Solar appScreener

Руководство системного администратора Solar appScreener модуль анализа безопасности цепочек поставок ПО (SCS)

Solar appScreener

Версия 3.13.9

Ноябрь 2023

СОДЕРЖАНИЕ

1. АННОТАЦИЯ

Настоящий документ представляет собой руководство по установке и настройке Solar appScreener модуль анализа безопасности цепочек поставок ПО (далее **Модуль SCS**). Документ предназначен для интеграторов и системных администраторов модуля SCS и содержит описание процедур установки, настройки и сопровождения модуля SCS.

2. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Аббревиатура	Расшифровка
АРМ	Автоматизированное рабочее место
БД	База данных
ОС	Операционная система
ПО	Программное обеспечение
СУБД	Система управления базами данных
UI	User Interface – интерфейс пользователя
CLI	Command Line Interface – интерфейс командной строки
SDLC	System Development Life Cycle – жизненный цикл разработки системы

3. СВЕДЕНИЯ О МОДУЛЕ АНАЛИЗА БЕЗОПАСНОСТИ ЦЕПОЧЕК ПОСТАВОК ПО (SCS)

3.1. Назначение

Модуль SCS предназначен для выявления рисков атаки на цепочки поставок ПО в библиотеках с открытым исходным кодом, используемых в коде приложения.

3.2. Описание возможностей

Модуль SCS предоставляет следующие возможности:

- Анализ рисков цепочек поставок.
- Мониторинг изменения уровня безопасности приложения.

  В модуле SCS реализована возможность генерации отчётов для получения информации по результатам анализа проектов. Отчёты можно генерировать в формате DOCX, PDF, CSV и HTML и отправлять по почте. Также результаты анализа можно просматривать и сравнивать непосредственно в веб-интерфейсе модуля SCS.

3.3. Перечень эксплуатационной документации для ознакомления

В поставку модуля SCS входят следующие эксплуатационные документы:

- Руководство системного администратора (для Windows и Linux);
- Руководство пользователя.

4. ТРЕБОВАНИЯ К СЕРВЕРНОЙ И КЛИЕНТСКОЙ ЧАСТЯМ

4.1. Требования к аппаратному обеспечению

4.1.1. Серверная часть

Система поставляется модулями. Для корректной работы требуется последовательно установить необходимый набор модулей. Модули можно установить на один сервер или на несколько серверов, связанных в одну сеть. Ссылки на скачивание модулей содержатся в инструкции по установке модуля SCS.

- APP модуль - обязательный модуль, веб приложение, отвечает за логику работы всей системы;
- SCS модуль - модуль анализа безопасности цепочек поставок ПО.

4.1.1.1. Минимальные характеристики оборудования для установки на одном сервере

Для функционирования модуля SCS на одном сервере, требуется оборудование со следующими минимальными характеристиками:

- 10 ядерный процессор с тактовой частотой 2.2 ГГц;
- объем оперативной памяти – 24 ГБ;
- минимальный объем жесткого диска – 600 ГБ SSD/ SAS HDD (при увеличении количества сканирований требуемый объем может увеличиться);
- поддерживаемые операционные системы.

В зависимости от количества сканирований безопасности цепочек поставок ПО (SCS) минимальные характеристики могут возрасти.

4.1.1.2. Минимальные характеристики для установки модулей на отдельных серверах

Для развертывания модуля SCS на отдельном сервере требуются следующие минимальные характеристики:

- 4 ядерный процессор с тактовой частотой 2.2 ГГц;
- объем оперативной памяти – 16 ГБ;
- минимальный объем жесткого диска – 100 ГБ SSD/ SAS HDD;
- поддерживаемые операционные системы (см. Требования к программному обеспечению).

4.1.2. Клиентская часть

АРМ администратора модуля SCS должно быть оборудовано персональным компьютером с подключением к внутренней сети компании.

4.2. Требования к программному обеспечению

4.2.1. Серверная часть

Для функционирования модуля SCS на серверном оборудовании должно быть установлено следующее программное обеспечение:

- Операционная система:
  - Ubuntu 20.04 LTS;
  - Ubuntu 22.04 LTS;
  - Red Hat Enterprise Linux 8 (RHEL8);
  - CentOS 7;
  - RedOS 7;
  - Astra Linux Special Edition 1.7.3+.

Обязательно требуется установить модуль **ENV**: предоставляется с дистрибутивом модуля SCS, строго под каждую операционную систему из списка.

Также на серверной части должны выполняться следующие требования к аппаратно-программному комплексу:

- чистая операционная система без предустановленного стороннего ПО;
- наличие и доступность пакетов из базовых репозиториев для операционной системы;
- привилегии пользователя root/administrator;
- свободные TCP-порты 80, 443, 61616.

4.2.2. Клиентская часть

В состав программного обеспечения компьютера должна входить программа-клиент, предоставляющая пользователю возможность навигации и просмотра web-ресурсов (браузер). Рекомендуемые браузеры (актуальные версии):

- Mozilla Firefox;
- Google Chrome;
- Safari;
- Internet Explorer;
- Microsoft Edge.

Рекомендуем в настройках браузера разрешить выполнение **JavaScript** и сохранение файлов **cookies**.

5. ФУНКЦИОНАЛЬНАЯ СТРУКТУРА

Solar appScreener модуль анализа безопасности цепочек поставок ПО (SCS) включает следующие модули:

- **Web-приложение**:
  - **UI**;
  - **API**;
  - **Backend**;
  - **DB**.
- **Message broker**.
- **Daemon**.
- **Модуль анализа безопасности цепочек поставок ПО (SCS)**:

Web-приложение. Модуль представляет собой веб-приложение, развернутое на сервере Apache Tomcat. Через **Web-приложение** осуществляется взаимодействие с остальными компонентами модуля SCS. **Web-приложение** включает компоненты **UI** и **Backend**.

