



Система
ИнфраМенеджер
ITSM

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И НАСТРОЙКЕ СИСТЕМЫ ИНФРАМЕНЕДЖЕР 7.0

Для версии 7.0.232 и выше

ОГЛАВЛЕНИЕ

0. Введение.....	3
1. Установка и настройка СУБД PostgreSQL.....	4
1.1. Локали системы.....	4
1.2. Установка PostgreSQL.....	4
1.3. Настройка PostgreSQL	4
2. Установка компонентов Веб-сервера ИМ.....	6
2.1. Установка DOCKER	6
2.2. Копирование комплекта контейнеров на сервер.....	6
2.3. Установка/переустановка контейнеров под Linux.....	7
3. Административный интерфейс и подключение к базе данных	11
3.1. Создание базы данных.....	11
3.2. Настройка подключения ИнфраМенеджер к базе данных.....	13
4. Аутентификация при работе с WEB–интерфейсом.....	18
5. Часто задаваемые вопросы и решения	19
5.1. Не получается подключить базу в ИнфраМенеджер 7.0.....	19



0. Введение

Перед началом установки ИнфраМенеджера 7.0 рекомендуем ознакомиться с разделом «Технические требования» на нашем сайте (<https://www.inframanager.ru/download/technical-requirements/>).

В данной инструкции описывается установка ИнфраМенеджера 7.0 на Linux на базе Debian и Ubuntu (в том числе Astralinux) с использованием postgresql 14.

Для установки потребуется доступ к терминалу сервера (скорее всего по SSH, 22 порт), а также доступ по протоколам HTTP (порт 80) и HTTPS (порт 443) для входа в систему (с помощью браузера). Возможно, потребуется доступ по порту 5432 для работы с базой данных через PgAdmin.

Пользователя, из-под которого будет устанавливаться и запускаться ИнфраМенеджер 7.0, будет необходимо добавить в группы docker и sudo (в инструкции это учтено).

Важно обратить внимание, что доступ к системе ИнфраМенеджер 7.0 осуществляется только по протоколу HTTPS – просим Вас учесть это при указании некоторых параметров в ходе установки (в инструкции это учтено).

В ходе установки для организации полноценного доступа к системе потребуется добавить записи в Ваш DNS сервер, а также (при необходимости) выпустить сертификат от Вашего центра сертификации.

В инструкции в качестве примера будет использоваться домен `it.company.example` – поменяйте его на доменное имя вашего хоста.



1. Установка и настройка СУБД PostgreSQL

1.1. Локали системы

ИнфраМенеджер 7.0 рекомендуется устанавливать в ОС с базовой локализацией **en_US.UTF-8**.

Проверить свою текущую локализацию в системе можно командой:

```
locale
```

Если локализация системы отличается, то можно задать локализацию **en_US.UTF-8** с помощью команды:

```
sudo dpkg-reconfigure locales
```

После чего перезапустить систему.

1.2. Установка PostgreSQL

Необходимо установить PostgreSQL 14 – это можно сделать следующей командой:

```
sudo apt install postgresql-14
```

1.3. Настройка PostgreSQL

Для успешного развертывания дампа, в необходимо сделать следующее (**Пункт «а» пропускаем, если выбрали язык системы en_US.UTF-8**):

1) Сконфигурируйте локали в файле конфигурации **«postgresql.conf»**:

```
# These settings are initialized by initdb, but they can be changed.
lc_messages = 'en_US.UTF-8'          # locale for system error message
lc_monetary = 'en_US.UTF-8'         # locale for monetary formatting
lc_numeric = 'en_US.UTF-8'         # locale for number formatting
lc_time = 'en_US.UTF-8'            # locale for time formatting
```

```
lc_messages = 'en_US.UTF-8'
lc_monetary = 'en_US.UTF-8'
lc_numeric = 'en_US.UTF-8'
lc_time = 'en_US.UTF-8'
```

Новая база данных тоже должна создаваться с такой же локализацией.

2) Сконфигурировать подключения в **«postgresql.conf»** (секция Connection settings):

- Раскомментировать (убрать # в начале строки) строку **listen_address** и задать в качестве значения **'*'**
- Раскомментировать строку **port**



```
#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'           # what IP address(es) to listen on;
                                # comma-separated list of addresses;
                                # defaults to 'localhost'; use '*' for all
                                # (change requires restart)
port = 5432                       # (change requires restart)
```

3) Сконфигурировать допустимые подключения в файле "[pg_hba.conf](#)"

В колонке **METHOD** (последний столбец) обязательно надо указывать метод md5.

Пример настроек:

```
# IPv4 local connections:
host      all             all             127.0.0.1/32      md5
host      all             all             172.16.0.0/12     md5
host      all             all             172.168.10.0/24   md5
host      all             all             0.0.0.0/0         md5
```

Какие подсети указаны:

- 127.0.0.1/32 – локальный хост
- 172.16.0.0/12 – диапазон подсетей, который Docker будет использовать для новых контейнеров/сервисов
- 172.168.10.0/24 – подсеть для сервисов ИнфраМенеджера
- 0.0.0.0/0 – любые IP адреса. Вместо неё можно указать корпоративную подсеть.

