ЗАО «Инфосистемы Джет»

Информационная система управления инцидентами информационной безопасности «Джет Сигнал»

Руководство системного программиста

Москва
2016

Аннотация

В документе описывается назначение и функции информационной системы управления инцидентами информационной безопасности «Джет Сигнал». Приводится методика её установки, настройки, а также описываются типовые приемы работы.

Содержание

1 Общие сведения о программе

1.1 Обозначение и наименование

Полное наименование программы: Информационная система управления инцидентами информационной безопасности «Джет Сигнал».

Сокращенное наименование: Система «Джет Сигнал» или Система.

1.2 Назначение и функции программы

Система «Джет Сигнал» предназначена для повышения эффективности обработки инцидентов информационной безопасности (далее – ИБ) за счет автоматизации следующих процессов:

- планирование, учет и контроль работы дежурных смен;
- сбор, регистрация и обработка информации об инцидентах ИБ;
- контроль исполнения поручений в рамках решения задач по устранению причин и последствий инцидентов ИБ, а также по обеспечению защиты объектов и активов ИБ;
- доведение информационно-распорядительных документов до подразделений;
- управление жизненным циклом инцидентов ИБ;
- ведение базы знаний;
- обмен формализованной информацией об инцидентах между подразделениями;
- обмен быстрыми сообщениями между операторами Системы.

Задачи, решаемые Системой «Джет Сигнал»:

- планирование, учет и контроль работы дежурных смен;
- ведение планов мероприятий по реагированию на инциденты ИБ;
- регистрация и ведение карточек инцидентов в автоматическом режиме и вручную;
- автоматическое предоставление плана мероприятий по устранению последствий инцидента ИБ в зависимости от типа инцидента;
- обмен формализованной информацией об инцидентах между задействованными подразделениями;
- создание и ведение информационных кампаний;
- регистрация и ведение поручений;
- регистрация и ведение распорядительных документов;
- ведение базы знаний и базы угроз;

- публикация информации (ведение новостей);
- обмен быстрыми сообщениями между пользователями (веб-чат);
- ведение справочной информации;
- ведение профилей пользователей;
- ведение ролей и прав доступа;
- просмотр журнала действий пользователей.

Система «Джет Сигнал» поддерживает схемы развёртывания:

- для приложения, которое установлено в виде одного автономного узла;
- распределённой информационной системы при развёртывании двух и более экземпляров приложения с обеспечением обмена информацией между ними.

В Системе реализовано взаимодействие узлов, функционирующих в разных сегментах информационной безопасности:

- двунаправленное взаимодействие при одинаковом классе защиты сегментов. Например, обмен данными между закрытыми сегментами;
- однонаправленное взаимодействие при разных классах защиты сегментов. Например, передача данных от интернет-сегмента к закрытому сегменту.

Система «Джет Сигнал» поддерживает работу с мандатной моделью управления доступом Astra Linux Special Edition 1.5. При этом в зависимости от назначения установки приложения мандатная модель управления доступом может быть как задействована, так и отключена в момент установки приложения.

1.3 Программные и технические средства

Система функционирует на платформах Intel x86, Intel x86 и AMD64. Рекомендуемые характеристики аппаратного обеспечения сервера:

- процессор: не менее 4 ядер с тактовой частотой не менее 3 ГГц;
- объем оперативной памяти: не менее 3 ГБ;
- жесткий диск: не менее 250 ГБ.

Сервер функционирует под управлением операционной системы Astra Linux Special Edition 1.5 и использует СУБД PostgreSQL 9.4, веб-сервер Apache 2.2.22, интерпретатор PHP 5.4.4, сервер очередей RabbitMQ.

Для управления работой Системы используется веб-интерфейс администратора безопасности и интерфейс командной строки (CLI) для системного администратора.

Компьютер, используемый в качестве АРМ администратора безопасности и системного администратора, должен отвечать следующим требованиям:

- процессор: не менее 4 ядер с тактовой частотой не менее 2,8 ГГц;
- объём оперативной памяти: не менее 4 ГБ;
- объём жёсткого диска: не менее 128 ГБ;
- разрешение экрана при работе с интерфейсом не менее 1024x768.

АРМ администратора безопасности и системного администратора должны быть оборудованы сетевым адаптером Ethernet.

Для корректной работы Системы требуются следующие пакеты:

- PHP:
  - php5-cli 5.4.4-2astra2 – интерфейс командной строки PHP;
  - php5-common 5.4.4-2astra2 – основные файлы PHP;
  - php5-curl 5.4.4-2astra2 – библиотека для выполнения сетевых запросов;
  - php5-gd 5.4.4-2astra2 – библиотека для работы с изображениями;
  - php5-imap 5.4.4-2astra2 – библиотека для чтения данных по протоколу IMAP;
  - php5-pgsql 5.4.4-2astra2 – библиотека для подключения PHP к PostgreSQL.
- PostgreSQL:
  - postgresql-9.4 9.4.5-1astra.se15 – сервер СУБД PostgreSQL;
  - postgresql-client-9.4 9.4.5-1astra.se15 – клиент СУБД PostgreSQL;
  - postgresql-client-common 165astra.se7 – основные файлы для клиента;
  - postgresql-common 165astra.se7 – основные файлы для сервера;
  - postgresql-contrib-9.4 9.4.5-1astra.se15 – дополнительные файлы для сервера.
- Apache2:
  - apache2 2.2.22-13astra.se15 – метапакет Apache2;
  - apache2.2-bin 2.2.22-13astra.se15 – бинарные файлы Apache2;
  - apache2.2-common 2.2.22-13astra.se15 – основные файлы Apache2;
  - apache2-mpm-prefork 2.2.22-13astra.se15 – пакет Apache2;
  - apache2-utils 2.2.22-13astra.se15 – утилиты для Apache2;
  - libapache2-mod-auth-pam 1.1.1-9astra.se3 – модуль для PAM-аутентификации в Apache2 (мандатная модель управления доступом);
  - libapache2-mod-php5 5.4.4-2astra2 – модуль для выполнения PHP на сервере.
- RabbitMQ:
  - rabbitmq-server 2.8.4-1 – сервер очередей RabbitMQ.
- Erlang:

- erlang-nox 1:15.b.1-dfsg-4+deb7u1astra.se1 – Erlang без X-интерфейса, используется RabbitMQ.

Для обеспечения дополнительной защищённости передаваемых данных между узлами системы, между которыми не требуется двунаправленный обмен интеграционными сообщениями, возможна организация сетевого взаимодействия через специализированный однонаправленный шлюз СТРОМ-1000.

