2024

Система Эгида-Аргус

Инструкция по установке

ООО «ИнфоЗонт»
https://infozont-group.ru
+7 (962) 577-24-67

Обозначения и сокращения

АБИ – Администратор безопасности информации
АРМ – Автоматизированное рабочее место
АС – Администратор системы
ИБ – Информационная безопасность
МЭ – Межсетевое экранирование
ОС – Операционная система
ПО – Программное обеспечение
ПЗУД – Подсистема защищённого удалённого доступа
ПОС – Подсистема оценки состояния(Эгида-Аргус)
СА – Системный администратор
СУБД – Система управления базами данных
УРМ – Удаленное рабочее место
ACL – Access Control List
AD – Active Directory
DNS – Domain Name System
HTTPS – HyperText Transfer Protocol Secure
IP – Internet Protocol
VPN – Virtual Private Network

Содержание

Оглавление

Система Эгида и модуль Эгида-Аргус....1
Инструкция по установке....1
Обозначения и сокращения....2
Содержание....3
1. Системные требования....4
1.1 Требования для системы Эгида....4
1.1.1 Аппаратные требования для серверной части....4
1.1.2 Требования к ПО и пакетам....4
1.1.3 Сетевая конфигурация....5
1.1.3.1 IP-адреса и hostname....5
1.1.3.2 Доступные порты....5
1.2. Требования для сервера СУБД....6
1.2.1 Параметры аппаратной конфигурации и устанавливаемой операционной системы....6
1.2.2 Требования к программному обеспечению и пакетам....7
....7
1.2.3 Сетевая конфигурация....7
2. Установка и конфигурация....7
2.1 Установка серверной части приложения....8
2.1.1 Общие положения по установке компонентов системы....8
2.1.2 Общие положения по установке СУБД и конфигурации БД....9
2.2. Установка и настройка компонента API....9
2.2.1. Начальные требования....9
2.2.2. Установка....10
2.2.3. Конфигурация systemd....11
2.3. Установка и настройка компонента ACMS....11
2.3.1. Начальные требования....11
2.3.2. Установка....12
2.3.3. Конфигурация systemd....13
2.4. Установка и настройка компонента Orchestratrion service....13
2.4.1. Начальные требования....13
2.4.2. Установка....14
2.4.3. Конфигурация systemd....15
2.5. Конфигурация веб-сервера для функционала API Gateway по маршрутизации запросов к компонентам системы (nginx)....15
2.5.1. Начальные требования....15
2.5.2. Конфигурация....15

2.6. Конфигурация веб-сервера для предоставления файлов веб-приложения (nginx).....16
2.6.1. Начальные требования.....16
2.6.2. Конфигурация.....16

1. Системные требования

1.1 Требования для системы Эгида

1.1.1 Аппаратные требования для серверной части

CPU – 2 процессора имеющие по 2 или более потков выполнения с тактовой частотой работы 2 GHz или более
RAM – 8 Gb
Disk – HDD объёмом не менее 150 Gb*
OS – Debian 11

* требования по объёму диска определены из соображений сохранения резервных копий, критически важных баз данных, используемых в системе.

1.1.2 Требования к ПО и пакетам

Nginx – последней актуальной стабильной версии
ASP.NET Core runtime – версия 6.* (актуальной минорной версии)
SSH Server – для обслуживания программного комплекса (согласно внутренним регламентам устанавливаемого ПО)

1.1.3 Сетевая конфигурация

1.1.3.1 IP-адреса и hostname

Для корректной работы системы, необходимо, чтобы хостовому устройству (виртуальному или физическому) будет выделен статический IP адрес и/или создана запись в используемом DNS, через которую можно будет получить таковой (например, egida.contoso.com).
Внутри программного комплекса можно выделить два контура слабо друг с другом связанных:

- Доступ к UI администратора и API управления
- Доступ для работы агентов автоматической оценки соответствия к серверной части системы

Данные контуры могут быть разделены по принадлежности к различным подсетям (для обеспечения работы агентской сети и для администрирования и получения данных из системы).

1.1.3.2 Доступные порты

В целях обеспечения безопасности работы сервиса, рекомендуется настроить firewall таким образом, чтобы необходимые для работы входящие сетевые подключения были разрешены(см. ниже).