4) Вручную задать пароль пользователю postgres. Для этого в консоли выполняем команды и вводим пароль:

```
sudo su
su - postgres
psql
\password
задаём пароль и повторяем его
\q
exit (выход из-под пользователя postgres)
exit (выход из-под пользователя root)
```

5) После всех настроек обязательно перезапустить сервис postgresql:

```
sudo systemctl restart postgresql
```

6) Проверьте, что postgresql запустился:

```
systemctl status postgresql
```



2. Установка компонентов Веб-сервера ИМ

2.1. Установка DOCKER

Для установки **Docker** необходимо выполнить следующие команды:

```
sudo apt update
sudo apt install docker.io docker-compose unzip
```

Или с помощью менеджера пакетов установить пакеты **docker.io**, **docker-compose** и **unzip**.

Пользователь, от которого запускается ИнфраМенеджер, должен быть добавлен в группу docker.

Для того, чтобы добавить пользователя в группу `docker`, надо выполнить следующую команду:

```
sudo usermod -aG docker $USER
```

После добавления пользователя в группу перезагружаем сервер для применения изменений.

2.2. Копирование комплекта контейнеров на сервер

При первой установке ПО или обновлении контейнеров необходимо скопировать следующий комплект файлов в рабочую папку (рекомендуется в папку **/opt/inframanager**) на сервере:

Название файла	Описание файла
web-im.tar.gz	Образ контейнера с веб-службой ИМ
admin-im.tar.gz	Образ контейнера с административным интерфейсом ИМ
sd-im.tar.gz	Образ контейнера с веб-интерфейсом Service Desk
we-im.tar.gz	Образ контейнера с сервисом рабочих процедур
search-im.tar.gz	Образ контейнера с сервисом поиска
mail-im.tar.gz	Образ контейнера с почтовым сервисом
telephony-im.tar.gz	Образ контейнера с сервисом телефонии
import-im.tar.gz	Образ контейнера с сервисом импорта
schedule-im.tar.gz	Образ контейнера с сервисом планировщика
nginx-im.tar.gz (или nginx.tar.gz)	Образ контейнера с web-сервером



Название файла	Описание файла
telegrambot-im.tar.gz (при наличии)	Образ контейнера с Telegram ботом (при наличии)
im_run_windows.cmd	Скрипт для запуска контейнеров для Windows
docker-compose-run-win.yml	YML-файл настройки папок с настройками для контейнеров для Windows
Папка LinuxDeploy: <ul style="list-style-type: none"> – .env – docker-compose.yml – docker-compose-volumes.yml – im_run_linux.sh 	В папке собраны файлы, которые нужны для запуска приложения на Linux: <ul style="list-style-type: none"> – Файл для указания настроек для запуска ИМ – YML-файл конфигурации контейнеров ИМ – YML-файл конфигурации папок с настройками для контейнеров ИМ для Linux – Скрипт для запуска контейнеров для Linux

Файлы архивов образов будут находится в папке **images**.

Для установки и запуска приложения ИнфраМенеджер 7.0 используется скрипт [im_run_linux.sh](#), который запускается с `sudo`.

2.3. Установка/переустановка контейнеров под Linux

Для работы ИнфраМенеджер 7.0 на сервере DNS необходимо добавить запись типа A для *im.company.example*

Конфигурирование YML файлов для запуска ИнфраМенеджер 7.0:

Директория с контейнерами должна быть доступна для чтения и записи. В файле **docker-compose-volumes.yml** указаны пути для хранения настроек приложения. По умолчанию настроена та же директория, в которой находится файл **docker-compose-volumes.yml** (`$PWD`), но по желанию эти пути можно переопределить.

В файле **.env** в необходимо определить значение переменной **VAR_APP_SERVER**. Для этого надо после знака равно без пробелов указать полный доменный адрес вашего сервера, обязательно указав в начале `https://`

Пример:

```
VAR_APP_SERVER=https://im.company.example
```



Загрузим docker-образы в репозиторий образов хоста. Для этого перейдите в папку **LinuxDeploy** и выполните следующую команду:

```
sudo bash im_run_linux.sh load
```

После этого начнётся распаковка архивов docker-образов и их добавление в репозиторий образов хоста.

После выполнения можно проверить, что образы добавлены в репозиторий, с помощью следующей команды:

```
docker images
```

Создадим папки для сервисов ИнфраМенеджера 7.0. Для этого запустим и остановим приложение с помощью следующих команд:

```
sudo bash im_run_linux.sh start
```

Эта команда создаст все необходимые папки, создаст и запустит контейнеры, а также запустит скрипты инициализации.

```
sudo bash im_run_linux.sh stop
```

Эта команда остановит и удалит контейнеры, при этом не удалит создавшиеся папки и файлы.