Шлюз должен соответствовать следующим характеристикам:

- обеспечить скорость передачи информации не менее 1 Гбит/с;
- иметь сетевые интерфейсы:
  - внешний: RJ-45, медь, витая пара, Ethernet 100/1000BASE-T; SFP, оптика, Ethernet 1000BASE-X;
  - внутренний: SC, многомодовая оптика, 850нм, 1000BASE-SX.

1.4 Требования к системному программисту

1. Наличие квалификации в администрировании ОС Debian Linux.
2. Наличие квалификации в администрировании ОС Astra Linux Special Edition 1.5 включая средства настройки и администрирования мандатной модели управления доступом на уровне пользователей, процессов, файлов.
3. Наличие квалификации в администрировании веб-сервера Apache2, управлении модулями, в том числе PHP.
4. Наличие квалификации в администрировании СУБД PostgreSQL с поддержкой механизма мандатных меток.
5. Наличие квалификации в администрировании сервера очередей RabbitMQ с поддержкой механизма мандатных меток в режиме совместимости.
6. Наличие квалификации в администрировании однонаправленного шлюза СТРОМ-1000, включая поставляемые со шлюзом средства однонаправленной передачи файлов.
7. Наличие квалификации в построении и администрировании ЛВС.

2 Структура программы

В состав «Джет Сигнал» входят подсистемы:

- управления дежурными сменами;
- управления инцидентами;
- управления поручениями;
- ведения базы знаний;
- взаимодействия с другими программами;
- администрирования;
- новостная лента;
- обмена сообщениями (веб-чат).

Подсистема управления дежурными сменами предназначена для планирования, учета и контроля дежурных смен.

Подсистема управления инцидентами предназначена для информационного и процессного обеспечения работы подразделений по обнаружению и предотвращению компьютерных атак, а также для расследования компьютерных инцидентов.

Подсистема управления поручениями предназначена для:

- ведения и контроля исполнения поручений в рамках задач по обеспечению защиты объектов и активов ИБ;
- ведения и контроля исполнения распорядительных документов.

Подсистема ведения базы знаний предназначена для ведения:

- информации об инцидентах.
- базы угроз и уязвимостей активов ИБ;
- регламентов ИБ.

Подсистема взаимодействия предназначена для обмена сообщениями в условиях территориальной и сетевой распределенности IT-инфраструктуры.

Подсистема взаимодействия предоставляет единый интерфейс обмена данными.

Подсистема администрирования предназначена для управления учетными записями пользователей, ведения ролевой модели, регистрации действий пользователей и системных событий. В подсистеме также реализованы функции ведения справочной информации.

Новостная лента предназначена для публикации новостей и сообщений, в том числе об актуальных угрозах информационной безопасности.

Подсистема обмена сообщениями (веб-чат) предназначена для обеспечения быстрого обмена сообщениями между пользователями.

Дополнительные сведения о структуре приведены в документе Информационная система управления инцидентами информационной безопасности «Джет Сигнал». Описание программы.

3 Установка и настройка программы

3.1 Установка системы «Джет Сигнал»

3.1.1 Среда установки

Система разворачивается на технических средствах с предварительно установленной операционной системой Astra Linux Special Edition 1.5.

3.1.1.1 Рекомендации к типовой настройке среды установки

Рекомендовано задать пароль учётной записи операционной системы «root», которая играет роль учётной записи привилегированного технологического пользователя.

Минимальные требования к «железу»

- HDD = 24Gb
  - 4Gb = swap
  - / = все остальное
- 4 CPU
- 4Gb RAM

3.1.2 Планирование и организация схемы узлов Системы

Система «Джет Сигнал» должна поддерживать схемы развёртывания:

- для приложения, которое установлено в виде одного автономного узла;
- распределённой информационной Системы при развёртывании двух и более узлов Системы с обеспечением обмена информацией между ними.

Система использует механизм независимой генерации первичных ключей для записей, хранимых в базе данных. Такой механизм позволяет обеспечить уникальность формируемых идентификаторов и исключить возникновение коллизий при обмене данными между экземплярами Системы.

Для обеспечения корректного функционирования механизма каждому экземпляру приложения должен быть присвоен уникальный целочисленный номер в диапазоне от 2 до 4095.

Идентификатор экземпляра приложения, равный числу 1, является специализированным и обозначает авторитетный узел для поставки обновлений справочников (см. п. 3.2). Если функциональность централизованного обновления справочников не требуется, то такой идентификатор не должен быть задействован в плане Системы.

Все идентификаторы экземпляров приложения должны быть зафиксированы в организационном документе до начала развёртывания Системы. Этот документ должен использоваться системными администраторами при настройке отдельных узлов Системы.

3.1.3 Подготовка технических средств

Перед установкой необходимо подготовить техническое средство (ТС), на которое устанавливается Система, следующим образом:

- проверить правильность подключения клавиатуры и дисплея (KVM-панели) к ТС;
- в случае использования KVM-панели – переключить KVM-панель на взаимодействие с ТС;
- включить ТС;
- при отсутствии в составе ТС CD/DVD-привода – подключить переносной CD/DVD-привод к свободному USB-разъёму ТС.

3.1.4 Установка программного обеспечения

Для корректного функционирования Системы требуется установка следующих программных пакетов, входящих в состав дистрибутива Astra Linux 1.5 Special Edition:

- веб-сервер Apache (apache2=2.2.22-13astra.se15);
- интерпретатор PHP (php5=5.4.4-2astra2) с включённым XCache v2.0.0;
- СУБД PostgreSQL (postgresql-9.4=9.4.5-1astra.se15);
- сервер очередей RabbitMQ (rabbitmq-server=2.8.4-1).

Все действия выполняются под учётной записью суперпользователя (root). Установочный диск Astra Linux 1.5 Special Edition должен быть корректно смонтирован.

По умолчанию файлы приложения устанавливаются в каталог /var/www/jetsignal. Если требуется установка Системы в другой каталог, это следует учесть при вводе команд.

