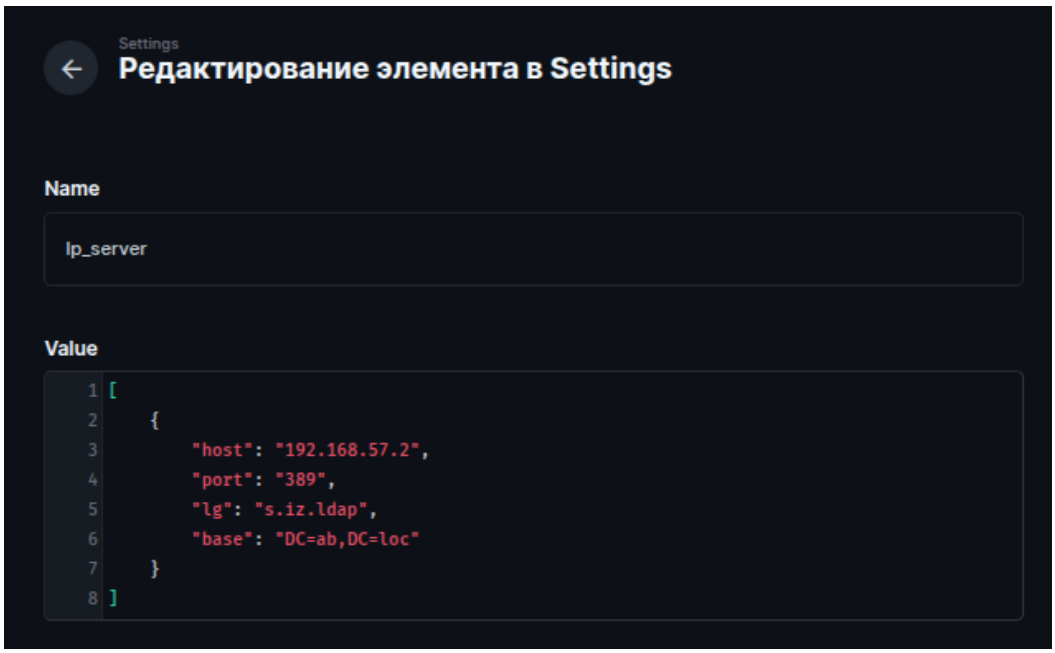


Для отправки алертов через электронную почту необходимо создать группу рассылки в коллекции Group и добавить в группу аккаунт (из коллекции Account).

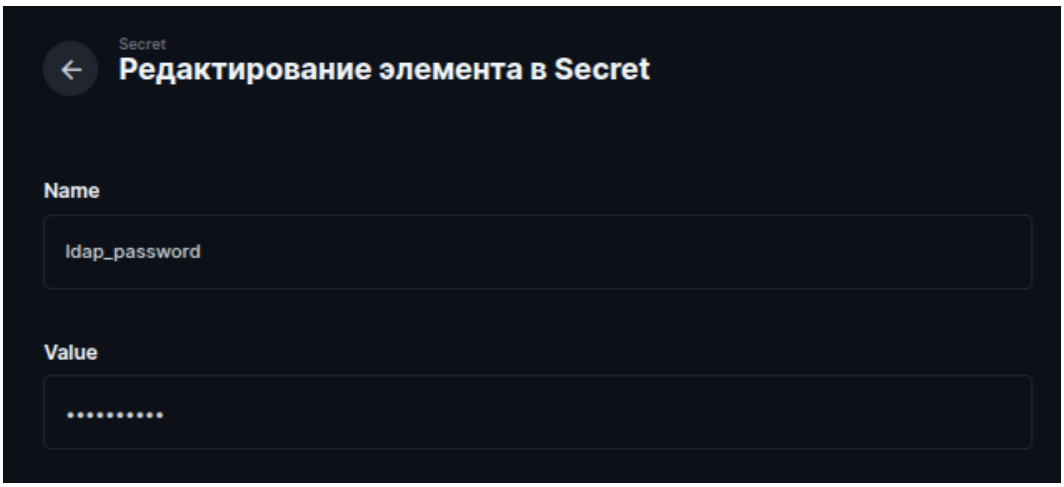
3.2.Настройка интеграций

Приведем пример настройки интеграции с каталогом LDAP, для получения информации о пользователях.

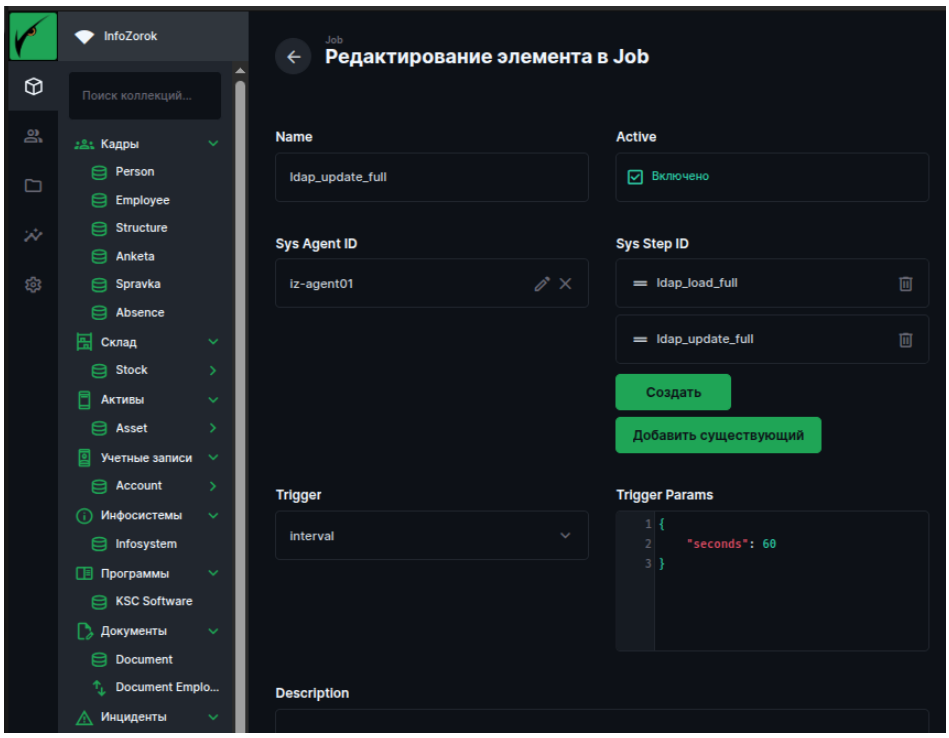
1.Добавляем в коллекцию Setting параметры подключения к контроллеру домена, откуда будут выгружаться данные:



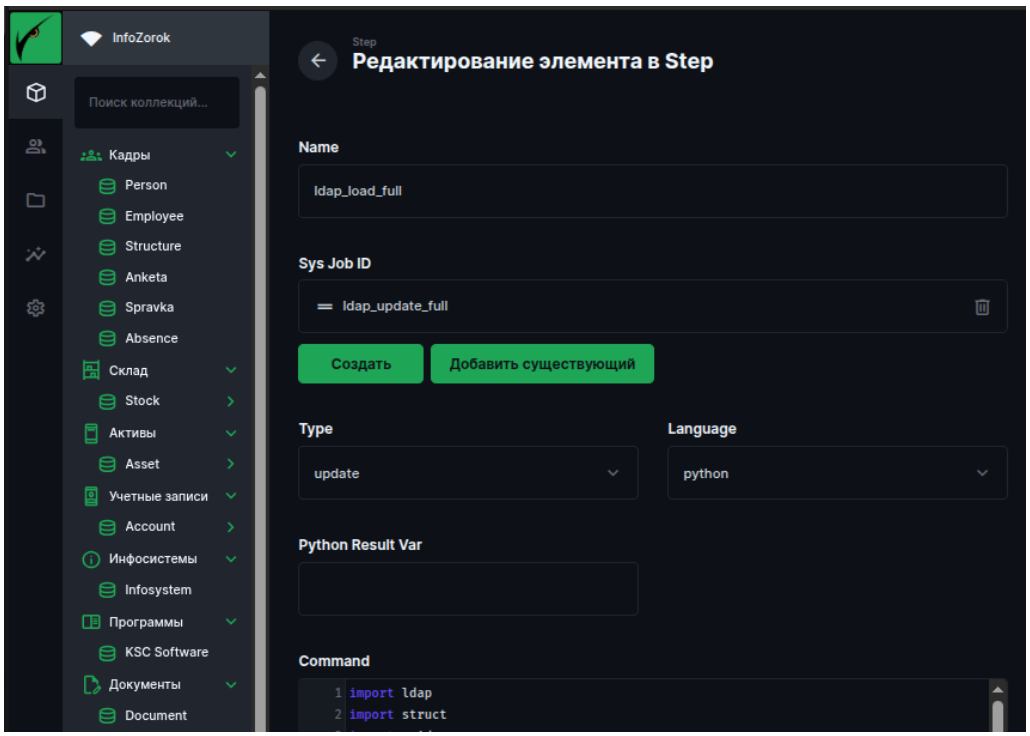
2.В коллекции Secret добавляем пароль для подключения к LDAP:



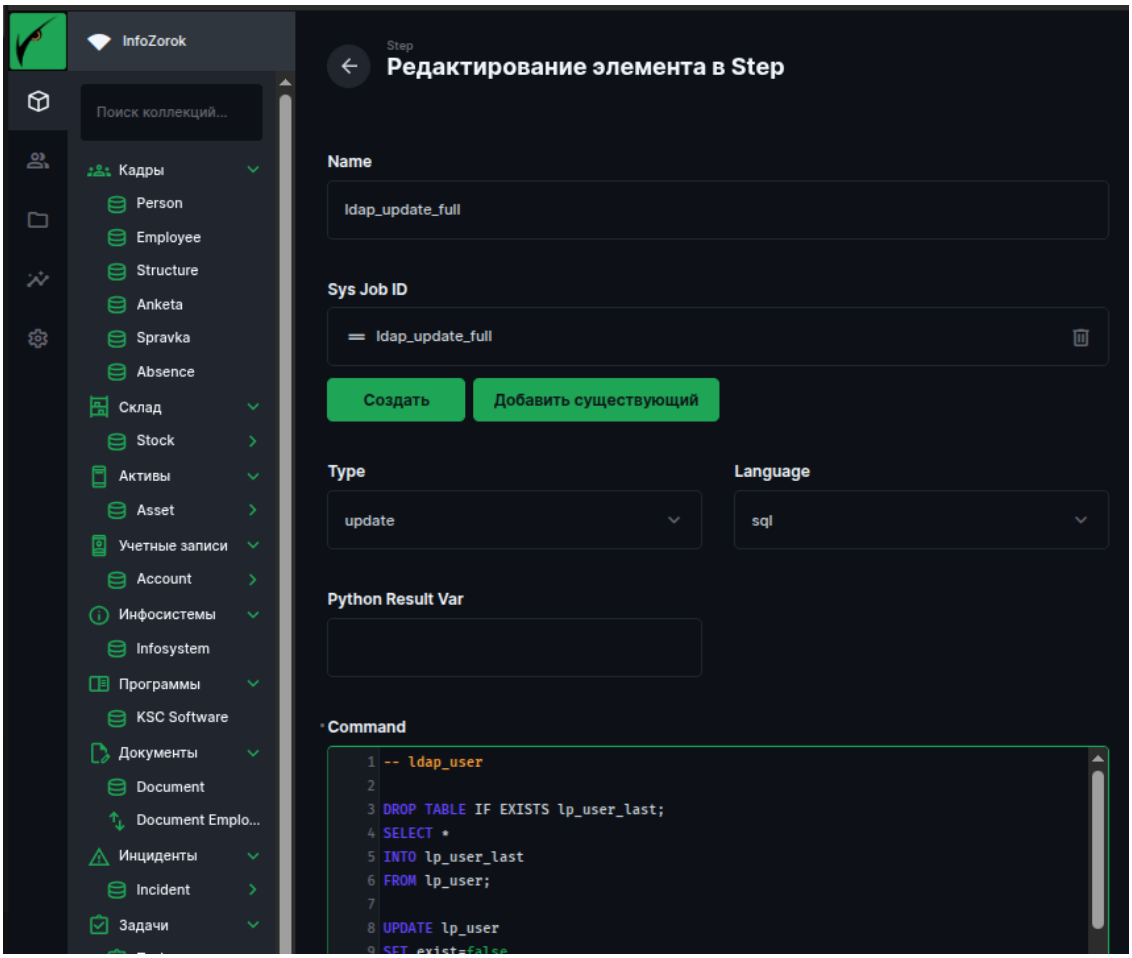
3.Настраиваем Job:
- name: ldap_update_full
- active: выключено
- agent: выбираем агент, который будет выполнять работу
- sys step id: ссылка на step-ы, которые загружают данные об учетных записях пользователей из LDAP и обновляют данные базе данных (создание описано ниже)
- trigger: interval (периодами)
- trigger params: { seconds: 60 } (работа будет запускаться раз в минуту)



4.Настраиваем Step выгрузки данных из LDAP:
- name: ldap_load_full
- sys job id: в рамках какой работы будет выполняться step — ldap_update_full
- type: update
- language: python (язык программирования, на котором описана инструкция)
- command: сама инструкция на языке python, которая получает параметры сервера и секрет (пароль) для подключения, подключается к LDAP и загружает данные в виде списка, сохраняет список во временную таблицу (полный листинг инструкции алерта можно посмотреть в уже настроенном шаге)

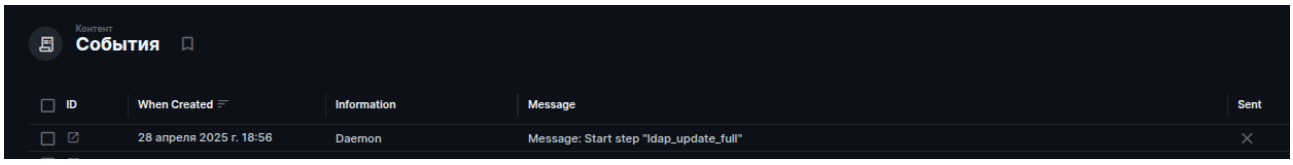


5.Настраиваем Step для обновления данных в базе:
- name: ldap_update_full
- sys job id: в рамках какой работы будет выполняться step — ldap_update_full
- type: update
- language: sql (язык программирования, на котором описана инструкция)
- command: инструкция на языке sql, которая делает копию таблицы для выявления изменений, используя временную таблицу, полученную на предыдущем шаге, обновляет целевую таблицу - помечает удаленные объекты, обновляет существующие, вставляет новые (полный листинг инструкции алерта можно посмотреть в уже настроенном шаге)



6.Активируем работу с помощью параметра «active» Job-а.

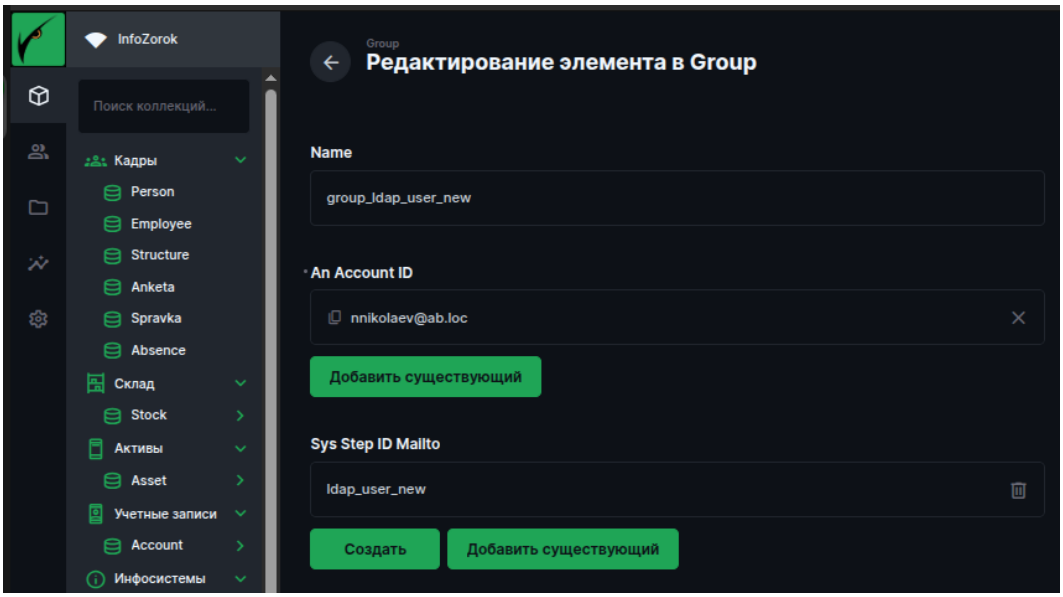
7.Проверяем в системном журнале, что работа запустилась и ошибок нет:



3.3.Настройка алертов

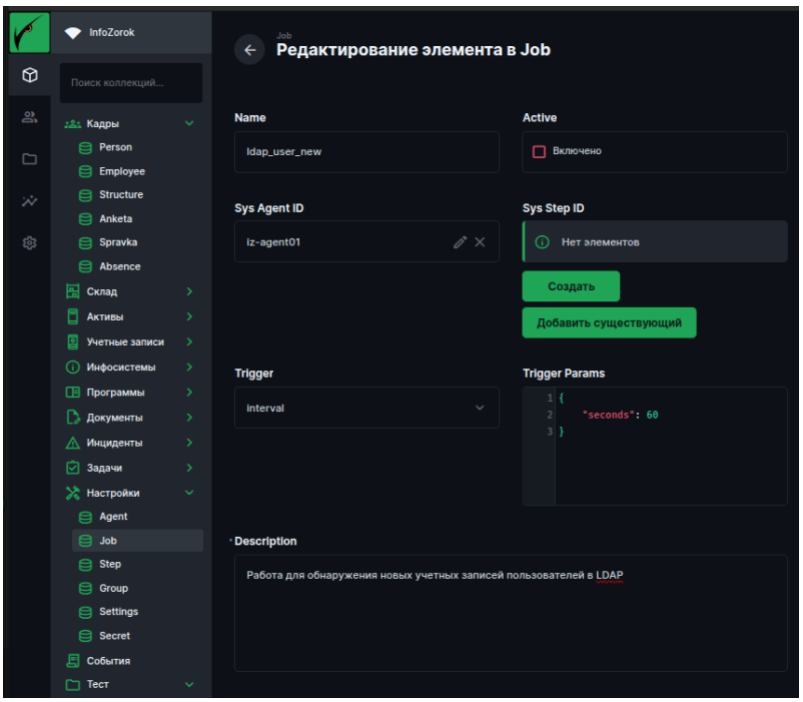
Приведем пример настройки алерта для выявления новых учетных записей пользователей в каталоге LDAP.

1.Настраиваем системную группу для отправки алертов со следующими параметрами:
- name: «group_ldap_user_new»
- an account id: список учетных записей для отправки
- sys step id mailto: ссылка на step, который выявляет новые учетные записи в LDAP (создание описано ниже)