При наличии внутреннего регламента сетевых настроек для целевой ОС они так же должны быть применены, если не конфликтуют с конфигурацией требуемой для работы системы.

Для функционирования и обслуживания системы необходимо открыть на уровне firewall следующие порты:

- Порт подключения агентов к серверной части системы 11540 по протоколу TCP
- Порт для работы web-интерфейса системы (по умолчанию 80 или 443) по протоколу TCP*
- Порт для управления сервером по протоколу SSH (по умолчанию порт 22) по протоколу TCP

* Для использования порта 443 необходим выпуск TLS-сертификата для соблюдения общепринятых конвенций по IPC. Настоятельно рекомендуется выпуск TLS-сертифката для обеспечения защиты данных в транзите при использовании web-UI системы и/или API.

1.2. Требования для сервера СУБД

1.2.1 Параметры аппаратной конфигурации и устанавливаемой операционной системы

CPU - Процессор имеющий 4 или более потоков выполнения с тактовой частотой работы 2 GHz или более
RAM - 8 Gb
HDD - с доступным объёмом не менее 300Gb*
OS - Debian 11

* Требования по объёму диска определены из расчёта асимптотического ожидаемого объёма сохраняемых данных в 5Gb в течение месяца на основе определённых ниже предположений о параметрах работы системы и глубиной архива на 3-3,5 года:

- В системе зарегистрировано для контроля оценки соответствия 50,000 узлов при среднем назначении в 1000 правил на хост с периодичностью выполнения для каждого «один раз в календарную неделю» (~4Gb в месяц).

- В системе фиксируется раз в сутки состояние контроллера(ов) домена суммарно включающих в себя 50,000 компьютеров с периодичностью фиксации состояний раз в 24 часа (~1 Gb месяц).

- Остальные параметры оказывают значительно меньшее влияние на занимаемое дисковое пространство базами данных системы и асимптотически на них заложено ~20 Gb на обозначенный срок функционирования системы.

1.2.2 Требования к программному обеспечению и пакетам

Для обеспечения работы системы и её дальнейшей поддержки на сервере должно быть установлено следующее программное обеспечение:

- PostgresSQL не ниже 13-й мажорной версии

1.2.3 Сетевая конфигурация

Для корректной работы системы, необходимо, чтобы хостовому устройству (виртуальному или физическому) будет выделен статический IP адрес и/или создана запись в используемом DNS, через которую можно будет получить таковой (например dbms-egida.contoso.com).

2. Установка и конфигурация

Установка и настройка программных средств осуществляется в следующем порядке:

1. Установка серверной части приложения(включая СУБД).
2. Установка и настройка компонента API.
3. Установка и настройка компонента ACMS.
4. Установка и настройка компонента Orchestratrion service.
5. Конфигурация веб-сервера для функционала API Gateway по маршрутизации запросов к компонентам системы (nginx).
6. Конфигурация веб-сервера для предоставления файлов веб-приложения (nginx).

2.1 Установка серверной части приложения

2.1.1 Общие положения по установке компонентов системы

Дистрибуция компонентов системы осуществляется путём **передачи бинарных файлов или файлов документов отображаемым веб-браузерами**.

Для сегрегации доступов приложений внутри хостовой операционной системы рекомендуется создавать отдельного пользователя с назначением ему соответствующих доступов.

В соответствии с этим, существует рекомендация по используемым путям развёртывания, которая выглядит следующим образом:

1. В качестве корневой директории используется **/opt/egida/** (далее **%egida_root%**).

2. Внутри корневого каталога, упоминаемого в пункте **1,** идёт разделение на следующие директории:

   2.1. **%egida_root%/bin** - используется для размещения бинарных файлов компонентов системы. Для каждого из устанавливаемых компонентов системы внутри данного каталога создаётся директория по имени компонента (например, api) (далее %bin%).

   2.2. **%egida_root%/etc** – используется для размещения файлов конфигурации компонентов системы. Для каждого из устанавливаемых компонентов системы внутри данного каталога создаётся директория по имени компонента (например, **api**) (далее **%etc%**).

