



**инфоэнерго**

ООО «АЛЬФА-ИНТЕГРАТОР-ИНФОЭНЕРГО»

**Автоматизированная информационная система  
сбора и предварительной обработки данных  
«Предбиллинг»  
(АИС Предбиллинг)**

**ИНСТРУКЦИЯ АДМИНИСТРАТОРА  
(ДЛЯ УСТАНОВКИ ПО)**

**Санкт-Петербург  
2023**

## Оглавление

Принятые термины и сокращения .....	2
1. Общие положения .....	2
2. Описание архитектуры АИС Предбиллинг .....	2
3. Установка и настройка системы .....	4
4. Настройка микросервисов. ....	7

## Принятые термины и сокращения

Термин	Определение
АИС Предбиллинг	Автоматизированная информационная система сбора и предварительной обработки данных «Предбиллинг» (АИС Предбиллинг)
ПО	Программное обеспечение
СУБД	Система управления базами данных
Rabbit MQ	Программный брокер сообщений на основе стандарта AMQP
AMQP	Протокол передачи сообщений

## 1. Общие положения

Настоящая Инструкция системного администратора разработана с целью:

- описания архитектуры системы и взаимосвязи её компонентов;
- описания процесса установки и настройки системы;
- определения порядка диагностирования проблем функционирования системы;
- упорядочения работы должностных лиц, связанной с диагностированием проблем функционирования системы.

В настоящем документе регламентируются действия при выполнении следующих мероприятий:

- установка и настройка системы;
- диагностирование проблем функционирования системы.

## 2. Описание архитектуры АИС Предбиллинг

Предбиллинг представляет собой программно-аппаратный комплекс, состоящий из серверов приложений, сервера базы данных и установленного на них ПО.

Схема развертывания АИС Предбиллинг в инфраструктуре Заказчика представлена на Рис.1