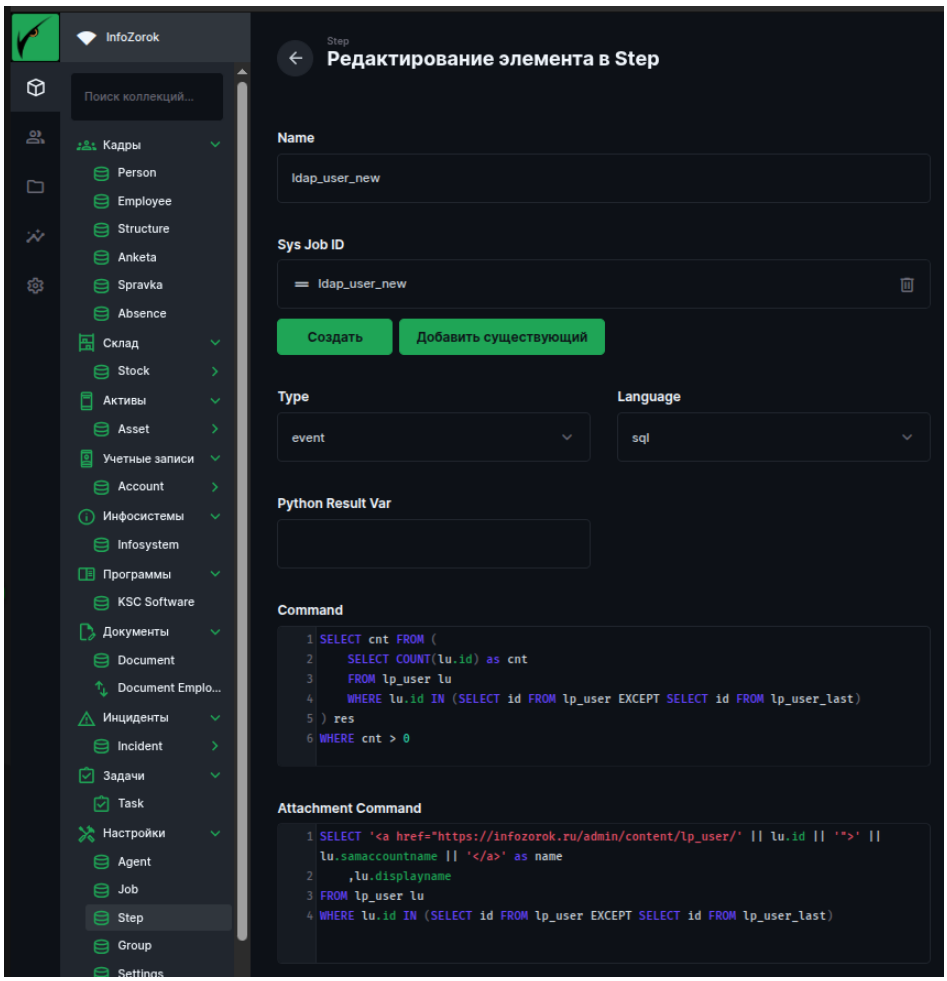


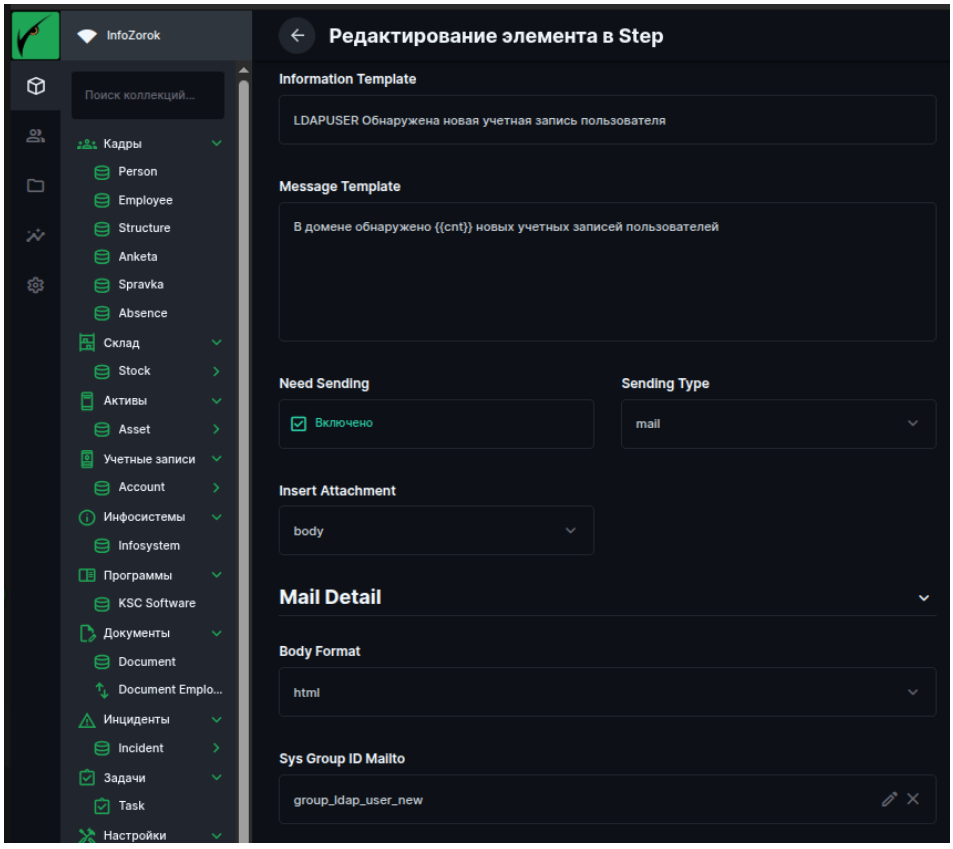
2.Настраиваем Job:
- name: ldap_user_new
- active: выключено
- agent: выбираем агент, который будет выполнять работу
- sys step id: ссылка на step, который выявляет новые учетные записи в LDAP (создание описано ниже)
- trigger: interval (периодами)
- trigger params: { seconds: 60 }

Пример создания работы:



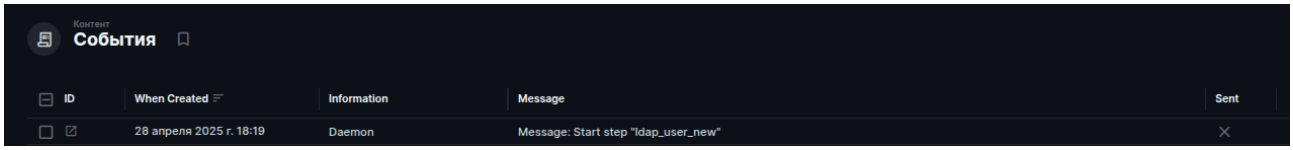
3.Настраиваем Step:
- name: ldap_user_new
- sys job id: в рамках какой работы будет выполняться step — ldap_user_new
- type: event
- language: sql
- command: инструкция на языке sql, которая выводит количество новых строк с момента последнего обновления
- attachment command: инструкция на языке sql, которая выводит сами новые строки добавленные в базу с момента последнего обновления
- information template: заголовок email-сообщения - «LDAPUSER Обнаружена новая учетная запись пользователя»
- message template: шаблон email-сообщения в котором используется плейсхолдер из command - «В домене обнаружено {{cnt}} новых учетных записей пользователей»
- need sending: требуется отправка, или алерт будет просто записываться в системный журнал
- sending type: каким способом будет отправляться уведомление - «mail»
- insert attachment: как будет добавлена информация из attachment command (в тело письма в виде таблицы, или файлом вложения) - «body»
- sys group id mailto: системная группа, на учетные записи которой будет отправляться алерт - «group_ldap_user_new»



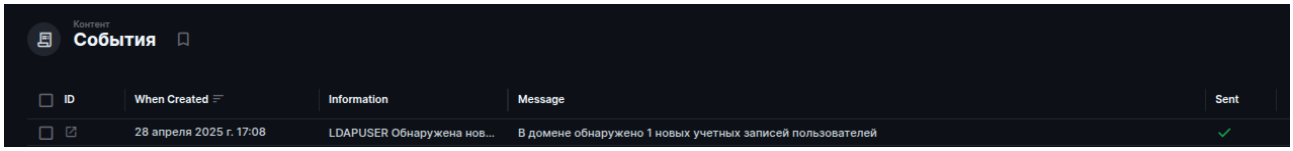


4.Активируем работу с помощью параметра active.