Для установки:

1) Выполнить команду установки необходимых пакетов (может занять несколько минут):

```
# apt-get install postgresql-9.4 postgresql-client-9.4 rabbitmq-server
apache2 libapache2-mod-php5 php5 php5-pgsql php5-xcache php5-imap php5-ldap
php5-gd php5-json php5-imagick php5-curl php5-mcrypt php5-mapscript php5-
xmlrpc php5-xsl php5-tidy php5-enchant php5-recode php5-pspell libapache2-
mod-auth-pam acl
```

2) Выполнить команду:

```
# usermod -a -G shadow www-data
```

3) Выполнить команды установки разрешений на macdb для пользователя www-data:

```
# setfacl -d -m u:www-data:r /etc/parsec/macdb
# setfacl -R -m u:www-data:r /etc/parsec/macdb
# setfacl -m u:www-data:rx /etc/parsec/macdb
# setfacl -x user:astra /etc/parsec/macdb
```

4) Выполнить команду:

```
# nano /etc/pam.d/apache2
```

И убедится, что в файле есть следующие строки (если их там нет, то нужно добавить):

```
@include common-auth
@include common-account
account required pam_tally.so
```

5) Выполнить команду выставления разрешений на запись журнала:

```
# setfacl -m u:www-data:rw /var/log/faillog
```

6) Включить необходимые модули Apache:

```
# a2enmod auth_pam
# a2enmod rewrite
```

3.1.5 Распаковка файлов приложения

Система поставляется на физическом носителе CD в виде архива **.tar.gz**, содержащего все необходимые файлы.

Для распаковки необходимо выполнить следующие действия:

1) Выполнить подключение к программе bash на техническом средстве под учётной записью root.
2) Подключить носитель к техническому средству.

```
ls -al /dev/cdrom*
... /dev/cdrom -> sr0

mount /dev/sr0 /media/cdrom
mount: блочное устройство /dev/sr0 защищен от записи, монтируется только
для чтения

ls /media/cdrom
JETSIGNAL
```

3) Скопировать файл **jetsignal-v0.*.tar.gz** (имя файла включает в себя версию, номер сборки и другую идентифицирующую информацию) в каталог **/var/www/**:

```
# cd /media/cdrom/JETSIGNAL
# cp jetsignal-*.tar.gz -t /var/www
```

4) Распаковать архив с файлами приложения:

```
# cd /var/www
```

```
# tar -zxvf jetsignal-*.tar.gz
```

5) Переименовать полученный каталог в **jetsignal**:

```
# mv jetsignal-build-20170427103008 jetsignal
```

6) Установить рекурсивно владельца и права на каталог:

```
# chmod -R 777 /var/www/jetsignal
# chown -R www-data:www-data /var/www/jetsignal
```

7) Выполнить команды установки мандатных меток на каталог сессий PHP:

```
# pdpl-file 3:0:0:ccnr /var/lib
# pdpl-file 3:0:0:ccnr /var/lib/php5
```

8) Выполнить команду создания каталогов под мандатные метки в случае использования приложения в режиме с мандатными метками:

```
# cd /var/www/jetsignal
# fixdirs.sh
```

Будет выполнено создание каталогов для хранения файлов и подкаталогов с учётом мандатной метки пользователя.

3.1.6 Настройка веб-сервера

Конфигурационный файл веб-сервера поставляется вместе с приложением и доступен по пути **/var/www/jetsignal/config/apache2.sample**.

Для настройки веб-сервера выполнить следующие действия:

1) Скопировать конфигурационный файл в каталог виртуальных хостов Apache2:

```
# cp /var/www/jetsignal/config/apache2.sample /etc/apache2/sites-
available/jetsignal
```

2) Очистить каталог с включёнными виртуальными хостами:

```
# rm –f /etc/apache2/sites-enabled/*
```

ВАЖНО!

Возможны ситуации, когда техническое средство, на которое устанавливается веб-сервер Apache2, используется другими приложениями. В этом случае не следует использовать команду очистки каталога с включёнными виртуальными хостами. Вместо этого необходимо с помощью команды **cat** вручную просмотреть содержимое всех файлов в каталоге **/etc/apache2/sites-enabled/** и удалить либо отредактировать только те из них, которые мешают корректному функционированию Системы и занимают порты 80 и 443.

Также можно разрешить проблему занятости портов путём использования доменного имени. Для этого следует сконфигурировать сетевые маршруты домена (для примера указан

jetsignal.local) на сервер и добавить директиву **ServerName** в файл **/etc/apache2/sites-available/jetsignal**.

3) Открыть файл конфигурации на редактирование:

```
# nano /etc/apache2/sites-available/jetsignal
```

4) Переместить курсор после строки:

```
<VirtualHost *:80>
```

5) С новой строки ввести директиву (для примера указан домен **jetsignal.local**):

```
ServerName jetsignal.local
```

6) Нажать клавиши **Ctrl + X**, затем клавишу **Y**.

Изменения в файле будут сохранены.

7) Создать символическую ссылку на данный виртуальный хост в каталоге включённых виртуальных хостов:

```
# ln -s /etc/apache2/sites-enabled/jetsignal /etc/apache2/sites-
available/jetsignal
```

8) Убедиться, что конфигурация виртуального хоста указана корректно:

```
# apachectl configtest
```

После выполнения этой команды должно появиться следующее сообщение:

```
root@astra:/# apachectl configtest
Syntax OK
```

9) Перезапустить веб-сервер Apache2:

```
# /etc/init.d/apache2 restart
```

После выполнения этой команды должно появиться следующее сообщение:

```
root@astra:~# /etc/init.d/apache2 restart
[....] Restarting web server: apache2
 ... waiting
. ok
```

3.1.7 Настройка базы данных

Для корректной работы Системы рекомендуется использовать базу данных, расположенную локально на том же сервере.

Выполнить команду создания кластера jetsignal на порту 6789:

```
# pg_createcluster -p 6789 9.4 jetsignal
```

Команда выведет следующие сообщения:

```
Creating new cluster 9.4/jetsignal ...
```

```
  config /etc/postgresql/9.4/jetsignal
  data   /var/lib/postgresql/9.4/jetsignal
  locale en_US.UTF-8
Flags of /var/lib/postgresql/9.4/jetsignal set as ------------e-C
  port   6789
```

Для настройки базы данных выполнить следующие действия:

1) Открыть на редактирование файл конфигурации PostgreSQL 9.4:

   ```
   # nano /etc/postgresql/9.4/jetsignal/postgresql.conf
   ```

2) Найти строку с параметром ac_ignore_socket_maclabel сменить значение параметра с true на false:

   ```
   ac_ignore_socket_maclabel       = false #true
   ```

3) Нажать клавиши Ctrl + X, затем клавишу Y.

   Изменения в файле будут сохранены.

4) Открыть на редактирование файл конфигурации системы безопасности PostgreSQL 9.4:

   ```
   # nano /etc/postgresql/9.4/jetsignal/pg_hba.conf
   ```

5) В строке, содержащей:

   ```
   # "local" is for Unix domain socket connections only
   local   all             all                     peer
   ```

   исправить последнее слово «…» на **trust.**

   Строка должна иметь следующий вид:

   ```
   local         all         all               trust
   ```

6) В строке, содержащей:

   ```
   # IPv4 local connections:
   host    all             all             127.0.0.1/32            md5
   ```

   исправить последнее слово «…» на **trust.**

   Строка должна иметь следующий вид:

   ```
   host          all         all         127.0.0.1/32            trust
   ```

7) Нажать клавиши Ctrl + X, затем клавишу Y.