3. **%egida_root%/varwww** – используется для размещения файлов, связанных с управлением системой через веб-браузер:

   3.1. **%egida_root%/varwww/spa** - используется для хранения файлов веб-приложения, отображаемых веб-браузером (далее **%varwww-spa%**).

   3.2. **%egida_root%/varwww/attachments** – используется для сохранения файлов-вложений, сохранённых в системе через веб-интерфейс (далее **%varwww-attachments%**).

4. В качестве корневого каталога для файлов логов компонентов системы используется /**var/log/egida**. Для каждого из компонентов системы выделяется отдельная директория внутри корневого каталога логов по имени компонента (например, **api**) (далее **%logs%**).

В случае, если в компании существует соглашение по дистрибуции и/или развёртыванию похожих систем, параметры каталогов могут быть изменены в соответствии с данным регламентом.

Большая часть серверных компонентов системы развёртывается при помощи системы управления службами в Linux systemd. Это позволяет настроить изоляцию переменных окружения для компонентов системы.

2.1.2 Общие положения по установке СУБД и конфигурации БД.

Все серверные компоненты в качестве системы управления базами данных используют решение PostgreSQL и рассчитаны на 13 мажорную версию и выше.

Конфигурация СУБД и БД должна осуществляться в соответствии с внутренними регламентами по администрированию (создание резервных копий, управление используемыми дисковыми пространствами) с учётом следующих рекомендаций:

1. Для каждого компонента системы должные быть создавать отдельные пользователи для обеспечения сегрегации доступов.

2. Должны быть прописаны правила подключения пользователей к базам данных с учётом сегрегации доступов (для каждого из компонентов выделяется отдельный пользователь, и для него должны быть предоставлен доступ на подключение только к тем БД, которые он использует).

3. В случае, если СУБД находится на отдельном сервере от компонентов серверной инфраструктуры (для взаимодействия необходимо коммуникация по сети), рекомендуется настроить шифрование коммуникации с СУБД при помощи TLS (в том числе ограничить способы подключения для пользователей).

Остальные параметры СУБД и баз данных могут быть скорректированы на основании особенностей работы отдельных компонентов системы в рабочей инфраструктуре.

2.2. Установка и настройка компонента API

2.2.1. Начальные требования

1. Установлен ASP.NET Core runtime 6-ой мажорной и наиболее высокой минорной версии на момент установки.

2. В СУБД создан пользователь для компонента (например, **egida_api_user**) и известны его параметры (**имя**, **пароль**, **параметры целевого экземпляра СУБД**).

3. Созданы необходимые БД в СУБД, упоминаемой в пункте 2:

    - БД с основной конфигурацией системы (например, **egida_global**), владельцем которой является пользователь из пункта 2;

    - БД содержащей отчёты оценки соответствия (например, **egida_reports**), владельцем которой является пользователь из пункта 2;

    - БД содержащей конфигурацию модуля мониторинга сети (например, **egida_networkanalysys**), владельцем которой является пользователь из пункта 2;

    - БД, содержащей конфигурацию модуля мониторинга контроллеров домена (например, **egida_ldap**), владельцем которой является пользователь из пункта 2.

4. В СУБД настроены политики подключения пользователя из пункта 2 к базам данных из пункта 3.

5. Определены целевые директории размещения бинарных файлов, файлов конфигурации и логов.

6. Определена директория для размещения вложений, сохраняемых в системе при помощи веб-интерфейса управления.

2.2.2. Установка

1. Копирование бинарных файлов компонента в директорию **%bin%** и установка соответствующих прав.

2. Копирование файлов конфигурации компонента в директорию **%etc%/api/** и установка соответствующих прав.

3. Создание директории **%logs%/api/** для файлов логов компонента и установка соответствующих прав.

4. Актуализация параметров работы модуля **systemd** в файле конфигурации (пример находится в директории **/appendix/etc/api/egida_api.service**):

- Пользователь и группа, под которыми должен работать модуль компонента (параметры **User** и **Group** соответственно).

- Рабочая директория должна быть установлена в директорию, упоминаемой в шаге **1** (параметр **WorkingDirectory**).