- **UI**. Компонента веб-приложения – пользовательский интерфейс. **UI** взаимодействует с **Backend**.
- **Backend.** Компонента веб-приложения. Выполняет все сложные/долгие операции:
  - обновление БД при запуске анализа;
  - взаимодействие с модулем **Message broker**:
    - добавление задачи в очередь;
    - опрос для установления статуса;
    - выгрузка результатов.
  - сохранение результатов в БД;
  - формирование списка рисков Supply chain, подготовка отчётов.

БД. База данных (PostgreSQL) для хранения информации.

Message broker. Реализует очередь с приоритетами и является промежуточным модулем между **Backend** и **Daemon**.

Daemon. Обращается к модулю **Message broker** для получения данных о задачах, запускает соответствующие модули анализа. Выполняет мониторинг работы модулей, отправляет в **Message broker** информацию об обновлениях статуса, а также результаты сканирования.

Модули анализа. Запускаются модулем **Daemon** через CLI. В выбранном формате фиксируют статус и результаты сканирования.

CLI. Command Line Interface. Взаимодействует с модулем **Backend** по сети, предоставляет доступ к функциональности модуля SCS через CLI.

6. ОПИСАНИЕ РАБОТЫ С МОДУЛЕМ АНАЛИЗА БЕЗОПАСНОСТИ ЦЕПОЧЕК ПОСТАВОК ПО (SCS)

6.1. Установка

6.1.1. Порядок установки

От того, какой вариант установки будет предпочтительным: на одном сервере или нескольких, зависят требования к аппаратным ресурсам сервера/серверов (см. Требования к аппаратному обеспечению). Количество серверов и работающих на них модулей можно скорректировать после установки, добавив или удалив необходимые элементы (сервера, модули) в систему.

Обратите внимание: на один сервер может быть установлено только по одному экземпляру модулей из списка ниже:

- Для развертывания системы требуется загрузить и установить обязательный модуль - **ENV** для выбранного дистрибутива ОС (см. Требования к программному обеспечению), его необходимо устанавливать на каждый новый сервер (при новой установке, при добавлении в систему нового сервера). Данный модуль подготавливает окружение сервера к комплексной работе с остальными модулями, устанавливает систему контейниризации (docker). Модуль ENV зависит от операционной системы развернутой на хосте.
- Далее следует установить на сервер модуль веб приложения - **APP**. Данный модуль отвечает за логику работы системы и управляет работой опциональных модулей. APP модуль не зависит от операционной системы и может быть развернут только на одном сервере, т.к. к нему привязывается лицензия. Одна лицензия = один работающий модуль.
- Модуль анализа безопасности цепочек поставок ПО (SCS) (опциональный), может быть установлен совместно с другими модулями или вынесен на отдельный сервер. Может быть развернут на нескольких серверах одновременно (увеличивает пропускную способность установки по SCS сканированию).

6.1.2. Инструкция по установке системы

Первым на сервер/сервера всегда устанавливается модуль ENV, затем обязательно следует установить на один сервер модуль веб приложения - APP. Опциональные модули (SCS) устанавливаются последними, порядок установки не важен.

Установка ENV модуля

1. Скачать и распаковать архив.
2. Открыть в терминале директорию распакованного архива.
3. Выполнить команду:

   sudo bash actions.sh install | sudo tee -a /tmp/appscreener_ENV.log

   и следовать инструкциям.

Установка APP модуля

1. Скачать и распаковать архив.
2. Открыть в терминале директорию распакованного архива.
3. Выполнить команду:

   sudo bash actions.sh | sudo tee -a /tmp/appscreener_APP.log

   и следовать инструкциям.

Установка модуля SCS

1. Скачать и распаковать архив (архивы).
2. Открыть в терминале директорию распакованного архива.
3. Выполнить команду:

   SCS модуль: sudo bash actions.sh | sudo tee -a /tmp/appscreener_SCS.log

   и следовать инструкциям.

4. При установке модуля на сервер, отличный от сервера модуля APP, в файле /opt/appscreener/core/sca/configs/daemon.env заменить значение поля **queueURI** на адрес сервера с установленным модулем APP, например: queueURI=tcp://10.10.10.10:61616 После чего перезапустить службу модуля командой: sudo systemctl restart appscreener-scs.service

Дальнейшая установка и настройка системы

1. На сервере с развернутым APP модулем в браузере перейти по адресу - `http://localhost` , или на рабочей станции в сети сервера в браузере открыть адрес - `http://<APP_module_installation_address>` .
2. Во всплывающем окне с предложением загрузить лицензию скопировать идентификатор установки и отправить его по почте для создания лицензии под конкретную инсталляцию.
3. Загрузить полученную лицензию через интерфейс.
4. Зайти в систему: логин: **admin**, пароль: ***put_admin_password_here***.
5. Смените пароль учётной записи в **Личном кабинете**.
6. Загрузить файл **Rules.zip** (находится в архиве с APP модулем). Войти в систему как администратор, на вкладке **Администрирование** > **Настройки системы** > **Правила** выбрать и загрузить файл Rules.zip.

Руководство по работе с системой можно загрузить из интерфейса, на вкладке **О продукте**.

6.2. Вход в систему

Для входа в веб-интерфейс модуля SCS (далее UI) введите в адресной строке браузера адрес **http://<host>**, где host – адрес сервера, на который был установлен модуль, или адрес сервера на котором установлен модуль веб приложения (APP.module), в случае если установка проводилась на нескольких серверах. Появится окно авторизации (рис. 6.1).

Для входа в систему введите логин, пароль и нажмите **Войти**.