   Изменения в файле будут сохранены.

Для корректной работы мандатной модели управления доступом необходимо настроить мандатные метки на созданной базе данных.

Для корректного подбора данного значения следует руководствоваться таблицей в приложении А «Соответствие мандатных меток уровням допуска».

В нашем случае нужно последовательно запустить выполнение команд для разрешения доступа администратора базы данных **postgres** к хранилищу мандатных меток:

```
# setfacl -m u:postgres:rx /etc/parsec/macdb
# setfacl -m u:postgres:rx /etc/parsec/capdb
# setfacl -d -m u:postgres:r /etc/parsec/macdb
# setfacl -d -m u:postgres:r /etc/parsec/capdb
# setfacl -R -m u:postgres:r /etc/parsec/macdb/*
```

8) Перезапустить PostgreSQL 9.4:

```
# /etc/init.d/postgresql stop
# /etc/init.d/postgresql start
```

9) Скопировать файл с примером конфигурации базы данных в файл конфигурации базы данных:

```
# cp /var/www/jetsignal/config/db.php.sample
/var/www/jetsignal/config/db.php
```

10) Открыть файл конфигурации базы данных приложения на редактирование:

```
# nano /var/www/jetsignal/config/db.php
```

11) В строке, указывающей на параметр dsn, убедиться, что указан следующий текст:

```
'dsn' => 'pgsql:host=127.0.0.1;dbname=jetsignal',
```

где:

– **host** – имя хоста (IP-адрес);
– **dbname** – имя созданной ранее базы данных.

12) При необходимости внести изменения и нажать клавиши Ctrl + X, затем клавишу Y.

Изменения в файле будут сохранены.

13) Подключиться к PostgreSQL под учётную запись postgres:

```
# su postgres
# psql -p 6789
```

14) Ввести команду:

```
# CREATE DATABASE jetsignal;
```

15) Нажать Enter.

Должно появиться сообщение, обозначающее, что база данных успешно создана:

```
CREATE DATABASE
```

16) Подключиться к созданной ранее базе данных – ввести команду:

```
# \c jetsignal
```

17) Ввести команды установки мандатных меток:

```
# MAC LABEL ON CLUSTER IS '{3,0}';
# MAC CCR ON CLUSTER IS OFF;
# MAC LABEL ON DATABASE jetsignal IS '{3,0}';
# MAC CCR ON DATABASE jetsignal IS OFF;
# MAC LABEL ON SCHEMA public IS '{3,0}';
# MAC CCR ON SCHEMA public IS OFF;
```

18) Выполнить команду выхода из программы psql:

```
# \q
```

19) Выполнить команду выхода из учётной записи пользователя postgres в сеанс пользователя root:

```
# exit
```

После выполнения этих операций созданная база данных **jetsignal** готова для работы с мандатной моделью управления доступом.

3.1.8 Настройка конфигурации узла

Для настройки выполнить следующие действия:

1) Скопировать пример файла конфигурации узлов Системы для экземпляра приложения:

```
# cp /var/www/jetsignal/config/unions.php.sample.simple
/var/www/jetsignal/config/unions.php
```

2) Открыть файл конфигурации узлов Системы на редактирование:

```
# nano /var/www/jetsignal/config/unions.php
```

Конфигурация узлов имеет следующую структуру:

– Узел, на который устанавливается Система. Признак узла **type** равен **self**: описывает параметры узла Системы.

– Другие узлы. Признак **type** не равен ‘**self**: описывает другие узлы Системы.

Для корректной работы приложения требуется обязательно указать следующие параметры:

– **unionId** – уникальный идентификатор узла в рамках развёртываемой Системы, взятый из организационного документа;

– **macLabel** – максимальный мандатный уровень доступа узла.

3.1.9 Формирование структуры базы данных

Для корректного формирования структуры базы данных предварительно должен быть правильно настроен файл конфигурации узла **unions.php**. Также в операционной системе обязательно должно быть корректно установлено время, так как механизм формирования идентификаторов записей в базе данных зависит от года, установленного в операционной системе.

Прежде всего необходимо включить механизм поддержки мандатных меток. Если в каталоге config отсутствует файл mac.php, то система считает, что механизм мандатных меток отключён и функционирует без их поддержки.

Для этого:

1) Создать файл конфигурации на основе примера:

```
# cp /var/www/jetsignal/config/mac.php.sample
/var/www/jetsignal/config/mac.php
```

2) Если необходимо, внести правки в файл конфигурации в формате языка PHP с учётом назначения параметров, используя текстовый редактор

3) # **nano /var/www/jetsignal/config/mac.php**:

– **enabled** – включён ли механизм мандатных меток, принимает значения true (включён) и false (выключен); для работоспособности системы с мандатными метками должен иметь значение true;

– **ccr** – включён ли механизм контроля целостности контейнера, принимает значения true (включён) и false (выключен); для работоспособности системы с мандатными метками должен иметь значение false;

– **maxLevel** – целочисленное значение максимального уровня мандатной метки, влияет на возможность объектов-контейнеров (кластеров, баз данных, схем, таблиц) хранить метки с уровнем до указанного включительно. Если требуется хранить в базе данных приложения объекты с меткой не выше 1 включительно, то следует установить значение 1. Если требуется хранить объекты с меткой 3 включительно, то следует установить значение 3.

– **levels** – массив в формате PHP, используется для определения доступных уровней и при их отображении в интерфейсе.

Для создания структуры БД или установки обновлений выполнить следующие действия:

1) Выполнить команду установки функций базы данных:

```
# yes | php /var/www/jetsignal/yii sequence/install
Are you sure? Connected database:
pgsql:host=127.0.0.1;port=6789;dbname=jetsignal (yes|no) [no]:Default
sequence-functions successfully created
```

2) Выполнить команду применения миграций:

```
# yes | php /var/www/jetsignal/yii migrate
```

Если имеются миграции для установки, Система выведет список доступных файлов миграций и будет выполнено применение миграций с выводом актуального состояния.

3) Выполнить команду установки последовательности по умолчанию для всех таблиц:

```
# yes | php /var/www/jetsignal/yii sequence/set-default
```

3.1.10 Установка мандатных меток на объекты базы данных

Для корректного функционирования системы с включённым режимом использования мандатных меток необходимо произвести установку мандатных меток **MAC LABEL** и сброс параметра **MAC CCR** для всех объектов базы данных.