Пути, где будут созданы каталоги, указаны в файле **docker-compose-volumes.yml**

Если у вас дистрибутив демо-версии ИнфраМенеджера 7.0 (с пометкой `_demo`), то в комплекте должен поставляться key-файл (`key.json`). Его необходимо разместить в директории **/путь до ИнфраМенеджер/settings/web/**

ИнфраМенеджер 7.0 работает только по протоколу HTTPS, который требует использование сертификата.

Для добавления сертификата в ИнфраМенеджер 7.0 расположите в директорию **/etc/ssl/im** файл ключа для сертификата и файл сертификата с именами **ssl_key.key** и **ssl_cert.crt**

Самоподписанные сертификаты можно сгенерировать командой:

```
sudo openssl req -x509 -nodes -days 3650 -newkey rsa:2048 -keyout ssl_key.key -out ssl_cert.crt
```

При выпуске сертификатов указывать FQDN *im.company.example*

Если решите выпустить сертификаты от своего центра сертификации, то получившийся сертификат надо потом перевести в формат PEM.

После создания директории **settings** в файле **settings/web/appsettings.Stage.json** укажите адрес вашего сервера в параметре **«AllowCors»** (обязательно с `http://`):




```
"AllowCors": ["http://server.inframanager"],
```

Для работы LDAP аутентификации, в том же файле необходимо изменить следующие настройки (если что, это можно сделать позже):

```
"LDAP": {  
  "LDAPCredentials": [  
    {  
      "LDAPIp": "ldap.domain.local",  
      "LDAPPort": "389",  
      "LDAPSecureSocketLayer": "false",  
      "NetBiosName": "im",  
      "FQDN": "domain.local"  
    }  
  ],  
}
```

Где:

- "LDAPIp" – IP адрес или доменное имя вашего LDAP-сервера
- "LDAPPort" – порт LDAP сервера. Обычно 389
- "NetBiosName" – NetBios имя домена, можно найти в настройках на самом контроллере домена. Оно будет входить в состав имени пользователя при аутентификации, например «im\Administrator»
- "FQDN" – FQDN вашего домена, можно найти в настройках на самом контроллере домена.

Также в другом разделе:

```
"Settings": {  
  "WebSettings": {  
    "LdapAuthentication": true,
```

- "LdapAuthentication": сделать true

Примечание: серверов LDAP можно указать несколько.

Запустим ИнфраМенеджер 7.0 – для этого повторно выполните команду:

```
sudo bash im_run_linux.sh start
```

Проверьте, что контейнеры успешно запущены и работают:

```
docker ps -a
```

Если какие-то контейнеры не запускаются, то посмотреть их логи можно следующей командой:



```
docker logs (имя контейнера)
```



3. Административный интерфейс и подключение к базе данных

3.1. Создание базы данных

Необходимо в вашей СУБД создать пустую базу данных и восстановить в неё дампы базы данных для ИнфраМенеджера 7.0. Для этого в архиве с образцами ИнфраМенеджера 7.0 (обычно по пути Databases/PostgreSQL) лежит 2 файла:

- `im_blank.backup` – файл с дампом чистой базы для ИнфраМенеджера
- `im_demo.backup` – файл с дампом демонстрационной базы для ИнфраМенеджера

Для знакомства с ИнфраМенеджер 7.0 рекомендуем использовать демонстрационную базу данных. В инструкции будет указана именно она (`im_demo.backup`), однако шаги для `im_blank.backup` идентичные. В примере будем создавать базу данных с именем `im_demo_228`.

Сделать все операции можно через `psql` (командная строка) или через `pgAdmin`

Через `psql`:

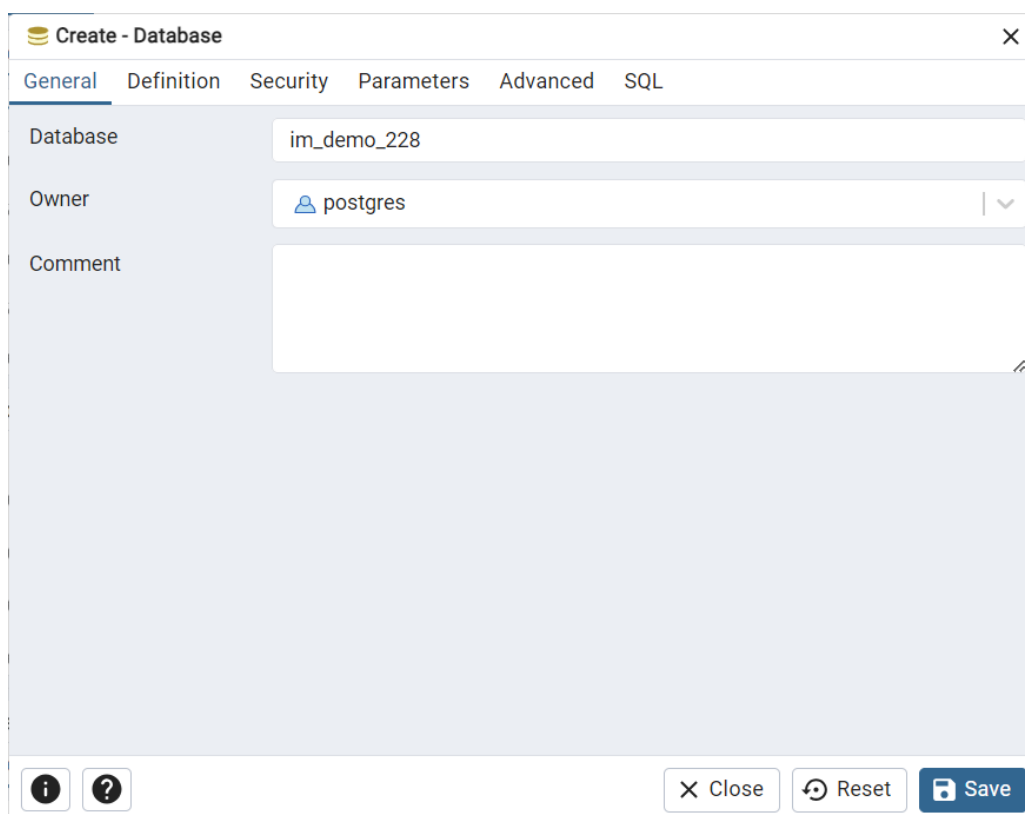
- Перейти под пользователя `postgres` и подключиться к СУБД
 - a) `sudo -s`
 - b) `su - postgres`
 - c) `psql`
- Создать базу данных
 - d)

```
CREATE DATABASE im_demo
WITH
OWNER = postgres
TEMPLATE = template0
ENCODING = 'UTF8'
LC_COLLATE = 'en_US.UTF-8'
LC_CTYPE = 'en_US.UTF-8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1
IS_TEMPLATE = False;
```
 - e) `ALTER DATABASE "im_demo" SET search_path TO im;`
- Для выхода из `psql` - `\q`
- Восстанавливаем БД из резервной копии
 - f) `pg_restore -U postgres -d im_demo -v "/путь до файла/im_demo.backup"`



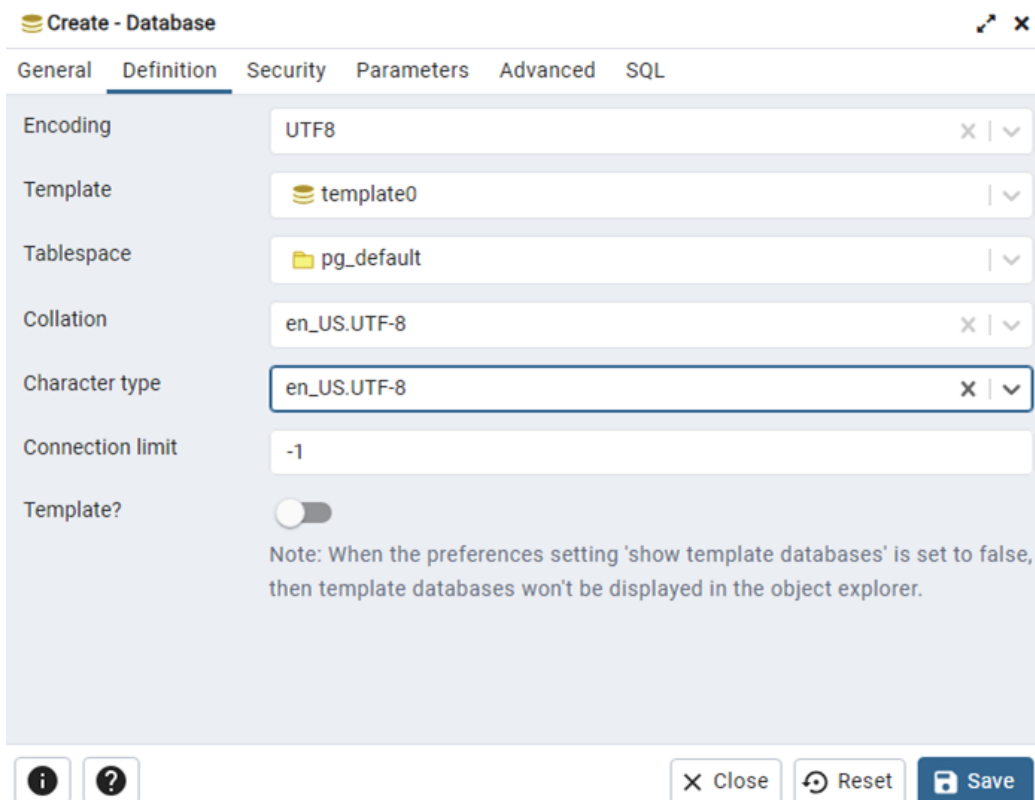
Через pgAdmin:

- Создать пустую базу данных со следующими параметрами:



The screenshot shows the 'Create - Database' dialog box in pgAdmin, with the 'General' tab selected. The 'Database' field contains 'im_demo_228'. The 'Owner' field is set to 'postgres'. The 'Comment' field is empty. At the bottom, there are buttons for 'Close', 'Reset', and 'Save'.