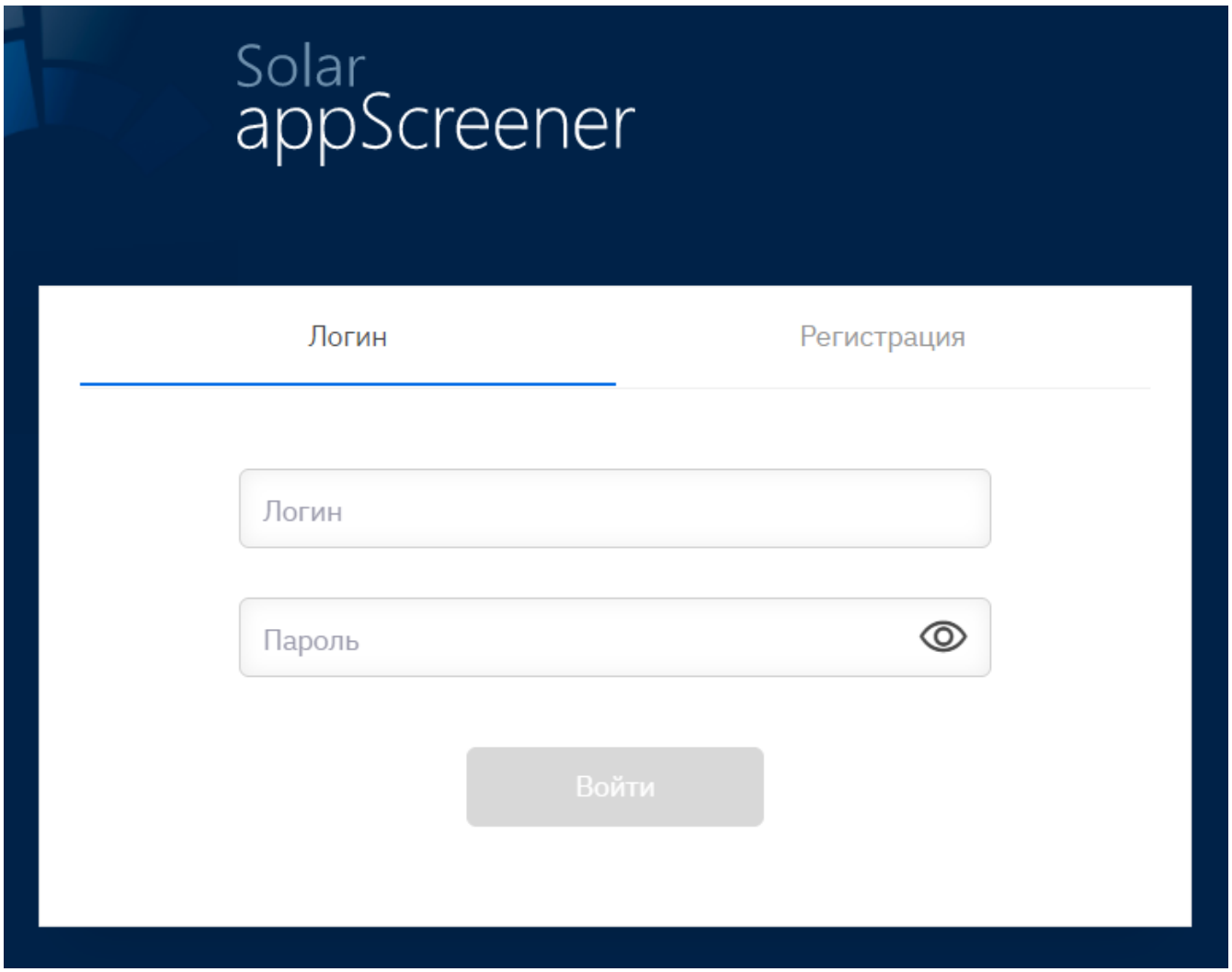

Рис. 6.1: Авторизация пользователя

При введении неверных идентификационных данных на экране отобразится сообщение **Неверный логин и/или пароль**.

Прежде чем начать использование модуля SCS, необходимо ознакомиться с Пользовательским соглашением и нажать кнопку **Принимаю** (рис. 6.2).

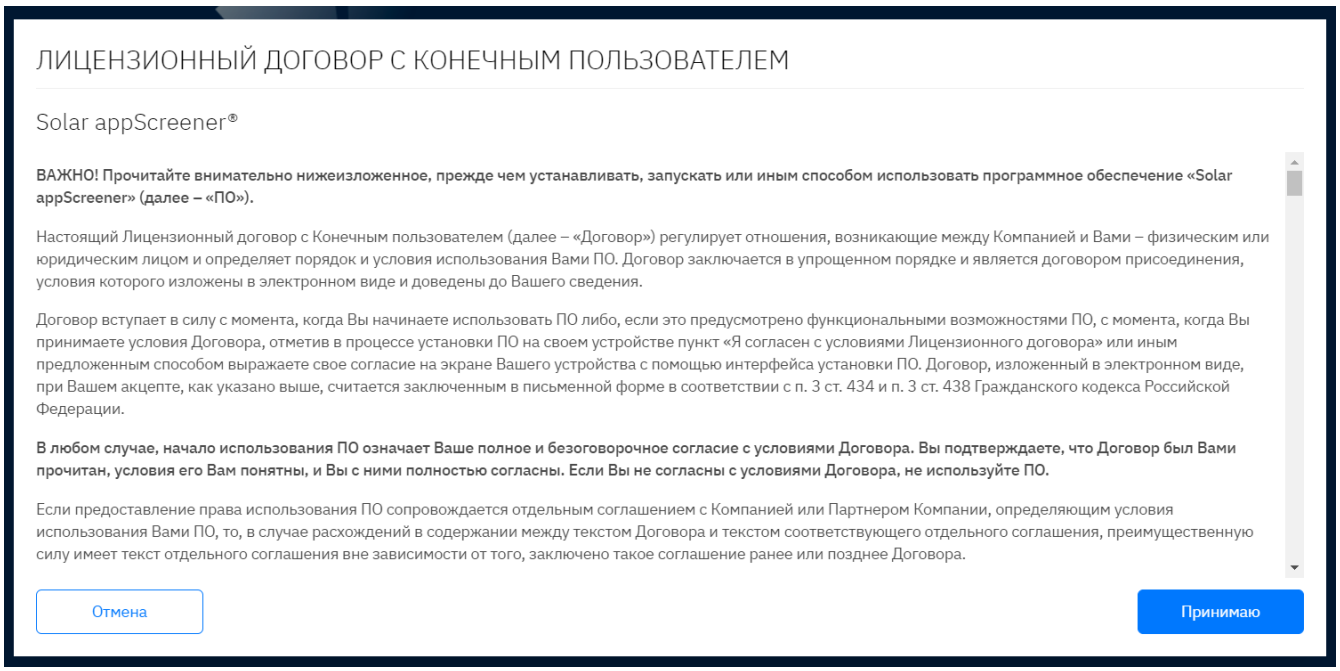

Рис. 6.2: Пользовательское соглашение

После успешного входа в систему отображается **Домашняя страница**.