5.Проверяем в системном журнале, что работа запустилась и ошибок нет:



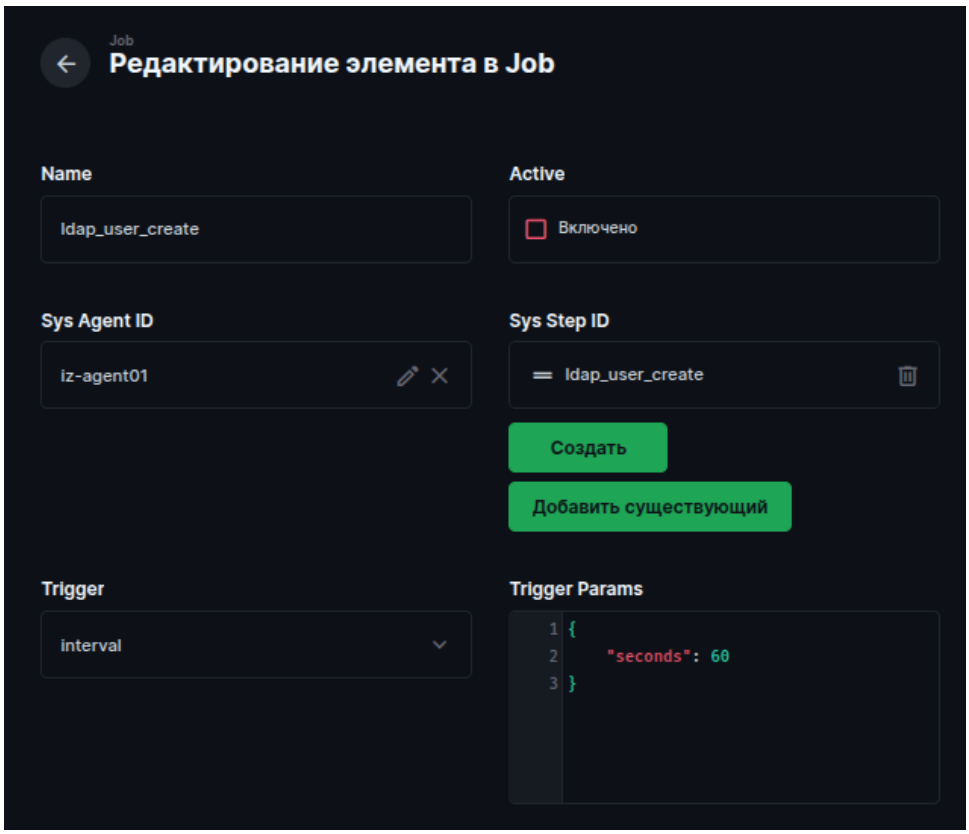
При добавлении новой учетной записи в LDAP в журнале и в почте появится соответствующий алерт:



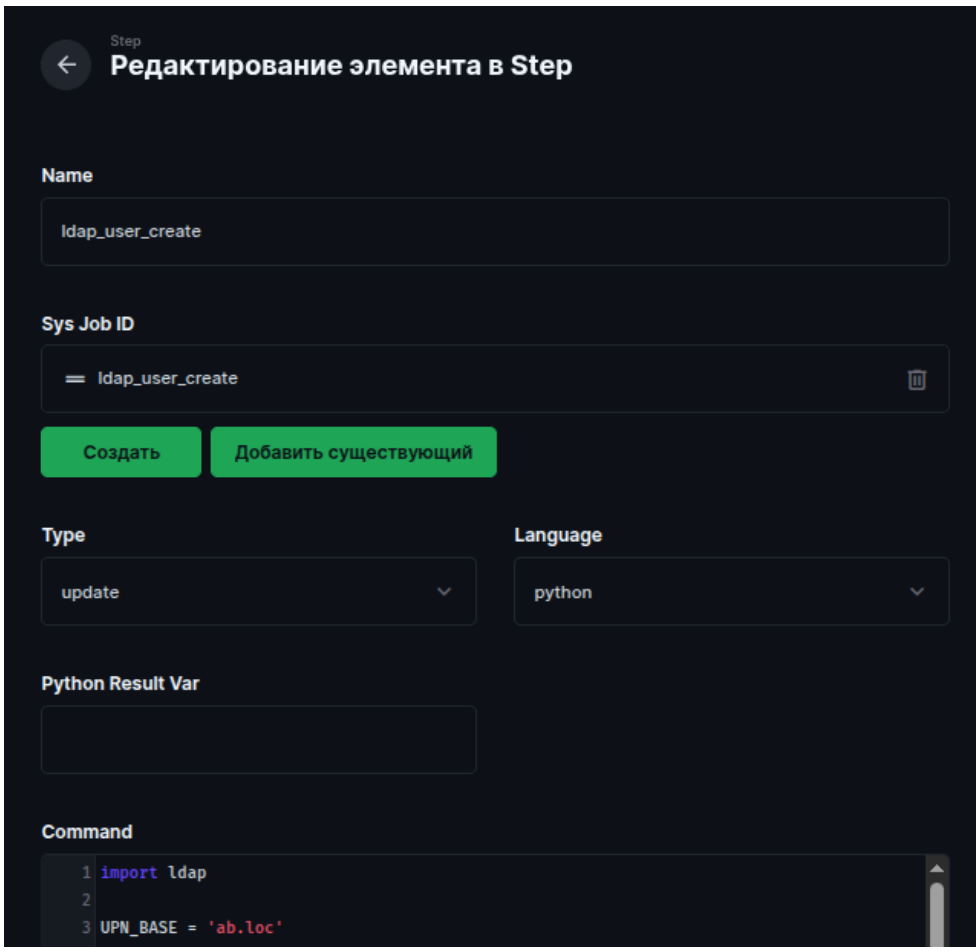
3.4.Настройка автоматизации

В качестве примера автоматизации приведем пример автоматического создания учетной записи пользователя при появлении в штатном расписании нового работника.

1.Настраиваем Job:
- name: ldap_user_create
- active: выключено
- agent: выбираем агент, который будет выполнять работу
- sys step id: ссылка на step, который будет подключаться к LDAP и создавать учетные записи пользователей с данными из кадровой системы (создание описано ниже)
- trigger: interval (периодами)
- trigger params: { seconds: 60 }

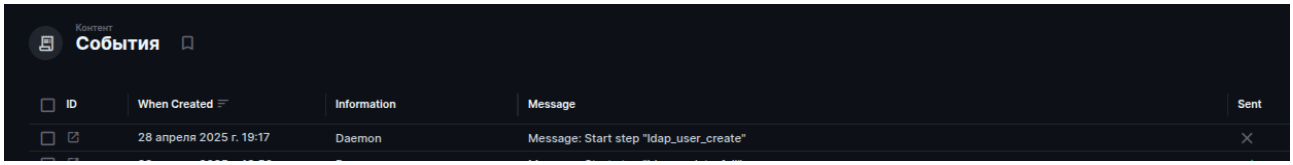


2.Настраиваем Step автоматического создания учетных записей в LDAP:
- name: ldap_user_create
- sys job id: работа, в рамках какой работы будет выполняться step — ldap_user_create
- type: update
- language: python
- command: инструкция на языке python, которая получает данные о новом работнике из соответствующей коллекции, подключается к LDAP и создает учетную запись с необходимыми атрибутами (полный листинг инструкции алерта можно посмотреть в уже настроенной работе)



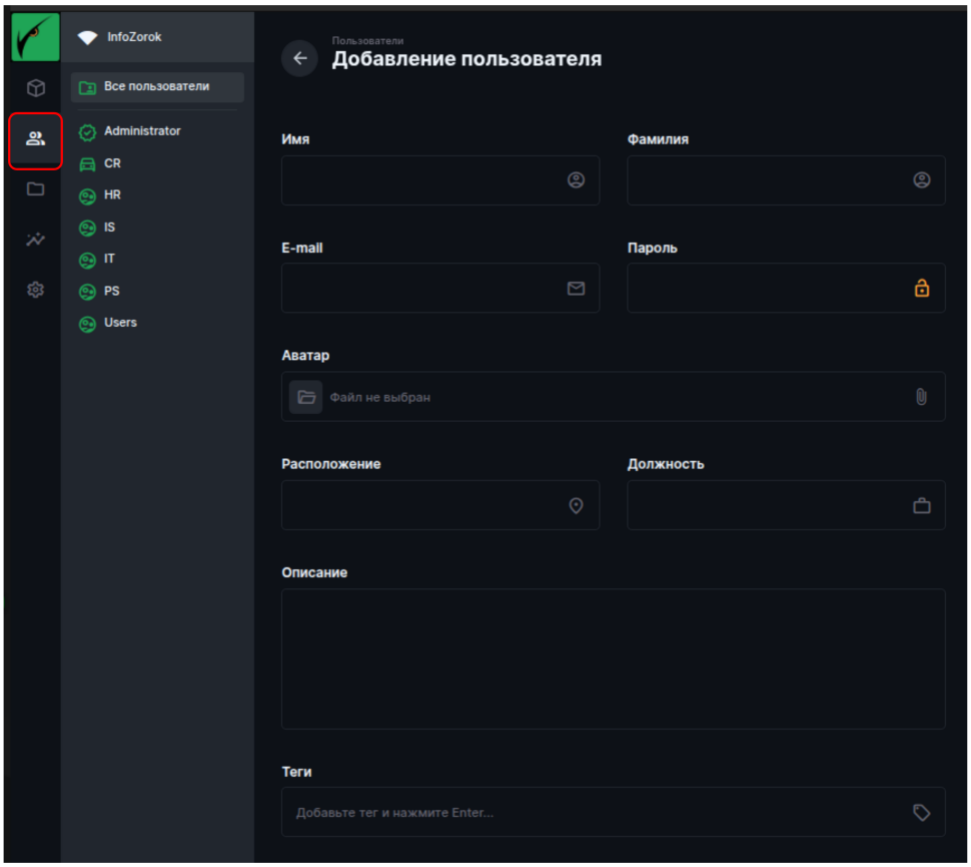
3.Активируем работу с помощью параметра «active» Job-а.