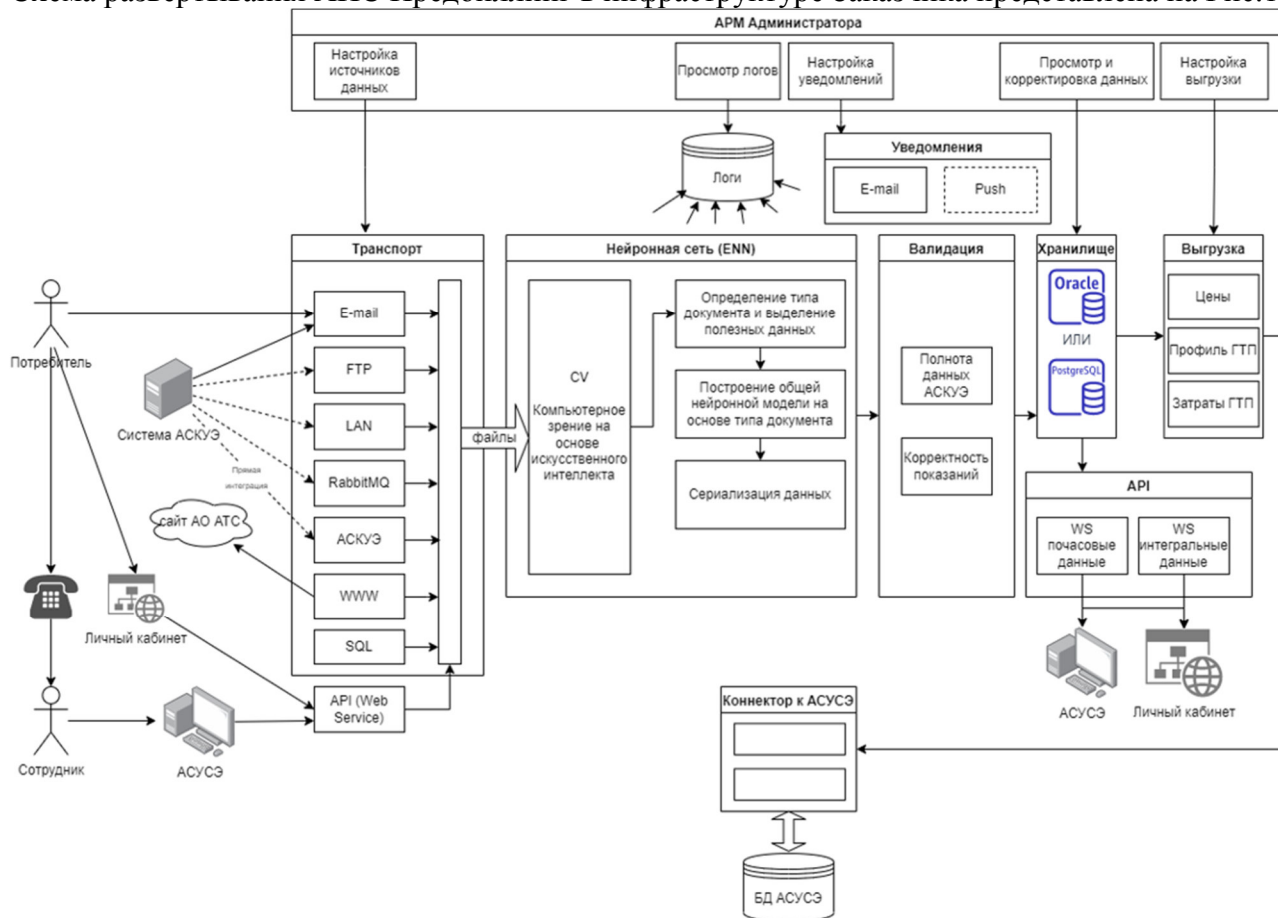


Рис 1. Схема развертывания АИС Предбиллинг

Предбиллинг включает в себя следующие микросервисы:

«Транспорт» - Отвечает за приём данных из всех источников.

«Валидация» - Отвечает за достоверность пришедших данных и последующее сохранение в АСУСЭ.

«АРМ Администратора» - Представляет из себя веб-интерфейс для администрирования АИС Предбиллинг.

Остальные блоки являются полностью автономными и служат для связи основных микросервисов.

Для микросервисов «Валидация», «Нейронная сеть» допускается запуск множества экземпляров для увеличения производительности.

На всех серверах операционной системой является РЕД ОС версии 7.3

Сервер приложений (также установлен на всех серверах) RabbitMQ версии 3.8.14

Сервер приложений (также установлен на всех серверах) ASP.NET Core Runtime версии 3.1

На сервере базы данных установлена СУБД Postgres Standard 14.5

На основном сервере АИС Предбиллинг также установлено следующее ПО:

- gs - Интерпретатор языков Adobe Systems PostScript. Нужен для изменений версий PDF файлов.
- libgdipplus - Графическая библиотека для обеспечения работы с PDF файлами.

### 3. Установка и настройка системы

Установка сервера приложений производится на одном сервере АИС Предбиллинг

#### 3.1 Rabbit MQ

##### 3.1.1 Установка и настройка Rabbit MQ

3.1.1.1 Проверить доступность модуля Rabbit MQ соответствующей версии в репозитории:

```
sudo dnf search rabbitmq-server
```

3.1.1.2 Установить модуль Rabbit MQ из репозитория и запуск сервиса:

```
sudo dnf install rabbitmq-server  
sudo systemctl start rabbitmq-server.service  
sudo systemctl enable rabbitmq-server.service  
sudo rabbitmq-plugins enable rabbitmq_management
```

3.1.1.3 Проверить установленную версию Rabbit MQ:

```
sudo rabbitmqctl version
```

3.1.1.4 Открыть необходимые порты для подключения к приложениям на firewall-е (в данном случае приложение на порту {5672,15672}/tcp):

- добавляем порт  
**sudo firewall-cmd --permanent --add-port={5672,15672}/tcp**
- перезапускаем firewall  
**sudo firewall-cmd --reload**
- проверяем включение правила  
**sudo firewall-cmd --list-all**

#### 3.2 Установка и настройка Postgres

Установка СУБД Postgres Pro производится на сервере баз данных.