При наличии созданных проектов **Домашняя страница** будет содержать до 4-х последних запущенных сканирований в проектах.

6.3. Управление учётными записями пользователей

В модуле анализа безопасности цепочек поставок ПО (SCS) используется механизм ролевого разграничения доступа. За каждой ролью закреплен набор доступных пользователю (группе пользователей) действий в системе.

В модуле анализа безопасности цепочек поставок ПО (SCS) реализованы 2 роли: администратор и пользователь по умолчанию. Права пользователя по умолчанию можно настроить. **Администратор** — пользователь с максимальными привилегиями. Ему доступны:

- раздел **Администрирование**;
- все проекты с максимально доступными правами;
- все доступные в лицензии возможности анализа;
- все приватные сущности, созданные другими пользователями.

Также к его функциям относится управление учётными записями пользователей.

Пользователь по умолчанию имеет стандартный набор прав, позволяющих выполнять анализ и проводить верификацию результатов. Опция **Настраиваемые права** позволяет гибко настроить профиль пользователя, указывая необходимые ограничения или наоборот выдавая дополнительные доступы к функциональности модуля SCS. По умолчанию, пользователи имеют доступ только к тем проектам, которые они создали. Доступ к существующим проектам в системе нужно выдавать отдельно.

6.3.1. Управление пользователями

6.3.1.1. Создание учётной записи пользователя

Для создания пользователя:

1. Перейдите в раздел **Пользователи > Создать пользователя**.
2. Введите логин, пароль и ФИО.
3. Введите e-mail, название организации, веб-сайт организации, должность и телефон (опционально).
4. Поля **Учётная запись действительна с/до** позволяют регулировать срок доступа пользователя к системе. До наступления/после завершения срока пользователь не сможет совершить вход в систему, при этом его учётная запись будет доступна администратору. По желанию, администратор сможет изменить срок доступа.
5. Поле **Доступно Supply chain сканирований** позволяет регулировать количество доступных для пользователя сканирований соответствующего типа. При отсутствии доступных сканирований пользователь также не сможет создать пустой проект.
6. Выберите общие права для пользователя:
7. Настройте параметры, которые не будут отображаться для других пользователей в сканированиях, запущенных данным пользователем, в разделе **Ограничения видимости**.
8. Настройте доступ к проектам в системе.
9. Выберите права в проектах или в группах проектов. Права пользователя на проект, на группу проектов и полученные через **группу пользователей** работают по принципу объединения. Если пользователь является автором проекта, он получает все доступные права в проекте (за исключением ограничений, которые могут быть заданы в разделах **Общие права доступа**, **Ограничения видимости**, **Анализировать языки**).
10. Нажмите **Сохранить**.

Для редактирования/удаления пользователя либо редактирования его прав нажмите на логин пользователя в списке, внесите требуемые изменения и нажмите **Сохранить/Удалить пользователя**.

6.3.1.2. Блокировка пользователя

В форме редактирования пользователя предусмотрен механизм ручной блокировки пользователя. Чтобы заблокировать пользователя:

1. Кликните на логин пользователя в списке пользователей.
2. В открывшейся форме редактирования пользователя укажите причину блокировки. Причина отобразится пользователю при попытке входа.
3. Нажмите **Заблокировать**.

После нажатия кнопки **Заблокировать** причину блокировки указать будет нельзя.

Для разблокировки пользователя перейдите в форму редактирования заблокированного пользователя и нажмите **Разблокировать**.

6.3.2. Управление группами

Для удобства назначения ролей пользователи могут быть объединены в группы. Чтобы создать группу пользователей:

1. Перейдите в раздел **Группы пользователей > Создать группу**.
2. Введите имя группы.
3. Добавьте описание группы (опционально).
4. Выберите состав группы из списка пользователей.
5. Выберите общие права и права в проектах.
6. Нажмите **Сохранить**.

Для редактирования/удаления группы нажмите на название группы в списке, внесите требуемые изменения и нажмите **Сохранить**/**Удалить группу**.

6.4. Администрирование системы

В разделе **Администрирование > Настройки системы** можно управлять настройками модуля SCS, загружать правила поиска рисков Supply chain, обновлять лицензию.

6.4.1. Общие настройки

На вкладке **Настройки системы > Общие** можно управлять настройками модуля SCS.

6.4.1.1. Система

В разделе **Система** можно работать со следующими настройками:

- **Максимальный размер загружаемого файла** — в поле можно выставить лимит на размер загружаемого файла в битах.
- **Продолжительность сессии** — определяет время, по истечении которого нужно будет повторно авторизоваться в веб-интерфейсе модуля SCS.
- **Приостановить систему** — для приостановки всех активных сканирований. Прогресс сохраняется.

6.4.1.2. Почта

В разделе **Почта** можно работать со следующими настройками:

- **Администратор** — на указанные здесь электронные адреса будут приходить уведомления о запусках сканирования и сбоях.
- **Обратная связь** — указанные здесь пользователи будут получать отзывы о работе системы.
- **От** — в этом поле можно указать адрес, который будет указан как отправитель.
- **Хост** — хост, используемый на почтовом сервере для подключения.
- **Localhost** — доменное имя почтового сервера.
- **Пароль** — пароль, используемый для аутентификации.
- **Порт** — порт, используемый на почтовом сервере для подключения.

- **SSL** — ssl для почтового сервера, принимает значения true/false.
- **TSL start** — starttls для почтового сервера, принимает значения true/false.
- **Пользователь** — логин, используемый для авторизации.