1) После включения механизма мандатных меток требуется выполнить установку их на объекты базы данных.
2) Выполнить команду установки мандатных меток на объекты базы данных:

```
# yes | php /var/www/jetsignal/yii mac/install
Are you sure? (yes|no) [no]:Mac was successfully configured
```

3) Будет выполнена установка мандатных меток на объекты базы данных с учётом параметра

3.1.11 Установка наборов данных

Для первичной установки наборов данных либо их обновления выполнить следующие действия:

1) Выполнить команду установки основных наборов данных:

```
# php /var/www/jetsignal/yii seed/apply database_seeder install
```

Будет выполнена установка справочников, данных аутентификации (операций, ролей, групп) и других данных, требующихся для работы системы.

2) Выполнить команду создания подразделения по умолчанию:

```
# php /var/www/jetsignal/yii seed/apply admin_user install
```

Будет выполнено создание подразделения по умолчанию.

3) Если требуется установка отдельных наборов данных, то следует предварительно просмотреть список имеющихся наборов данных. Для этого ввести команду:

```
# php /var/www/jetsignal/yii seed/apply <имя_набора_данных> install
```

4) Чтобы посмотреть список доступных для установки наборов данных, ввести команду:

```
# ls /var/www/jetsignal /var/www/jetsignal/seeds/install
```

Появится список:

```
AuthAssignment.php  AuthItem.php  ChatGroup.php
IncidentCategory.php      IncidentReferences.php
InstructionReferences.php  OwnerType.php  RoleGroup.php
ScheduleOccasion.php  Template.php
AuthItemChild.php   Auth.php      CommonReferences.php
IncidentCriticalness.php  IncidentSource.php      InstructionType.php
Position.php   Role.php         ScheduleTemplate.php  Threat.php
AuthItemGroup.php   AuthRule.php  DatabaseSeeder.php
IncidentPriority.php      IncidentType.php        InstructionUrgency.php
Rank.php       RoleToGroup.php  SuperJetUser.php      WikiPage.php
```

5) Следует перевести имя файла из формата, например, WikiPage.php в формат wiki_page.

```
# php yii seed/apply wiki_page install
```

6) Нажать Enter.

Установка наборов данных выполняется по следующей схеме:

1) У каждого набора данных есть уникальный ключ для записи, по которому её можно идентифицировать (в зависимости от набора данных им может быть поле id, кортеж (id, union_id), кортеж (name, union_id) и т. д.).
2) Если по этому ключу запись найдена, то имеющиеся поля будут перезаписаны на поля, указанные в наборах данных.
3) Если по этому ключу запись не найдена, то будет создана новая запись.

При установке наборов данных существующие записи не удаляются.

Будет выполнена установка справочников, данных аутентификации (операций, ролей, групп) и других данных, требующихся для работы системы.

3.1.12 Создание и настройка учётной записи пользователя

Для первого входа в узел Системы, корректной работы HTTP Basic Authentication и функционирования мандатной модели управления доступом в Astra Linux SE 1.5 требуется создание служебной учётной записи администратора на уровне операционной системы и базы данных. В качестве такой служебной учётной записи используется учётная запись пользователя **admin.**

Общий алгоритм создания и настройки учётной записи пользователя выглядит следующим образом:

1) Создаётся учётная запись пользователя на уровне операционной системы.
2) Настраивается уровень мандатного доступа учётной записи пользователя.
3) Устанавливаются полномочия на чтение и выполнение файлов приложения, на запись системных журналов и временных файлов приложения.
4) Создаётся учётная запись пользователя на уровне кластера базы данных
5) Создаётся учётная запись пользователя в таблице user Системы, устанавливается роль и подразделение по умолчанию.

3.1.1.2 Автоматизированный механизм создания учётной записи пользователя

Для создания учётной записи выполнить следующие действия:

1) Выполнить команду создания учётной записи пользователя admin:

   # php /var/www/jetsignal/yii user/create
2) Ввести имя пользователя:

   Enter a username: admin1
9) Ввести пароль:

   Enter a user password: Ilikejet!
3) Ввести перечень ролей:

   Enter a comma separated roles list: admin
4) Ввести диапазон мандатных меток в формате X:Y, где X – минимальная мандатная метка пользователя, Y – максимальная мандатная метка пользователя (в качестве примера указан диапазон мандатных меток с 0 по 3):

   Enter a mac range: 0:3
5) Выполнить команду установки полномочий на объекты БД и файлы:

   # php /var/www/jetsignal/yii user/grant admin1
6) Система задаст вопрос, на который необходимо ввести yes и нажать клавишу Enter.
7) В результате выполнения команды должен быть следующий вывод:

   ```
   root@astra:/ # php /var/www/jetsignal/yii user/grant admin1
   Are you sure? (yes|no) [no]:yes
   Successfully granted
   ```
8) **Ручной механизм создания учётной записи пользователя возможет только после проведения работ по «первичному» развертыванию системы. См. Приложение В.**

3.2 Настройка RabbitMQ

1) Настройка Management console

```
rabbitmq-plugins enable rabbitmq_management
nano /etc/rabbitmq/rabbitmq.config
```

```
[{rabbit, [{loopback_users, []}]}].
```

2) Перевести RabbitMQ в режим compatibility

```
nano /etc/init.d/rabbitmq-server
#Заменить переменную PATH на слудующую
PATH=/usr/lib/parsec/bin:/sbin:/usr/sbin:/bin:/usr/bin
#Добавить в функцию start_rabbitmq () строку -- capability 0x100
start_rabbitmq () {
    status_rabbitmq quiet
    if [ $RETVAL != 0 ] ; then
        RETVAL=0
        ensure_pid_dir
        set +e
        RABBITMQ_PID_FILE=$PID_FILE start-stop-daemon --quiet \
            --chuid rabbitmq --start --exec $DAEMON \
            --pidfile "$RABBITMQ_PID_FILE" --background --capability
0x100
        $CONTROL wait $PID_FILE >/dev/null 2>&1
        RETVAL=$?
        set -e
        if [ $RETVAL != 0 ] ; then
            remove_pid
        fi
    else
        RETVAL=3
    fi
}
nano /etc/parsec/privsock.conf
#добавить следующие строки
/usr/lib/erlang/erts-5.9.1/bin/beam
/usr/lib/erlang/erts-5.9.1/bin/beam.smp
/usr/lib/erlang/erts-5.9.1/bin/child_setup
/usr/lib/erlang/erts-5.9.1/bin/dyn_erl
/usr/lib/erlang/erts-5.9.1/bin/epmd
/usr/lib/erlang/erts-5.9.1/bin/erl
/usr/lib/erlang/erts-5.9.1/bin/erlc
/usr/lib/erlang/erts-5.9.1/bin/erlexec
/usr/lib/erlang/erts-5.9.1/bin/escript
/usr/lib/erlang/erts-5.9.1/bin/heart
/usr/lib/erlang/erts-5.9.1/bin/inet_gethost
/usr/lib/erlang/erts-5.9.1/bin/run_erl
/usr/lib/erlang/erts-5.9.1/bin/start
/usr/lib/erlang/erts-5.9.1/bin/start_erl
/usr/lib/erlang/erts-5.9.1/bin/to_erl
```