- Файл переменных окружения должен быть установлен в соответствии с расположением в директории, упоминаемой на шаге **2** (параметр **EnvrionmentFile**).

5. Актуализация параметров, хранящихся в файле переменных окружения для модуля **systemd** (пример находится в директории **/appendix/etc/api/egida_api.env**):

- Параметры подключения к БД, содержащей основную конфигурации системы (параметр **ConnectionStrrings__global**).

- Параметры подключения к БД, содержащей отчёты оценки соответствия (параметр **ConnectionStrrings__reports**).

- Параметры подключения к БД, содержащей конфигурацию модуля мониторинга сети (параметр **ConnectionStrrings__NetworkAnalysys**).

- Параметры подключения к БД, содержащей конфигурацию модуля мониторинга контроллеров домена (параметр **Ldap__ConnectionStrrings__main**).

- Параметры путей файлов логирования, в частности путь и имя файла внутреннего логирования и корневой директории файлов логов компонента, упоминаемой в шаге **3**, и (параметры **NLog__internalLogFile** и **NLog__variables__logDirectory** соответственно).

  Примечание: файл внутреннего логирования рекомендуется держать в директории логов компонента, хотя это и не обязательно.

2.2.3. Конфигурация systemd

1. Создание файла-ссылки с настройкой модуля в **systemd** в директории **/etc/systemd/system/**. Сделать это можно при помощи следующей команды:
   `ln -s %etc%/ api /egida_api.service /etc/systemd/system/ egida_api.service`

2. Активация модуля путём выполнения следующей команды:
   `systemctl daemon-reload && systemctl enable egida_api.service`

3. Запуск модуля путём выполнения следующей команды:
   systemctl start egida_ api.service

4. Проверка статуса модуля путём выполнения следующей команды:
   systemctl status egida_ api.service

2.3. Установка и настройка компонента ACMS

2.3.1. Начальные требования

1. Установлен ASP.NET Core runtime 6-ой мажорной и наиболее высокой минорной версии на момент установки.

2. В СУБД создан пользователь для компонента (например **egida_acms_user**) и известны параметры для его подключения (имя, пароль, параметры целевого экземпляра СУБД).

3. Созданы необходимые БД в СУБД, упоминаемой в пункте 2.

4. база данных с основной конфигурацией модуля управлением доступа УРМ на основании оценки соответствия (например **egida_acms**), владельцем которой является пользователь из пункта 2.

5. В СУБД настроены политики подключения пользователя из пункта 2 к базам данных из пункта 3.

6. Определены целевые директории размещения бинарных файлов, файлов конфигурации и логов.

2.3.2. Установка

1. Копирование бинарных файлов компонента в целевую директорию и установка соответствующих прав.

2. Копирование файлов конфигурации компонента в целевую директорию и установка соответствующих прав.

3. Создание директории для файлов логов компонента и установка соответствующих прав.

4. Актуализация параметров работы модуля systemd в файле конфигурации (пример находится в директории **/appendix/etc/amcs /egida_acms.service**):

   - Пользователь и группа, под которыми должен работать модуль компонента (параметры **User** и **Group** соответственно).

- Рабочая директория должна быть установлена в директорию, упоминаемой в шаге **1** (параметр **WorkingDirectory**).

- Файл переменных окружения должен быть установлен в соответствии с расположением в директории, упоминаемой на шаге **2** (параметр **EnvrionmentFile**).

5. Актуализация параметров, хранящихся в файле переменных окружения для модуля systemd (пример находится в директории **/appendix/etc/amcs /egida_acms.env**):

   - Параметры подключения к БД, содержащей основную конфигурации системы (параметр **Persistence__ConnectionStrrings__main**).

   - Параметры путей файлов логирования, в частности путь и имя файла внутреннего логирования и корневой директории файлов логов компонента, упоминаемой в шаге 3, и (параметры **NLog__internalLogFile** и **NLog__variables__logDirectory** соответственно).