В разделе также можно настроить системные оповещения. Для оповещений об окончании лицензии и подписки на тех. поддержку можно настроить период и частоту напоминаний, а также выбрать вариант отображения: письма и/или оповещения в интерфейсе.

6.4.1.3. Управление пользователями

Регистрация позволяет управлять настройками:

- **Подтверждение** — активация этого чекбокса добавляет необходимость подтверждения адреса электронной почты при регистрации пользователя модуля SCS через REST.
- **Доступное время работы в системе до подтверждения почты** — в этом поле можно указать время, в течение которого пользователь может пользоваться продуктом без подтверждения.
- **Срок действия** — в этом поле можно указать дни действия учётной записи по умолчанию.
- **Уведомление администратора** — при активации этого чекбокса администратор системы будет получать уведомления на почту о регистрации пользователя через REST.

Права по умолчанию позволяют задать права по умолчанию для пользователей, созданных через вызов API и созданных администратором, а также настроить для них права в проекте.

Ротация пароля позволяет настроить:

- **Ротация пароля** — активация чекбокса включает принудительную смену пароля учётных записей пользователей с определенной периодичностью.
- **Срок действия пароля** — в поле можно указать продолжительность действия пароля.
- **Ежедневно уведомлять пользователя о истечении срока действия пароля** — в поле можно указать срок, в течение которого пользователь будет получать оповещения о необходимости поменять пароль учётной записи.

Блокировка пользователя при превышении лимита попыток входа позволяет настроить:

- **Блокировать пользователя при превышении лимита попыток входа** — активация чекбокса включает блокировку пользователя при превышении попыток входа с неверным паролем.
- **Срок блокировки** — в поле можно указать продолжительность действия блокировки пользователя.
- **Лимит попыток входа** — в поле можно указать лимит попыток авторизации с неверным паролем.

6.4.2. Настройка LDAP

Для настройки **LDAP** (рис. 6.3):

1. Перейдите в **Настройки системы > LDAP**.
2. Нажмите **Добавить подключение**.
3. Укажите **параметры** нового подключения (в тултипе справа от поля можно найти подсказку). Обратите внимание, что в некоторых случаях для AD может требоваться подключение только по доменному имени (не по ip и другим адресам).
   - DNS сервер или `C:\windows\system32\drivers\etc\hosts` содержит адрес, совпадающий с главным доменным именем сертификата.
   - Для авторизации в AD может требоваться доменный префикс в логине.
4. Нажмите **Проверить соединение**, после успешного выполнения нажмите **Подключить**.
5. Синхронизируйте пользователей LDAP. Для синхронизированных пользователей система создаст учётные записи в модуле SCS. Работать с ними можно как с локальными пользователями

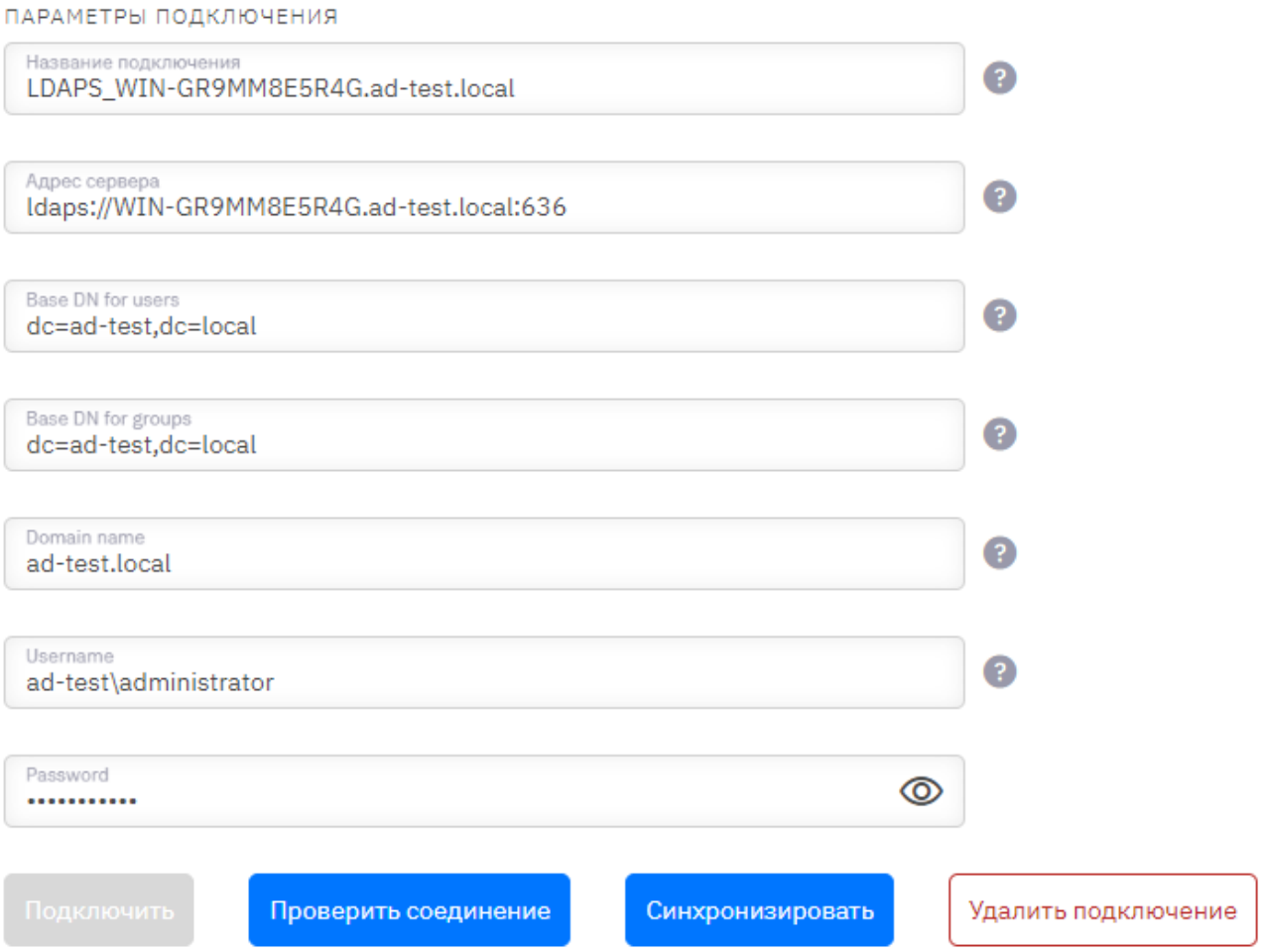

Рис. 6.3: Раздел Настройки системы > LDAP

6.4.3. Лицензия

Для обновления **Лицензии**:

1. Перейдите в **Настройки системы > Лицензия**.
2. Выберите файл с лицензией.