Database	im_demo_228
Owner	postgres
Comment	

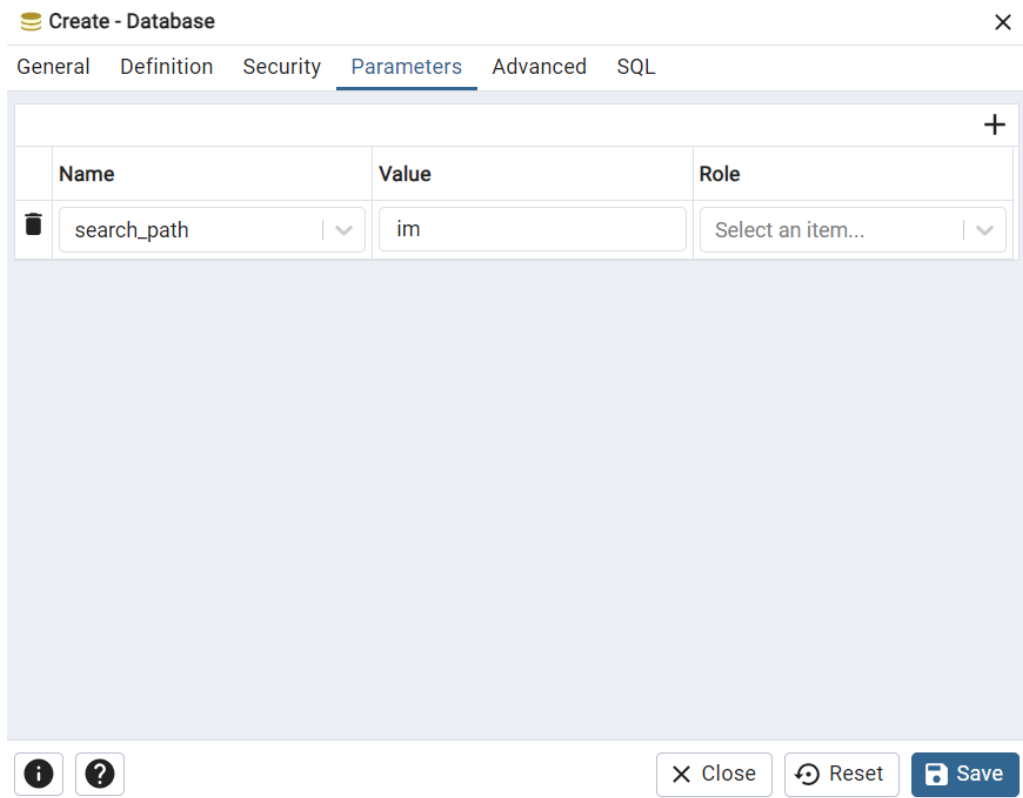


The screenshot shows the 'Create - Database' dialog box in pgAdmin, with the 'Definition' tab selected. The 'Encoding' is set to 'UTF8'. The 'Template' is 'template0'. The 'Tablespace' is 'pg_default'. The 'Collation' is 'en_US.UTF-8'. The 'Character type' is 'en_US.UTF-8'. The 'Connection limit' is '-1'. The 'Template?' checkbox is unchecked. A note at the bottom states: 'Note: When the preferences setting 'show template databases' is set to false, then template databases won't be displayed in the object explorer.' At the bottom, there are buttons for 'Close', 'Reset', and 'Save'.

Encoding	UTF8
Template	template0
Tablespace	pg_default
Collation	en_US.UTF-8
Character type	en_US.UTF-8
Connection limit	-1
Template?	<input type="checkbox"/>

Note: When the preferences setting 'show template databases' is set to false, then template databases won't be displayed in the object explorer.



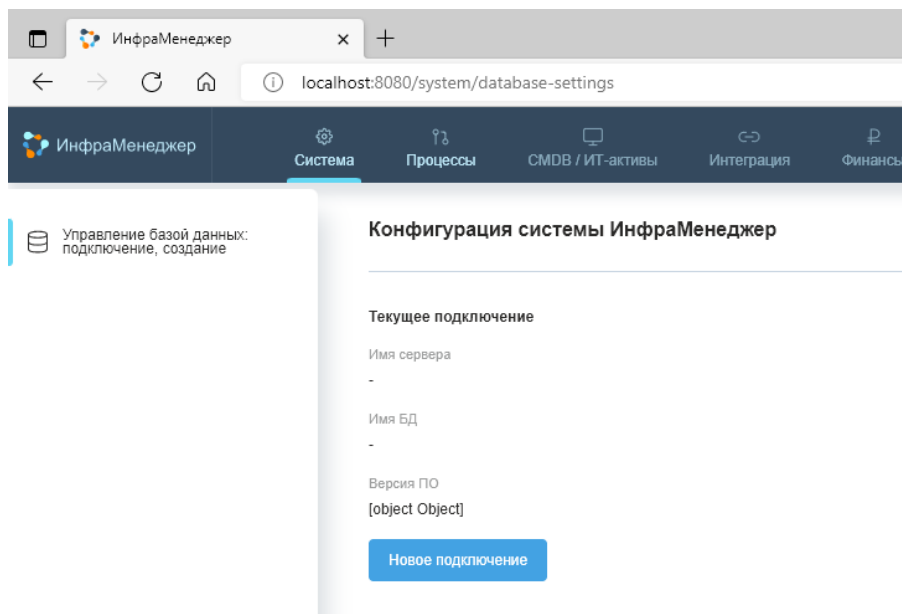


- Восстановить в неё имеющийся дамп пустой или демонстрационной базы данных.