4.Проверяем в системном журнале, что работа запустилась и ошибок нет:



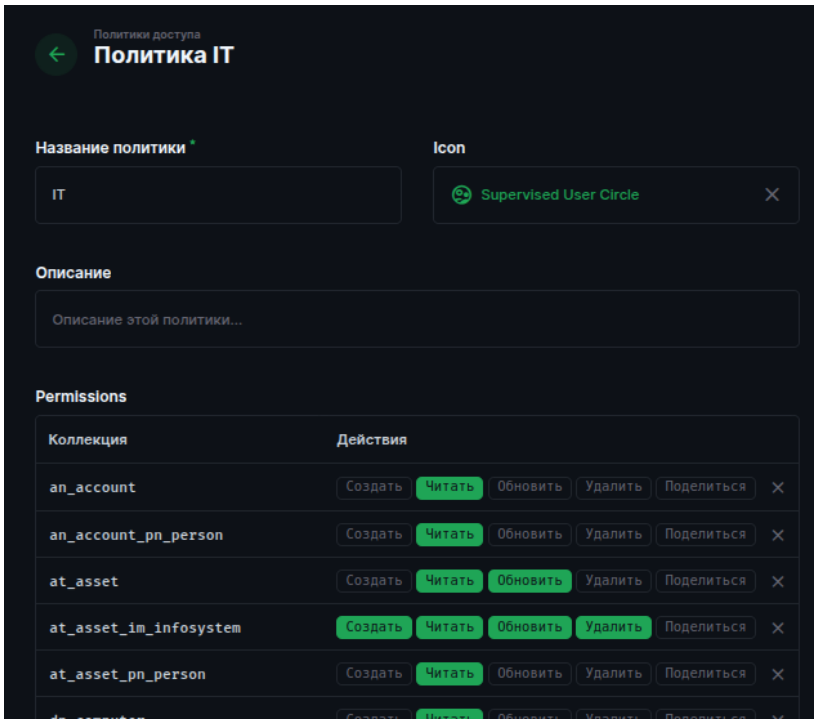
3.5.Регистрация пользователей и разграничение прав доступа

Регистрация пользователей системы осуществляется в модуле Пользователи. Здесь можно задать:
- имя пользователя
- адрес электронной почты
- первоначальный пароль
- необходимость ввода второго фактора
- параметры отображения интерфейса и др.

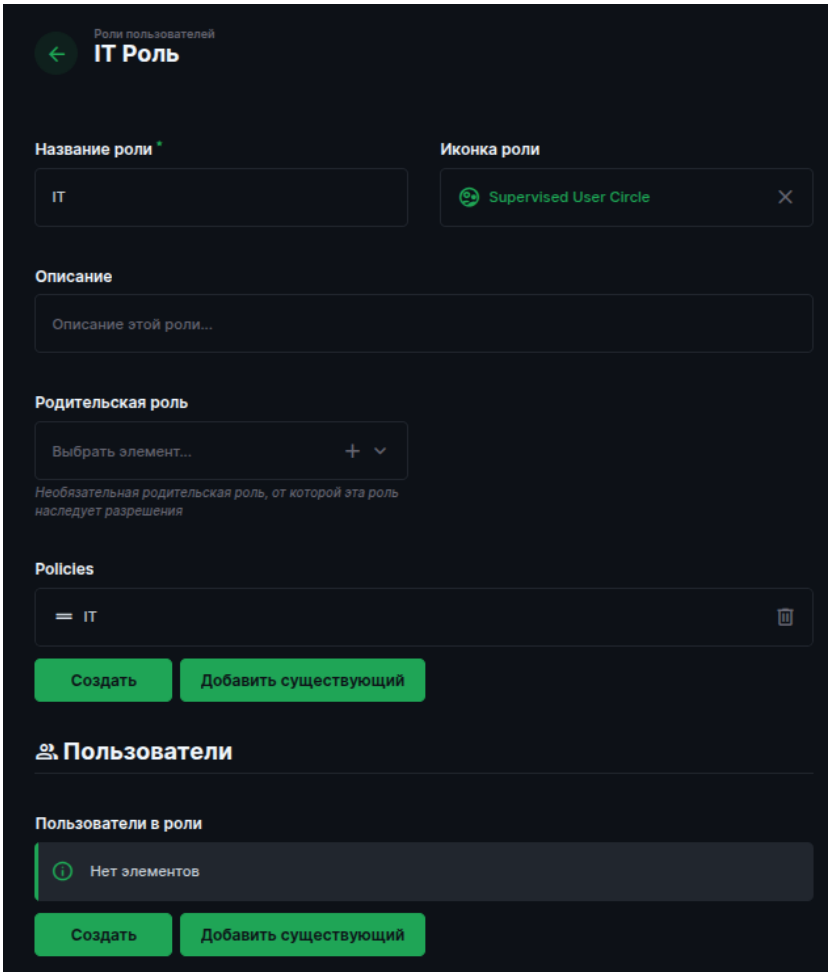


Разграничение прав доступа в системе выполняется с помощью создания политики прав доступа, где указываются какие права и к каким объектам системы будет иметь доступ получатель этой политики. Политику можно назначить на конкретного пользователя, или при унифицированном подходе - роли, а роль, в свою очередь пользователям. Роли можно включать друг в друга, создавая иерархию.

Пример определения политики:



Пример определения роли:



4.Устранение неисправностей

В случае выявления неисправности, причины некорректной работы можно отследить в системном журнале — коллекция Event.
Если устранить причины неисправности самостоятельно не представляется возможным, вы можете обратиться в службу технической поддержки вендора по адресу электронной почты support@ib-nova.ru. В обращении необходимо указать подробное описание и приложить содержимое системного журнала.