3. Нажмите **Сохранить**.

6.5. Система регистрации событий

Подсистема регистрации событий модуля SCS регистрирует события для каждого модуля (APP, SCS).

Подсистема регистрации событий фиксирует следующие события:

- действия пользователей:
  - запуск, остановка, просмотр результатов и другие действия со сканированиями и проектами;
  - авторизация;
  - действия с правилами;
  - действия администратора;
- технические события системы:
  - состояние запущенных сканирований;
  - состояние работы подключенных модулей анализа;
  - блокировка/разблокировка пользователей;
  - отправка e-mail сообщений;
  - проверка ограничений пользователя;
  - сообщения об ошибках системы;
  - файловые операции.

Все события заносятся в журналы событий.

6.5.1. Журналы событий

Модуль интеграции анализа рисков цепочек поставок ПО (SCS) может создавать несколько файлов журналов событий. Они отличаются уровнем детализации:

- ERROR - события с ошибками системы;
- WARN - события с предупреждениями, которые потенциально могут привести к событиям с ошибкой;
- INFO - события с информацией состояния/действий системы;
- DEBUG - события с отладочными данными для анализа работы системы разработчиками.

Все модули записывают и хранят журналы событий в файловой системе сервера, на котором они функционируют.

Журналы событий APP модуля

В составе веб-приложения (APP модуль) работают нескольких подсистем:

- NGINX - отвечает за хранение каркаса портала и служит reversproxy для обращений извне. Файлы журнала событий nginx находятся на установке системы по пути **/opt/appscreener/app/services/frontend/logs** и состоят из:
  - access.log (журнал обращений клиентов к веб-серверу);
  - error.log (журнал ошибок веб-сервера).

- BACKEND - отвечает за логику работы веб-приложения. Файлы журнала событий подсистемы находятся на сервере по пути **/opt/appscreener/app/services/backend/logs** и состоят из:
  - appscreener-error.log (только события с ошибками);
  - appscreener-warn.log (события с ошибками и предупреждениями);
  - appscreener-debug.log (события с ошибками, предупреждения, информация о работе системы и отладочные данные).
- ARTEMISMQ - отвечает за контроль работы анализаторов (очередь сканирований). Файлы журнала событий ArtemisMQ можно получить на сервере с работающим модулем APP, выполнив команду:

  ```
  sudo docker logs artemis
  ```
- POSTFIX - отвечает за работу почтового сервера, для отправки из системы модуля SCS сообщений. Файлы журнала событий postfix можно получить на сервере с работающим модулем APP, выполнив команду:

  ```
  sudo docker logs postfix
  ```

Журналы событий SCS модуля

Модуль отвечает за работу алгоритмов анализа рисков цепочек поставок ПО. При работе модуля на нескольких серверах, журналы событий будут храниться в файловой системе сервера отдельно для каждого экземпляра модуля. Журналы модуля хранятся в директории **/opt/appscreener/core/scs/services/scs-daemon/logs**.

Все журналы производят ежедневную ротацию и архивирование в аллоцированные директории.

6.6. Резервирование данных

Для резервирования данных системы требуются копии:

1. **Базы данных приложения**. Дамп базы данных можно получить, выполнив на хосте с APP модулем: `sudo docker exec app-db pg_dump -U backend backend > db_dump.sql`
2. **Директорий**:
   - /opt/appscreener/app/services/backend/files/
   - /opt/appscreener/app/services/frontend/nginx/ssl/
3. **Файлов конфигурации**:
   - На сервере с APP модулем:
     - /opt/appscreener/app/configs/backend.env
     - /opt/appscreener/app/configs/postgresql.env
     - /opt/appscreener/app/services/frontend/nginx/templates/default.conf.template
   - На сервере с SCS модулем:
     - /opt/appscreener/core/scs/configs/scs-daemon.env
4. **Логов приложения** (опционально, см. Журналы событий).

7. ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ О РАБОТЕ С МОДУЛЕМ SCS

7.1. Подключение встроенного почтового сервера

В дистрибутив поставки входит почтовый сервер и клиент **Postfix**, который можно использовать как релей для пересылки писем. Клиент использует TLS-протокол для подключения к сторонним почтовым серверам.

Обратите внимание:

- Во внешних сервисах письма могут приходить в папку **Спам**, если явно не указать фильтрацию входящих писем.
- Письма могут приходить с некоторой задержкой (из-за попадания в «Серый список»).
- Журнал событий сервера можно посмотреть с помощью команды, которую необходимо запустить на хосте, где развёрнут **APP** модуль: `sudo docker logs postfix`

Для настройки встроенного почтового сервера:

1. Перейдите в раздел **Администрирование** > **Настройки системы** > **Общие** > **Почта**.
2. Заполните поля **Администратор**, **Обратная связь** и **От**. В поле **Хост** добавьте значение **postfix**, в поле **Порт** — значение **25**, остальные поля оставьте пустыми.