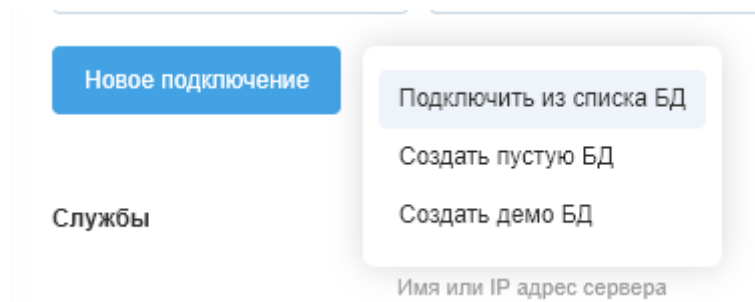
3.2. Настройка подключения ИнфраМенеджер к базе данных

Приступим к настройке соединения с БД:

- Перейдите по адресу <https://ваш домен/admin/system/database-settings>
- Нажмите кнопку «Новое подключение»:



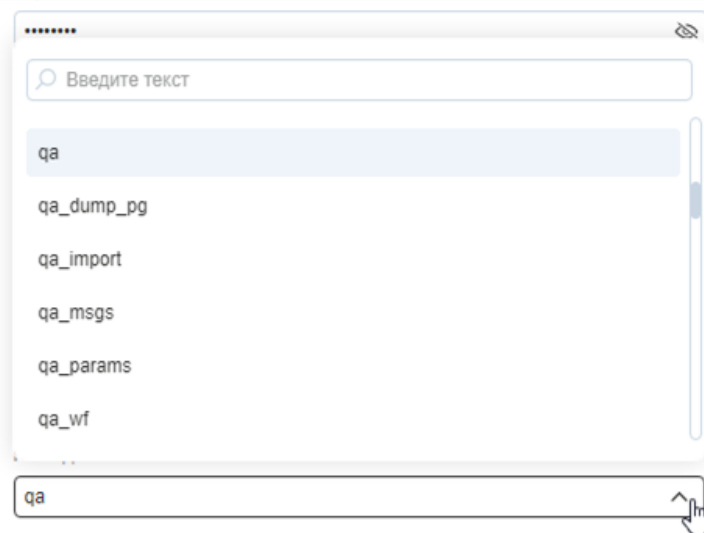
- Из появившегося меню выбрать «Подключить из списка БД»



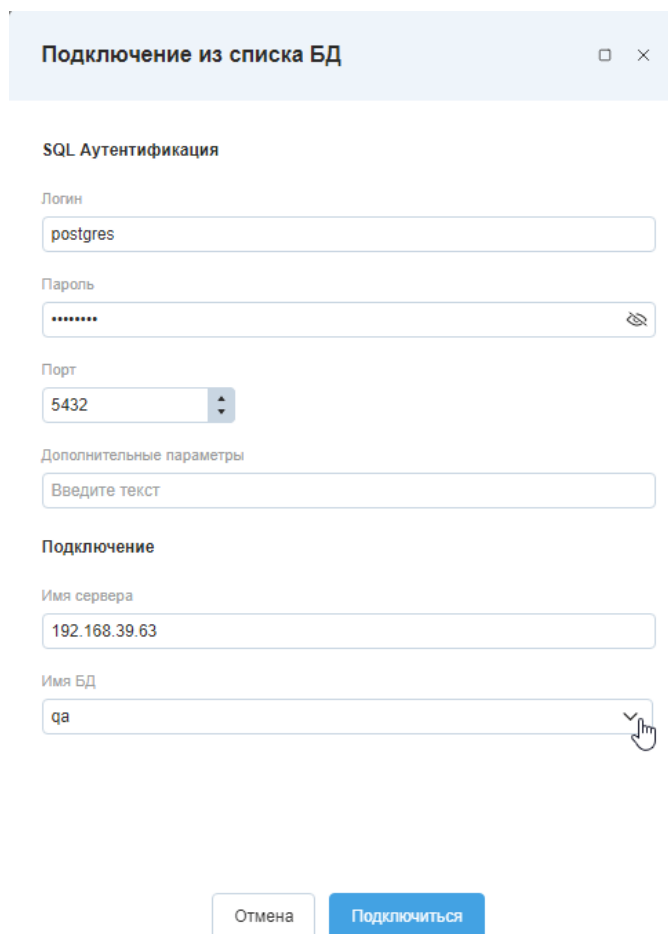
- Ввести логин и пароль от пользователя СУБД, а также порт подключения к СУБД. Для подключения к БД с логином и паролем в рамках SQL авторизации необходимо ввести правильные логин и пароль соответствующей учетной записи SQL, например, суперпользователя postgres, заведенного при установке СУБД.