     Примечание: файл внутреннего логирования рекомендуется держать в директории логов компонента, хотя это и не обязательно

2.3.3. Конфигурация systemd

1. Создание файла-ссылки с настройкой модуля в systemd в директории **/etc/systemd/system/** Сделать это можно при помощи следующей команды:
   ```
   ln -s %etc%/ acms /egida_acms.service /etc/systemd/system/ egida_acms.service
   ```

2. Активация модуля путём выполнения следующей команды:
   ```
   systemctl daemon-reload && systemctl enable egida_ acms.service
   ```

3. Запуск модуля путём выполнения следующей команды:
   ```
   systemctl start egida_acms.service
   ```

4. Проверка статуса модуля путём выполнения следующей команды:
   ```
   systemctl status egida_ acms.service
   ```

2.4. Установка и настройка компонента Orchestratrion service

2.4.1. Начальные требования

1. Установлен ASP.NET Core runtime 6-ой мажорной и наиболее высокой минорной версии на момент установки.

2. В СУБД создан пользователь для компонента (например **egida_orcherstration_user**) и известны его параметры (имя, пароль, параметры целевого экземпляра СУБД).

3. Созданы необходимые БД в СУБД, упоминаемой в пункте 2:

   - БД с основной конфигурацией системы (например **egida_global**), с доступом на чтение и запись для пользователя из пункта **2**.

   - БД, содержащая отчёты оценки соответствия (например **egida_reports**), с доступом на чтение и запись для пользователя из пункта **2**.

   - БД, содержащая внутренние объекты компонента (например **egida_orchestration**), владельцем которой является пользователь из пункта **2**.

4. В СУБД настроены политики подключения пользователя из пункта **2** к базам данных из пункта **3.**

5. Определены целевые директории размещения бинарных файлов, файлов конфигурации и логов.

6. Определена директория для размещения вложений, сохраняемых в системе при помощи веб-интерфейса управления.

2.4.2. Установка

1. Копирование бинарных файлов компонента в целевую директорию и установка соответствующих прав.

2. Копирование файлов конфигурации компонента в целевую директорию и установка соответствующих прав.

3. Создание директории для файлов логов компонента и установка соответствующих прав.

4. Актуализация параметров работы модуля systemd в файле конфигурации (пример находится в директории **/appendix/etc/orchestration/egida_orchestration.service**):

   - Пользователь и группа, под которыми должен работать модуль компонента (параметры **User** и **Group** соответственно).

   - Рабочая директория должна быть установлена в директорию, упоминаемой в шаге **1** (параметр **WorkingDirectory**).

   - Файл переменных окружения должен быть установлен в соответствии с расположением в директории, упоминаемой на шаге **2** (параметр **EnvrionmentFile**).

5. Актуализация параметров, хранящихся в файле переменных окружения для модуля systemd (пример находится в директории **/appendix/etc/orchestration/egida_orchestration.env**):

- Параметры подключения к БД, содержащей основную конфигурации системы (параметр **ConnectionStrings__global**).

- Параметры подключения к БД, содержащей отчёты оценки соответствия (параметр **ConnectionStrrings__reports**) .

- Параметры подключения к БД, содержащей внутренние объекты службы оркестрации (параметр **ConnectionStrings__orchestration**).

- Параметры путей файлов логирования, в частности путь и имя файла внутреннего логирования и корневой директории файлов логов компонента, упоминаемой в шаге 3, и (параметры **NLog__internalLogFile** и **NLog__variables__logDirectory** соответственно).

  Примечание: файл внутреннего логирования рекомендуется держать в директории логов компонента, хотя это и не обязательно.

2.4.3. Конфигурация systemd

1. Создание файла-ссылки с настройкой модуля в **systemd** в директории **/etc/systemd/system/**. Сделать это можно при помощи следующей команды:
   ```
   ln -s %etc%/ orchestration /egida_orchestration.service /etc/systemd/system/
   egida_orchestration.service
   ```

2. Активация модуля путём выполнения следующей команды:
   ```
   systemctl daemon-reload && systemctl enable egida_orchestration.service
   ```

3. Запуск модуля путём выполнения следующей команды:
   ```
   systemctl start egida_orchestration.service
   ```

2.5. Конфигурация веб-сервера для функционала API Gateway по маршрутизации запросов к компонентам системы (nginx).

2.5.1. Начальные требования

1. Установлен nginx последней стабильной версии.

2. Выделена корневая директория для хранения файлов конфигураций (**%etc%**).