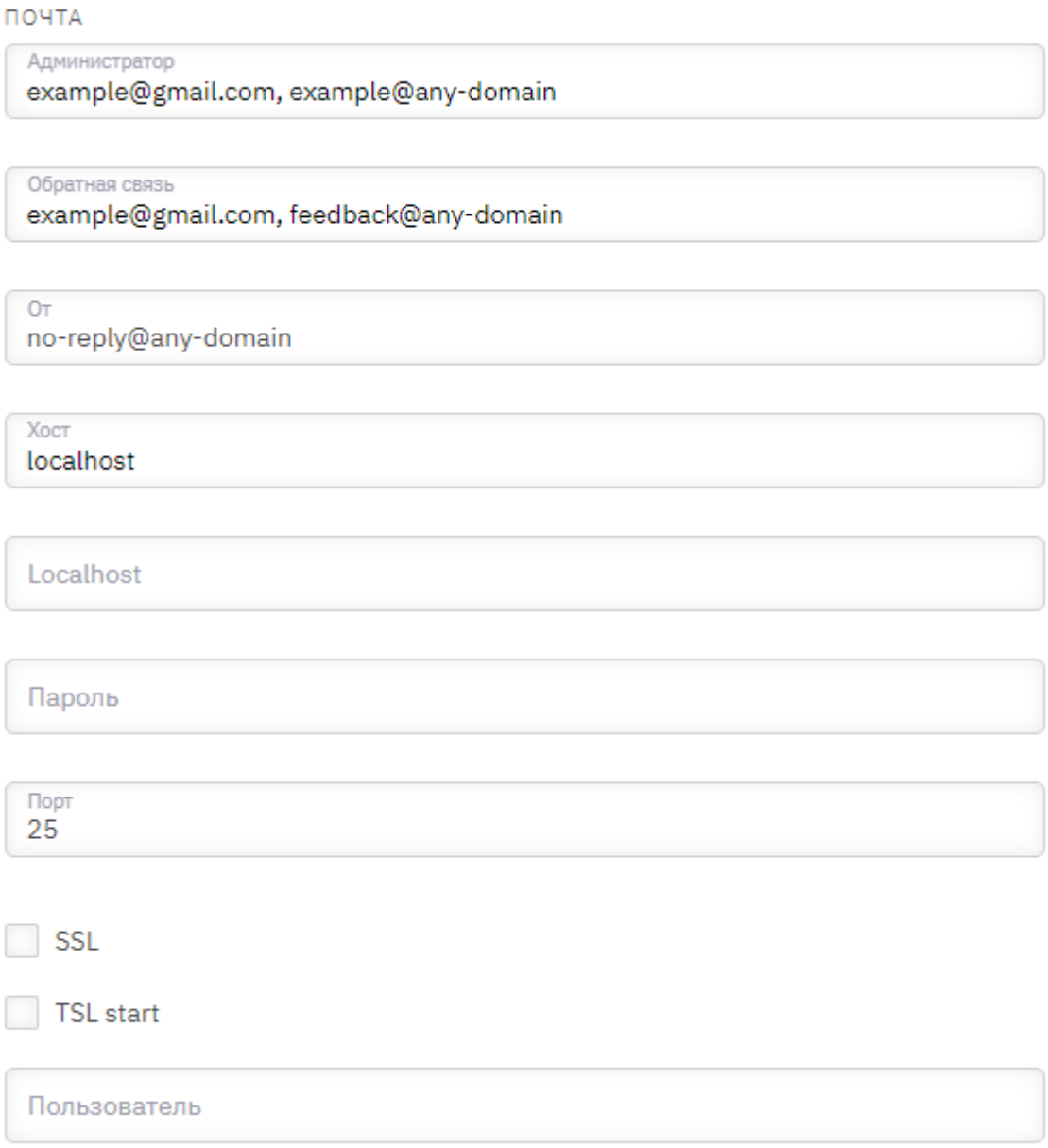

Рис. 7.1: Настройка почтового сервера

3. Настройте политику оповещений в разделе **Администрирование** > **Настройки системы** > **Общие** > **Оповещения**.
4. Нажмите **Сохранить**.

7.2. Добавление самоподписных сертификатов в доверенные для работы через HTTPS и LDAPS

Для добавления сертификата в доверенные необходимо::

1. Загрузить на сервер где развернуто веб-приложение (APP.module) системы, файл сертификата в формате PEM. Этот формат представляет собой ASCII-файл, закодированный по схеме Base64. Возможные расширения: `.pem`, `.crt`, `.cer`. Для примера используется `.cer`-файл сертификата LDAPS: **example.cer**.

2. Выполнить команды:

```
docker cp example.cer backend:/

docker exec backend keytool -importcert -noprompt -cacerts -storepass

changeit -file "/example.cer" -alias "example-cert"
```

3. Проверить наличие только что добавленных сертификатов в списке доверенных:
   `docker exec backend keytool -cacerts -storepass changeit -list | grep examplecert`
4. Остановить процессы сканирования или дождаться их завершения. Перезапустить работу APP модуля:

   sudo systemctl restart appscreener-app.service

7.3. Настройка доступности API извне

Порядок настройки следующий:

1. Подключитесь к хосту, где развернут **APP** модуль.
2. Запустите терминал.
3. Узнайте адрес машины с помощью команды: **ip a**.
4. Откройте конфигурационный файл: **/opt/appscreener/app/configs/backend.env**
5. Измените значения полей на следующие (HTTP/HTTPS в зависимости от того, какой протокол используется):
   - installation.externalBackendAddress=http://frontend/app/
   - installation.externalFrontendAddress=http://frontend/
6. Сохраните изменения.
7. Перезапустите сервис **APP** модуля: **sudo systemctl restart appscreener-app.service**

7.4. Увеличение памяти для сервиса Tomcat

Чтобы увеличить объём памяти для сервиса Tomcat:

1. На хосте с модулем **APP** откройте конфигурационный файл: **/opt/appscreener/app/configs/backend.env**
2. В следующей строке замените значение `Xmx4096M` на `Xmx8192M`:
   `CATALINA_OPTS="-Xms1024M -Xmx4096M"` Должно получиться:
   `CATALINA_OPTS="-Xms1024M -Xmx8192M"`
3. Выйдите с помощью **Ctrl+x**, сохранив изменения.
4. Перезапустите модуль **APP**: **sudo systemctl restart appscreener-app.service**

7.5. Что делать, если сканирование завершилось со статусом «Ошибка»?