3) Перезапустить сервер RabbitMQ

```
/etc/init.d/rabbitmq-server stop
/etc/init.d/rabbitmq-server start
```

4) Создать наборы Витруальных хостов и пользователей для системы

```
# cd /var/www/jetsignal
# rabbit_manage.sh
```

3.3 Создание пользователей интеграции

На каждом узле интеграции должны быть созданы пользователи интеграции. Число пользователей интеграции определяется максимальным мандатным уровнем узла.

- Для уровня 3 создаются пользователи int0, int1, int3. Где 0, 1, 3 – мандатная метка.
- Для уровня 1 создаются пользователи int0, int1. – где 0, 1 мандатная метка,
- Для уровня 0 создается пользователь int0.

Создание пользователя происходит на уровнях:

- Операционной системы
- Базы данных
- Назначение мандатной метки пользователю

Для заведения пользователя необходимо выполнить команду

 # php /var/www/jetsignal/yii user/create

Далее будет предложено заполнение полей:

1) Имя пользователя;
2) Пароль пользователя;
3) Роль пользователя (admin);
4) Мандатная метка, формат - 0,0 ; 0,1 ; 0,3 (см. Приложение А).

После этого создается запись в БД, пользователь на уровне БД и пользователь в ОС.

Если необходимо обновить пароль / роль / мандатную метку, необходимо вызвать команду:

 # php /var/www/jetsignal/yii user/set

и заполнить пункты 2), 3), 4)

3.4 Настройка интеграционной шины

Поле **type** определяет тип конфигурации: локальный экземпляр или территориально удаленный экземпляр. Допустимы три значения:

- **self** – конфигурация локального экземпляра Системы;
- **queue** – параметры связи с удаленным экземпляром Системы, интегрируемым с помощью сервера очередей;
- **file** – параметры связи с удаленным экземпляром Системы, интегрируемым с помощью файловой передачи.

3.4.1 Параметры для локальной системы self

Параметры, необходимые для работы Системы:

name – имя экземпляра Системы – участвует в формировании очередей.

unionId – id экземпляра Системы. При первоначальном развёртывании экземпляра идентификатор записывается в БД и является ключевой настройкой экземпляра Системы. Принимает целое значение не менее 2. Недопустимо наличие в системе двух и более экземпляров Систем с одинаковыми **unionId.**

mqConnections – массив параметров подключение к виртуальным хостам сервера очередей. Внутри массива записи идентифицируются по ключу, которым является мандатная метка.

macLevelLabel – мандатная метка, допустимые значения 0, 1, 3. Приложение определяет мандатную метку передаваемых данных и выбирает по ней подключение к VirtualHost сервера очередей.

mqHost, mqPort, mqUsername, mqPassword, mqVirtualHost – определяют настройки подключения к серверу очередей. Для типа self – это локальный сервер очередей, для типа queue – удаленный сервер очередей интегрируемой Системы.

mqInbound – имя входящей обрабатываемой очереди, в которую после маршрутизации подступают все сообщения, обрабатываемые Системой.

folder – полный путь к каталогу интеграционного модуля Системы, предназначенному для хранения журналов регистрации (логи) В каталоге должен быть создан каталог /logs и набор подкаталогов 0. 1. 3 для хранения логов всех мандатных уровней.

integrationFolder – полный путь к основному каталогу интеграционного модуля Системы. В нем должны быть созданы папки для входящих файлов и обработанных файлов. А именно:

./stage – хранение файла в процессе обработки в интеграционном модуле

./archive – перемещение файла после обработки

./error – сохранение файла в случае ошибки в обработке

А так же набор подкаталогов в каждом из указанных каталогов: 0, 1 для хранения файлов требуемых мандатных уровней.

fileServerBuffer – полный путь к каталогу, в котором интеграционный модуль временно хранит файлы-вложения до перемещения в каталог storage, которое происходит при получении приложением интеграционного сообщения. Предлагается использовать путь к основному каталогу интеграции + /upload. **fileServerBuffer** не может быть расположен вне основного каталога приложения.

integrationSourceFolder – полный путь к каталогу, в котором интеграционный модуль ожидаем получать входящие сообщения в виде файлов, переданных через однонаправленный шлюз. Фактически данный каталог должен содержать набор подкаталогов ./0, ./1 – для разграничения файлов по мандатным меткам. Однако, в настройке подкаталоги **integrationSourceFolder** 0, 1 указывать нельзя. Приложение определит требуемый подкаталог для опроса в зависимости от мандатной метки запущенного процесса.

dbBufferReadRows – число записей, единовременно поступающих на отправку. Этот параметр используется в процессе отправки накопленных за день изменений сущностей, по которым не ведётся история изменений.

filePollingInterval – интервал опроса папки на входящие сообщения, который выполняет файловый адаптер (в секундах).

minimumFileAge – «возраст» (в секундах): текущая дата минус дата последнего изменения файла. Параметр, необходимый для того, чтобы в обработку не попадали файлы, находящиеся в процессе копирования.

minimumAttachmentAge - возраст» (в секундах): текущая дата минус дата последнего изменения файла. Параметр, необходимый для того, чтобы в обработку не попадали файлы-вложения, находящиеся в процессе копирования. Должен быть не менее в чем 2 раза меньше чем **minimumFileAge.** Условие требуется для обеспечения порядка обработки – сначала файлы-вложения, затем соответствующие им файлы – интеграционные сообщения.

fileWaitTimeout – параметр определяющий какое время следует ожидать файла-вложения, если файл – интеграционное сообщение пришел раньше.

pollingEmailInterval – интервал (в секундах) опроса папки на входящие сообщения, который выполняет почтовый адаптер.

emailServer, emailPort, emailUser, emailPass – параметры подключения к Email-серверу для обработки входящих сообщений от SIEM.

sourceSiem – имя SIEM-системы, с которой интегрируется приложение – оно определяет правила трансформации. На основании этой настройки определяется, из какого формата будет выполняться трансформация во внутренний формат экземпляра Системы.