- После ввода данных появится поле «Имя сервера» - в него надо ввести адрес/доменное имя сервера СУБД Postgres, к которому производится подключение, после чего нажать курсором мыши на форме вне поля «Имя сервера».
- При успешном подключении будет подгружен список доступных баз данных для подключения ИнфраМенеджера:





- Выбрать нужную базу данных и нажать «Подключиться».



- После чего необходимо перезагрузить все контейнеры. Для этого выполните команду:

```
sudo bash im_run_linux.sh restart
```

- Убедитесь, что прошла миграция базы данных.

Миграция отвечает за обновление базы данных ИнфраМенеджера до актуальной версии продукта. Процедура миграции базы данных нужна для корректной работы ИнфраМенеджер. Её необходимо выполнять при первой установке ИнфраМенеджер и при последующем обновлении продукта.

В предыдущих версиях ИнфраМенеджер 7.0 миграция выполнялась отдельно – теперь же модуль миграции встроен в контейнер web-run и выполняется перед запуском основного приложения

Для того, чтобы проверить, прошла ли миграция, выполните команду **docker logs web-run**. Если миграция обрабатывает успешно, то Вы увидите список выполненных скриптов с сообщением об успешном завершении:

```
Loaded image: migration-im:demo_v7.0.228
Creating migration-run ... done
Attaching to migration-run
migration-run | Start db up
migration-run | Beginning transaction
migration-run | Beginning database upgrade
migration-run | Checking whether journal table exists..
migration-run | Journal table does not exist
migration-run | Executing Database Server script '2022-01-01_form_build_tables.sql'
migration-run | Checking whether journal table exists..
migration-run | Creating the "im"."ScriptsMigration" table
migration-run | The "im"."ScriptsMigration" table has been created
migration-run | Executing Database Server script '2022-10-07_FirstMigration.sql'
migration-run | Executing Database Server script '2022-10-12_DropRowVersion_Settings.sql'
migration-run | Executing Database Server script '2022-10-12_sequence_manufacturer.sql'
migration-run | Executing Database Server script '2022-10-12_unique_index_manufacturer.sql'
migration-run | Executing Database Server script '2022-10-14_insert_icons.sql'
migration-run | Executing Database Server script '2022-10-17_AdapterIdDefault.sql'
migration-run | Executing Database Server script '2022-10-17_creeping_line_name.sql'
migration-run | Executing Database Server script '2022-10-17_creeping_line_subject_value.sql'
migration-run | Executing Database Server script '2022-10-17_massive_incident.sql'
migration-run | Executing Database Server script '2022-10-18_ui_csv_configuration_row_version .sql'
migration-run | Executing Database Server script '2022-10-18_ui_settings_row_version.sql'
migration-run | Executing Database Server script '2022-10-21_CabinetAddFields.sql'
migration-run | Executing Database Server script '2022-10-21_CreateUserIM.sql'
migration-run | Executing Database Server script '2022-10-21_create_primarykey_to_object_access.sql'
migration-run | Executing Database Server script '2022-10-22_Fix_FormBuilder.sql'
migration-run | Executing Database Server script '2022-10-22_UI_RtcTypes_workorderType_workorder_Priorities.sql'
```

```
migration-run | Executing Database Server script '2024-05-01_Rename_empty_locations.sql'
migration-run | Executing Database Server script '2024-05-01_Update_ConfigurationUnitDependencies.sql'
migration-run | Executing Database Server script '2024-05-02_Update_ActivePorts.sql'
migration-run | Executing Database Server script '2024-05-03_Add_Default_To_InfrastructureSegment.sql'
migration-run | Executing Database Server script '2024-05-03_The_external_ID_field_set_null_to_manufacturers.sql'
migration-run | Executing Database Server script '2024-05-05_Create_view_asset.sql'
migration-run | Executing Database Server script '2024-05-06_Update_func_IsLogical.sql'
migration-run | Executing Database Server script '2024-05-07_Change_Network_Device_Nullable.sql'
migration-run | Executing Database Server script '2024-05-07_The_width_and_height_fields_are_reset_to_null_for_active_equipment_types.sql'
migration-run | Executing Database Server script '2024-05-07_Update_view_AssetNumber.sql'
migration-run | Executing Database Server script '2024-05-08_Update_func_get_full_object_location_path.sql'
migration-run | Executing Database Server script '2024-05-13_The_DefaultValue_of_external_id_field_were_dropped_for_types_of_equipment.sql'
migration-run | Executing Database Server script '2024-05-16_Update_func_GetFullObjectLocation.sql'
migration-run | Executing Database Server script '2024-05-16_Update_func_GetFullObjectName.sql'
migration-run | Upgrade successful
migration-run | Success!
migration-run exited with code 0
Going to remove migration-run
Removing migration-run ... done
```