3. Согласован формат обращения к веб-приложению (будет ли выделено ли отдельное DNS имя, будет ли использован TLS сертификат для защиты данных в транзите).

4. Развёрнуты компоненты API и ACMS и известны эндпоинты для взаимодействия с ними.

2.5.2. Конфигурация

1. Копирование файлов конфигурации nginx в директорию **%etc%/nginx/egida_api**

2. Установка прав на чтение для директории **%etc%/nginx/** группе **www-data**

3. Актуализация эндпоинтов компонентов API и ACMS в файле **%etc%/nginx/egida_api/egida_api_backends.conf**

4. В случае, если для API был выделен отдельное DNS имя (например **api.egida.contoso.com**), необходимо актуализировать в файле **%etc%/nginx/egida_api** параметр **server_name**. Например: **server_name api.egida.contoso.com**

5. Актуализация путей сохранения фалов логов доступа сервера nginx в параметрах **access_log** и **error_log**

6. Создание файла-ссылки для импорта конфигурации веб-сервера для проксироваия запросов API к приложениям.
   Сделать это можно при помощи следующей команды:
   ```
   ln -s %etc%/nginx/egida_api/egida_api_gateway.conf
   /etc/nginx/sites-enabled/egida_spa.conf
   ```

7. Применение конфигурации веб-сервера nginx.
   Сделать это можно при помощи следующей команды:
   ```
   /sbin/nginx –s reload
   ```

2.6. Конфигурация веб-сервера для предоставления файлов веб-приложения (nginx)

2.6.1. Начальные требования

1. Установлен nginx последней стабильной версии

2. Выделена корневая директория для хранения файлов конфигураций (**%etc%**)

3. Выделена корневая директория для хранения файлов, связанных с управлением системой через веб-браузер (**%varwww-spa%**)

4. Согласован формат обращения к веб-приложению (будет ли выделено ли отдельное DNS имя, будет ли использован TLS сертификат для защиты данных в транзите)

5. Сконфигурирован API Gateway для маршрутизации запросов к компонентам системы

2.6.2. Конфигурация

1. Копирование файлов, связанных с управлением системой через веб браузер в директорию **%varwww-spa%**

2. Установка прав для директории **%varwww-spa%** прав для чтения группе **www-data**
   Сделать это можно при помощи следующей команды:
   ```
   chown -R root:www-data %varwww-spa% && chmod find %varwww-spa% . -type f -exec
   chmod 644 {} + && find %varwww-spa% -type d -exec chmod 755 {} +
   ```

3. Копирование файлов конфигурации nginx в директорию **%etc%/nginx/**

4. Установка прав для директории **%etc%/nginx/** для чтения для группы **www-data**

5. Установка корня директории предоставляемых файлов (параметр **root**) на директорию **%varwww-spa%**
   Пример: **root /opt/egida/varwww/spa**

6. В случае, если для веб-интерфейса программного комплекса было выделено отдельное DNS имя (например egida.contoso.com), необходимо переопределить имя виртуального сервера в файле **%etc%/nginx/egida_spa.conf** (имя параметра **server_name**).
   Пример: server_name egida.contoso.com

7. Актуализация адресов приложений API для осуществления функции обратного проксирования запросов на API:

- В сегменте **location /api/** в свойство **proxy_pass** необходимо выставить схему и адрес развёртывания API.
  Например:
  ```
  location /api/ {
        proxy_pass https://api-egida.contoso.com;
  }
  ```

- В сегменте l**ocation/odata/** в свойство **proxy_pass** необходимо выставить схему и адрес развёртывания API.
  Например:
  ```
  location /odata/ {
        proxy_pass https://api-egida.contoso.com;
  }
  ```

8. Создание файла-ссылки для импорта конфигурации веб-сервера для передачи статических файлов и проксироваия запросов API к приложениям.
   Сделать это можно при помощи следующей команды:
   ```
   ln -s %etc%/nginx/egida_spa.conf /etc/nginx/sites-enabled/egida_spa.conf
   ```

9. Применение конфигурации веб-сервера nginx.
   Сделать это можно при помощи следующей команды:
   `/sbin/nginx –s reload`