В существующей версии допускается значение **SIEM** – означает, что сообщения будут передаваться без трансформации в формате JSON. Значение поля **sourceSiem** должно совпадать с одним из значений справочника **Интегрируемые системы** в экземпляре Системы.

На основе этого параметра также выполняется перекодировка значений справочников.

relation – отношение главенства системы. Для type = self, relation всегда должен быть равен self

integrationMode – режим работы интеграционного модуля. Допустимые значения: queue, file.

queue – интеграция с использованием сервера очередей RabbitMQ (значение по умолчанию)

file – интеграция с использованием только файлового механизма обмена. Перемещение файлов между узлами системы должно быть обеспечено скриптами, разделяемым ресурсом или иными механизмами.

chatIntegrationMode – способ интеграции сообщений чата. Допустимые значения: queue, http.

queue – интеграция с использованием сервера очередей RabbitMQ (значение по умолчанию)

http – интеграция с использованием прямой передачи по http. (Быстрее, чем queue, однако, гарантированная доставка не обеспечивается)

3.4.2 Параметры для экземпляров Системы, интегрируемых с помощью очередей (тип queue)

Параметры, необходимые для интеграции:

type => **queue** – определяет тип удаленного экземпляра Системы, который интегрируется с помощью очередей.

mqConnections – массив параметров подключение к виртуальным хостам сервера очередей. Внутри массива записи идентифицируются по ключу, которым является мандатная метка.

macLevelLabel – мандатная метка, допустимые значения 0, 1, 3. Приложение определяет мандатную метку передаваемых данных и выбирает по ней подключение к VirtualHost сервера очередей.

mqHost, mqPort, mqUsername, mqPassword, mqVirtualHost – определяют настройки подключения к серверу очередей. Для типа **self** – это локальный сервер очередей, для типа queue – удаленный сервер очередей интегрируемого экземпляра.

name – имя Системы.

unionId – id Системы.

mqInbound – имя удаленной очереди входящих сообщений для экземпляра Системы. Важно, чтобы этот параметр совпадал с настройкой **mqInbound (type=self)** того удаленного экземпляра Системы, с которым интегрируется локальный экземпляр.

isBroadcast – булево поле. Признак того, что в экземпляр Системы нужно передавать данные при широковещательной рассылке по цепочке.

fileServerAddress – IP-адрес экземпляра Системы

fileServerBuffer – относительный путь от корня экземпляра Системы до буферной папки, в которой интеграционный модуль хранит файлы в течение сеанса обмена данными.

fileServerUsers – массив логинов и паролей пользователей интеграции под которыми происходит аутентификация при передаче файлов-вложений по протоколу http между узлами системы.

macLevelLabel – мандатная метка, допустимые значения 0, 1, 3. Приложение определяет мандатную метку передаваемых данных и выбирает по ней логин и пароль пользователя.

username, password– логин и пароль пользователя на удаленной машине, у которого есть права доступа в Систему.

relation – отношение главенства системы. Допустимые значения: parent, child.

parent – в узел, отношение которого – родитель, отправляются все инциденты для информации. На узел, являющийся родителем допустимо назначение инцидента на исполнение.

child – в узел, отношение которого – ребенок, допустима отправка распорядительных документов.

3.4.3 Параметры для удаленной системы типа file

Параметры, необходимые для интеграции с удаленной Системой:

type => **file** – определяет тип удаленного экземпляра Системы, который интегрируется с помощью файлов.

name – имя Системы

folder – каталог файлового моста на локальной машине, который обеспечивает обмен между локальной и удаленной машиной.

isBroadcast – признак того, что в Систему нужно передавать данные при широковещательной рассылке по цепочке.

relation – отношение главенства системы. Допустимые значения: parent

Изменение любых параметров требует перезагрузки служб интеграционного модуля.

3.4.4 Настройка каталогов

Для настройки требуется:

1) Перед запуском служб интеграционного модуля создать необходимые для него каталоги. Для этого нужно создать основной каталог интеграционного модуля, в котором должны быть созданы подкаталоги:

– archive;

– error;

- logs;
- source;
- stage;
- upload.

А так же набор подкаталогов в каждом из указанных каталогов: 0, 1, 3 для хранения файлов требуемых мандатных уровней. Количество подкаталогов 0, 1, 3 определяется максимальным мандатным уровнем узла. Так для узла открытого контура нужно создавать только подкаталог 0.

На каждый интеграционный каталог под пользователем root последовательно нужно назначить правила хранения файлов разных мандатных меток с помощью команды pdpl-file:

Пример для папки «logs» в системе с максимальным мандатным уровнем 3:

pdpl-file -v 3:0:0:ccnr integration

pdpl-file -v 3:0:0:ccnr integration/logs

pdpl-file -v 3:0:0:ccnr integration/logs/3

pdpl-file -v 1:0:0:ccnr integration/logs/1

pdpl-file -v 0:0:0:ccnr integration/logs/0

*Предлагается для настройки **fileServerBuffer** использовать папку **upload** основного интеграционного каталога.*

2) Установить на интеграционный каталог права 777 рекурсивно.

3.5 Запуск интеграционных процессов

3.5.1 Общая информация о запуске интеграционных процессов

Интеграционные процессы состоят из 4-х процессов:

- Получение входящих файлов: **filelistener**
- Передача сообщений в другие узлы: **broker**
- Сохранение входящих сообщений из очереди**: replicator**
- Получение входящих email-сообщений: **emaillistener**

Каждый из процессов должен быть запущено под пользователем интеграции обладающим требуемой мандатной меткой. Мандатная метка пользователя приобретается при проведении операции login в операционной системе. Запуск процессов может быть автоматизирован с помощью скрипта integrationStart.sh. Операция login не может быть автоматизирована.

Просмотр запущенных процессов выполнятся командой:

ps aux | grep yii

3.5.2 Запуск передачи сообщений в другие узлы

Компонент Broker запускается командой

nohup php /var/www/jetsignal/yii integration/integration/broker online >> integration/logs/0/out &

Integration broker /var/www/jetsignal/modules/integration/controllers/IntegrationController.php runs in online mode.