Если все скрипты миграции для этой базы выполнены, то Вы увидите следующее:




```
Loaded image: migration-im:demo_v7.0.228
Creating migration-run ... done
Attaching to migration-run
migration-run | Start db up
migration-run | Beginning transaction
migration-run | Beginning database upgrade
migration-run | Checking whether journal table exists..
migration-run | Fetching list of already executed scripts.
migration-run | No new scripts need to be executed - completing.
migration-run | Success!
migration-run exited with code 0
Going to remove migration-run
Removing migration-run ... done
```



4. Аутентификация при работе с WEB-интерфейсом

Для того, чтобы пользователь мог авторизоваться в своем личном кабинете в Web-интерфейсе ИнфраМенеджер, необходимо, чтобы в его свойствах (карточка пользователя в БД ИнфраМенеджер):

- 7) Было правильно заполнено поле Логин (например, в формате <имя домена>\<имя пользователя>);
- 8) Было указано свойство «Разрешить web-идентификацию в 'Службу поддержки'».

Если эти условия выполнены, то для подключения к WEB-интерфейсу наберите в адресной строке браузера: <имя или адрес WEB-сервера>.

Клиентская часть ПО, пример: <https://im.company.example/sd>

Административный интерфейс, пример: <https://im.company.example/admin>

ВАЖНО: взаимодействие с системой должно происходить через DNS-имя, а не через ip-адрес. Не забудьте прописать в своем локальном DNS сервер, на котором разворачивается система.

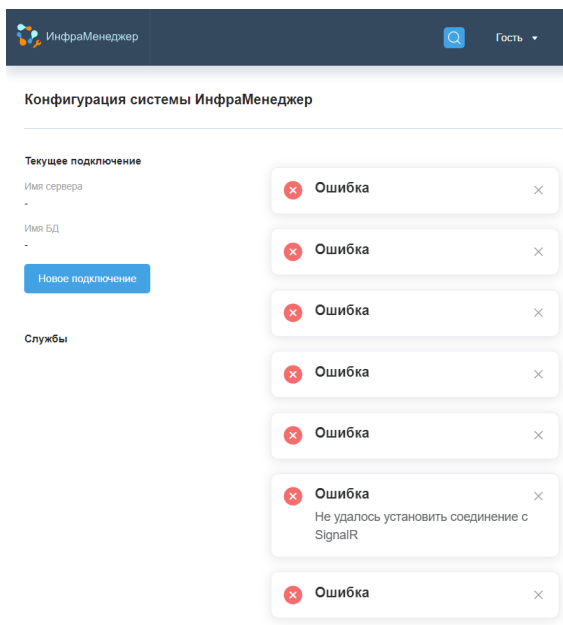


5. Часто задаваемые вопросы и решения

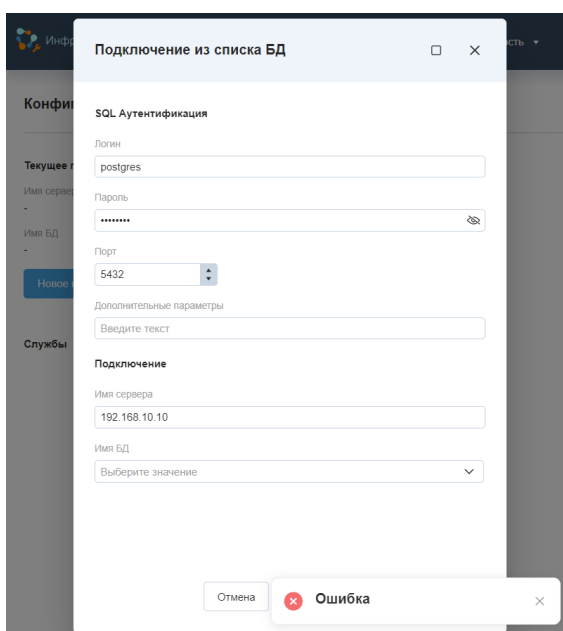
В данном разделе описаны некоторые частые вопросы/сложности, которые могут возникнуть в ходе установки ИнфраМенеджер 7.0, и пути их решения.

5.1. Не получается подключить базу в ИнфраМенеджер 7.0

Ситуация: при переходе на страницу настройки подключения к базе данных (https://admin.ваш_домен/system/database-settings) вываливается список таких ошибок:



При попытке подключения базы данных появляется уведомление об ошибке, а список баз не появляется:



Решение: проверьте заполнение переменных **VUE_APP_SERVER_URL** и **VITE_APP_SERVER_URL** в файле `docker-compose-run.yml` (в блоках `admin` и `sd`). Там должно быть указано доменное имя хоста с `https://`

```
VUE_APP_SERVER_URL: https://server.inframanager
```

После исправления выполните скрипт **im_run_linux.sh** для перезагрузки системы.