3.2.1 Проверить доступность модуля Postgres соответствующей версии в репозитории и включить необходимый поток в случае необходимости:

```
sudo dnf search Postgres
```

3.2.2 Установить Postgres из репозитория:

```
sudo dnf install Postgres server Postgres contrib
```

3.2.3 Открыть порт для подключения к Postgres на firewall-е (по-умолчанию это порт 5432/tcp):

- добавляем порт  
**sudo firewall-cmd --permanent --zone=public --add-port=5432/tcp**
- перезапускаем firewall  
**sudo firewall-cmd --reload**
- Проверяем включение правила  
**sudo firewall-cmd --list-all**

### 3.2.4 Инициализировать базу данных

**sudo Postgres-setup initdb**

### 3.2.5 Запустить и включить в автозапуск службу

**sudo systemctl enable --now Postgres**

### 3.2.6 Проверить доступность и установленную версию СУБД:

**sudo -u postgres psql -c "SELECT version();"**

### 3.2.7 Задать пароль пользователю postgres, от имени которого запускается СУБД:

**sudo passwd postgres**

### 3.2.8 Задать пароль суперпользователю СУБД postgres:

**sudo -u postgres psql**

*\password postgres*

*\q*

### 3.2.9 Настроить парольную аутентификацию в файле конфигурации pg\_hba.conf (расположение по-умолчанию \$HOME/data/pg\_hba.conf в каталоге пользователя postgres):

```
# "local" is for Unix domain socket connections only
local all all md5
# IPv4 local connections:
host all all 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# LAN connections
host all all 0.0.0.0/0 md5
```

В разделе LAN connections можно ограничить доступ только определенными сетями, указав соответствующую маску

### 3.2.10 Настроить следующие параметры в файле конфигурации Postgres Pro.conf (расположение по-умолчанию \$HOME/data/Postgres Pro.conf в каталоге пользователя postgres):

```
#Включаем прослушивание на внешних интерфейсах системы
listen_addresses = '*'
#Устанавливаем количество соединений
max_connections = 100
#Включаем уровень записи в WAL, необходимый для восстановления из резервной копии
wal_level = replica
#Включаем режим архивации
archive_mode = on
#Устанавливаем команду архивации, вместо /opt/Postgres Pro/wal_backup/ должен быть каталог, в котором будут размещаться резервные копии WAL
archive_command = 'test ! -f /opt/Postgres Pro/wal_backup/%f.gz && /usr/bin/gzip -c %p > /opt/Postgres Pro/wal_backup/%f.gz'
#Настраиваем политику удержания WAL
wal_keep_segments = 60
#Параметры локализации
lc_messages = 'en_US.UTF-8'
lc_monetary = 'ru_RU.UTF-8'
lc_numeric = 'ru_RU.UTF-8'
lc_time = 'ru_RU.UTF-8'
#Параметры отображения даты
datestyle = 'iso, mdy'
```

3.2.11 Перезапустить сервис Postgres для применения новых параметров:

**sudo systemctl restart postgresql**

3.2.12 Настройка резервного копирования

Создать скрипт резервного копирования. В данном примере каталог запуска Postgres - /opt/postgresql, на сервере хранятся 2 последних базовых резервных копии (в каталоге db\_backup) и WAL за последние 2-е суток (в каталоге wal\_backup):

```
#!/bin/bash

PG_HOME=/opt/postgresql
export PG_HOME
mkdir $PG_HOME/pg_backup
/usr/bin/pg_basebackup -U postgres -D $PG_HOME/pg_backup -Ft -z -Xf

INDEX=$(date +"%u")

test -e $PG_HOME/db_backup/base.${INDEX}.tar.gz && rm $PG_HOME/db_backup/base.${INDEX}.tar.gz
cp $PG_HOME/pg_backup/base.tar.gz $PG_HOME/db_backup/base.${INDEX}.tar.gz

test -e $PG_HOME/db_backup/base.last.tar.gz && rm $PG_HOME/db_backup/base.last.tar.gz
ln $PG_HOME/db_backup/base.${INDEX}.tar.gz $PG_HOME/db_backup/base.last.tar.gz

rm -r $PG_HOME/pg_backup
test -e $PG_HOME/db_backup/base.${INDEX}.tar.gz && test -e $PG_HOME/db_backup/base.$(date --date="-1 day" +"%u").tar.gz && find $PG_HOME/db_backup -type f -mtime +1 -exec rm {} \;
find $PG_HOME/wal_backup -type f -mtime +2 -exec rm {} \;
```

3.3 Сервисы под управлением .NET Core

3.3.1 Установка среды выполнения .NET Core

Установка сервера приложений производится на всех серверах АИС Предбиллинг.