При возникновении ошибок во время сканирования обратитесь в службу поддержки. К письму приложите скриншот страницы сканирования с описанием ошибок по языкам программирования. Если ошибок несколько, то приложите несколько скриншотов.

В более сложных случаях для диагностики ошибок необходимы файлы журнала событий системы:

```
/opt/appscreener/core/sca/services/sca-daemon/logs
/opt/appscreener/app/services/backend/logs
/opt/appscreener/app/services/frontend/logs
```

7.6. Миграция данных Solar appScreener модуль анализа безопасности цепочек поставок ПО (SCS)

7.6.1. Миграция сервера на другой хост

Для миграции сервера модуля SCS с **Host_1** на **Host_2** необходимо:

1. На Host_1 выполнить:

   ```
   sudo docker exec app-db pg_dump -U backend backend > db_dump.sql
   ```

2. Развернуть новую установку модуля SCS на Host_2.
3. Перенести дамп базы на Host_2.
4. Перенести директорию **/opt/appscreener/app/services/backend/files/** с Host_1 на Host_2 по тому же пути.
5. На Host_2 выполнить:

   sudo systemctl stop appscreener-app.service

   sudo docker volume rm -f app-postgres-data

   sudo docker compose -f /opt/appscreener/app/app.compose.yml up -d app-db

   sudo docker cp db_dump.sql app-db:/

   (*выполнить команду из директории, где находится дамп базы данных*)

   sudo docker exec app-db bash -c “psql -U backend backend < db_dump.sql”

   sudo systemctl start appscreener-app.service

7.6.2. Миграция сервера с Windows на Linux

Чтобы выполнить миграцию сервера модуля SCS с Windows на Linux, необходимо:

На **Windows_host**:

1. Завершить или дождаться завершения всех активных сканирований.
2. В домашней директории открыть файл **pg_dump** в PowerShell:

   C:\appscreener\3rd-party\PostgreSQL\13\bin\pg_dump -E UTF-8 -f C:\appscreener\db_dump.sql -U backend backend

3. При запросе пароля, ввести его при подключении к БД (расположен в **Environment Variables > System variables > hibernate.connection.password**).

4. Перенести на **Linux_host** следующие файлы и папки:
   - **C:\appscreener\db_dump.sql**
   - **C:\appscreener\files\b**
   - **C:\appscreener\files\s**

На **Linux_host**:

1. Установить систему модуля SCS.
2. Копировать с **Windows_host** на **Linux_host** файлы и папки для миграции в удобную директорию.
3. Остановить работу веб-приложения:

   sudo docker stop backend
4. Удалить БД новой установки:

   sudo docker exec app-db dropdb -p 5432 -h localhost -U backend –maintenance-db=postgres -f -e backend
5. Создать пустую БД:

   sudo docker exec app-db createdb -p 5432 -h localhost -U backend backend
6. Перейти в директорию с файлами для миграции (**db_dump.sql, /b, /s**) и провести миграцию файлов из дампа:

   sudo cat db_dump.sql | sudo docker exec -i app-db psql -p 5432 -h localhost -U backend -d backend
7. Провести копирование директорий **b** и **s**:

   sudo cp -r s/ /opt/appscreener/app/services/backend/files

   sudo cp -r b/ /opt/appscreener/app/services/backend/files
8. Перезапустить службу приложения **APP**:

   sudo systemctl restart appscreener-app.service

7.6.3. Миграция базы данных на другой хост

Для миграции базы данных модуля SCS на новый хост необходимо:

На хосте с модулем **APP**:

1. Завершить или дождаться завершения всех активных сканирований.
2. Остановить сервис приложения:

   sudo systemctl stop appscreener-app
3. Сделать дамп базы данных:

   sudo docker exec app-db pg_dump -U backend backend > db_dump.sql
4. Запомнить или записать значение поля *hibernate.connection.password* в **/opt/appscreener/app/configs/backend.env**.

На хосте, где вы хотите развернуть СУБД **PostgreSQL**:

5. Установить СУБД PostgreSQL версии не ниже 13.0.
6. Создать в БД пользователя с именем *backend* и паролем из поля *hibernate.connection.password*.
7. Создать пустую базу данных с именем *backend* от пользователя с именем *backend*.
   - *для хоста*: **sudo createdb -p 5432 -h localhost -U backend backend** (может потребоваться пользователь postgres, **sudo su - posrgres**)
   - *для Docker*: **sudo docker exec createdb -p 5432 -h localhost -U backend backend**
8. Выполнить загрузку дампа БД (*db_dump.sql*) в пустую базу:
   - *для хоста*: **sudo cat db_dump.sql sudo psql -p 5432 -h localhost -U backend -d backend** (может потребоваться пользователь postgres, **sudo su - posrgres**)
   - *для Docker*: **sudo cat db_dump.sql sudo docker exec -i psql -p 5432 -h localhost -U backend -d backend**
9. Сделать подключение к серверу доступным извне.

На хосте с модулем **APP**:

10. Открыть файл **/opt/appscreener/app/configs/backend.env** для редактирования.
11. Изменить **hibernate.connection.url=jdbc:postgresql:/app-db:5432/backend?ssl=falsee**, изменив имя подключения к контейнеру **app-db** на адрес сервера с развернутой БД, например: **hibernate.connection.url=jdbc:postgresql:/####.###.#.####:####/backend?ssl=falsee**.
12. Сохранить изменения и запустить сервис приложения:

    sudo systemctl start appscreener-app

8. ПОЛУЧЕНИЕ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

Для получения консультации по техническим вопросам можно обратиться по адресу support.appscreener@rt-solar.ru.

С условиями поддержки можно ознакомиться на сайте компании Solar Security. При оформлении запроса следует указать номер контракта на техническую поддержку, описать проблему, указать свое полное имя, адрес электронной почты и номер телефона.