**sudo dnf install aspnetcore-runtime-3.1**

3.3.2 Запуск и мониторинг сервисов

Для поддержания непрерывной работы сервисов на .NET Core необходимо создать службу systemd. Для этого необходимо создать файл с описанием службы - например, /etc/systemd/system/prebill.service

Файл службы имеет следующее содержимое:

```
[[Unit]
Description=Предбиллинг основной сервис

[Service]
WorkingDirectory=/home/prebill/dotnet/prebill.service
ExecStart=/usr/bin/dotnet prebill.service.dll
Restart=on-failure
RestartSec=20
SyslogIdentifier=dotnet-prebill-svc
User=prebill
Environment=ASPNETCORE_ENVIRONMENT=Test
Environment=DOTNET_ENVIRONMENT=Test
Environment=TNS_ADMIN=/opt/oracle/product/11.2.0.4/db_1/network/admin
Environment=SERVICE=service

[Install]
WantedBy=multi-user.target
```

Служба запускается командой:

```
sudo systemctl start prebill.service
```

Для обеспечения ее непрерывной работы выполняется команда:

```
sudo systemctl enable prebill.service
```

Это позволит службе автоматически перезапускаться при перезапуске сервера и других возможных сбоях.

При любых изменениях сервиса необходимо перезапустить службу командой:

```
sudo systemctl restart prebill
```

Для проверки состояния службы можно выполнить следующие команды:

```
sudo systemctl status prebill
```

```
sudo journalctl -u prebill
```

Данную настройку нужно сделать для каждого микросервиса.

## 4. Настройка микросервисов.

### 4.1 Файл конфигураций

Файл конфигурации задаёт правила работы микро сервиса и чем будет он заниматься. Файл состоит из 2х частей `appsettings.{DOTNET_ENVIRONMENT}.json` и `appsettings-service.{SERVICE}.json`. `DOTNET_ENVIRONMENT` и `SERVICE` это переменные среды, которые задаются при настройке сервиса .NET Core на прошлом шаге.

#### 4.1.1 appsettings

В файле есть следующие настройки:

- `ConnectionString`: список соединений с БД ASUSE и PREBILL
- `DbTypes`: список типов вышеуказанных БД
- `Smtп`: настройки SMTP
- `emailAlert`: указывается почта от которой и на которую будут слаться ошибки
- `emailFiles`: указываются доступы к почте для отправки сообщений
- `RabbitConfig`: настройки RabbitMQ
- `Storage`: настройки локального хранилища
- `Connectors`: указываются способы получения файлов
- `Workers_Interval`: указываются интервалы сбора файлов

#### 4.1.2 appsettings-service

В файле есть следующие настройки:

- `Workers`: задаёт логику работы микросервису.
  1. `TransportWorker`: занимается забором файлов из разных источников
  2. `RecognizerWorker`: занимается распознаванием файлов
  3. `ValidationWorker`: занимается валидацией данных после распознавания
  4. `PersistWorker`: занимается записью данных в ASUSE
  5. `NotifyWorker`: занимается рассылкой почты с результатом обработки файлов
  6. `RequeueWorker`: занимается пере постановкой файлов на вторичную обработку



В рамках одного файла может быть несколько настроек SERVICE.

#### 4.1.3 Применение настроек

После любых изменений нужно перезапустить сервис командой

**sudo systemctl restart prebill**

#### 4.2 Настройка веб сервиса

В файле appsettings.{DOTNET\_ENVIRONMENT}.json нуно указать номер порта, на котором будет работать сервис. По умолчанию это порт 5000, если не указан никакой порт.

После изменения порта нужно поубавить в разрешённые порты в firewall.

- добавляем порт  
**sudo firewall-cmd --permanent --add-port=5000/tcp**
- перезапускаем firewall  
**sudo firewall-cmd --reload**
- проверяем включение правила  
**sudo firewall-cmd --list-all**