Exception 'PhpAmqpLib\Exception\AMQPRuntimeException' with message 'Broken pipe or closed connection'

in /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Wire/IO/StreamIO.php:207

Stack trace:

#0 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Wire/AMQPReader.php(149): PhpAmqpLib\Wire\IO\StreamIO->read(7)

#1 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Wire/AMQPReader.php(106): PhpAmqpLib\Wire\AMQPReader->rawread(7)

#2 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Connection/AbstractConnection.php(508): PhpAmqpLib\Wire\AMQPReader->read(7)

#3 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Connection/AbstractConnection.php(555): PhpAmqpLib\Connection\AbstractConnection->wait_frame(0)

#4 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Channel/AbstractChannel.php(217): PhpAmqpLib\Connection\AbstractConnection->wait_channel(0, 0)

#5 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Channel/AbstractChannel.php(328): PhpAmqpLib\Channel\AbstractChannel->next_frame(0)

#6 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Connection/AbstractConnection.php(213): PhpAmqpLib\Channel\AbstractChannel->wait(Array)

#7 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Connection/AbstractConnection.php(180): PhpAmqpLib\Connection\AbstractConnection->connect()

#8 /var/www/jetsignal/vendor/php-amqplib/php-amqplib/PhpAmqpLib/Connection/AMQPStreamConnection.php(60): PhpAmqpLib\Connection\AbstractConnection->__construct('int0', 'P@ssw0rd', 'o', false, 'AMQPLAIN', NULL, 'en_US', Object(PhpAmqpLib\Wire\IO\StreamIO), 0)

#9 /var/www/jetsignal/modules/integration/adapters/MQAdapter.php(79): PhpAmqpLib\Connection\AMQPStreamConnection->__construct('127.0.0.1', '5672', 'int0', 'P@ssw0rd', 'o')

#10 /var/www/jetsignal/modules/integration/adapters/MQAdapter.php(43): app\modules\integration\adapters\MQAdapter->initiateQueues(Array, '127.0.0.1', '5672', 'int0', 'P@ssw0rd', 'o')

#11 /var/www/jetsignal/modules/integration/logic/Broker.php(121): app\modules\integration\adapters\MQAdapter->__construct('HI-ALL', Array)

#12 /var/www/jetsignal/modules/integration/controllers/IntegrationController.php(125): app\modules\integration\logic\Broker->start('online')

#13 [internal function]: app\modules\integration\controllers\IntegrationController->actionBroker('online')

#14 /var/www/jetsignal/vendor/yiisoft/yii2/base/InlineAction.php(57): call_user_func_array(Array, Array)

#15 /var/www/jetsignal/vendor/yiisoft/yii2/base/Controller.php(156): yii\base\InlineAction->runWithParams(Array)

#16 /var/www/jetsignal/vendor/yiisoft/yii2/console/Controller.php(128): yii\base\Controller->runAction('broker', Array)

#17 /var/www/jetsignal/vendor/yiisoft/yii2/base/Module.php(523): yii\console\Controller->runAction('broker', Array)

#18 /var/www/jetsignal/vendor/yiisoft/yii2/console/Application.php(180): yii\base\Module->runAction('integration/int...', Array)

#19 /var/www/jetsignal/vendor/yiisoft/yii2/console/Application.php(147): yii\console\Application->runAction('integration/int...', Array)

#20 /var/www/jetsignal/vendor/yiisoft/yii2/base/Application.php(380): yii\console\Application->handleRequest(Object(yii\console\Request))

#21 /var/www/jetsignal/yii(20): yii\base\Application->run()

#22 {main}

Где 0 – мандатная метка пользователя, запустившего процесс.

На узле, в котором, не требуется получение сообщений, может быть запущен только один процесс – Broker.

Для передачи данных под мандатной меткой 1, процесс должен быть запущен под пользователем int1.

Для передачи данных под мандатной меткой 3, процесс должен быть запущен под пользователем int3.

3.5.3 Получение входящих файлов

Компонент Filelistener запускается командой

nohup php /var/www/jetsignal/yii integration/integration/filelistener >> integration/logs/0/out &

Где 0 – мандатная метка пользователя, запустившего процесс.

На узлах системы, в которых не производится обработка входящих файлов не требуется запускать данный компонент.

Для получения файлов под мандатной меткой 1, процесс должен быть запущен под пользователем int1.

3.5.4 Сохранение сообщений из входящей очереди

Компонент Replicator запускается командой

nohup php /var/www/jetsignal/yii integration/integration/replicator >> integration/logs/0/out &

Где 0 – мандатная метка пользователя, запустившего процесс.

Replicator производит прослушивание сообщений из очереди входящих сообщений и сохранение в БД приложения.

Для репликации данных под мандатной меткой 1, процесс должен быть запущен под пользователем int1.

Для репликации данных под мандатной меткой 3, процесс должен быть запущен под пользователем int3.

3.5.5 Запуск обработки входящих Email-сообщений

Для этого нужно перейти в корневой каталог развернутого экземпляра Системы и запустить выполнение команды:

nohup php /var/www/jetsignal/yii integration/integration/emaillistener >> integration/ logs/0/out &

Где 0 – мандатная метка пользователя, запустившего процесс.

Эта команда запустит обработчик Email-сообщений.

На одном узле системы может быть запущен только один обработчик email-ов единовременно.

Система принимает сообщения в формате доставки: простой текст.

Кодировка: UTF-8.

Content-Transfer-Encoding: base64, 8bit.

3.5.6 Управление сервером очередей RabbitMQ

Текущий статус RabbitMQ:

/etc/init.d/rabbitmq-server status

Команда запуска:

/etc/init.d/rabbitmq-server start

Команда остановки:

/etc/init.d/rabbitmq-server stop

Команда перезагрузки:

/etc/init.d/rabbitmq-server restart

3.6 Настройка опциональных возможностей Системы

3.6.1 Настройка схемы движения инцидента

Схема движения инцидента может быть изменена в соответствии с потребностями процесса обработки инцидента.

Внимание! Настройка конфигурационных файлов схемы движения инцидентов может нарушить работоспособность приложения, поэтому должна проводиться с резервным копированием изменяемых файлов и с учётом возможного времени восстановления системы.

Схема движения инцидента описана в конфигурационных файлах модуля **`infosec`**:

- **`/var/www/jetsignal/modules/infosec/config/incidentStatusMatrix.php`** – матрица переходов, описывающая возможности перехода из одного статуса в другой статус;
- **`/var/www/jetsignal/modules/infosec/config/incidentDependencies.php`** – перечень действий и условий, которые должны выполняться при переходе из определённых статусов в другие определённые статусы.

Конфигурационный файл матрицы переходов между статусами имеет следующий вид:

```
<?php
return [
    'highCriticalness' => [
        [0,1,0,0,0,0,0,0,0], // 1 ST_NEW
        [0,0,1,1,0,0,0,0,0], // 2 ST_IN_PROGRESS
        [0,0,0,0,0,1,0,0,0], // 3 ST_DECLINED
```

```
            [0,0,0,0,0,1,0,0,0], // 4 ST_COMPLETED
            [0,1,0,0,0,0,0,0,0], // 5 ST_RETURNED
            [0,0,0,0,1,0,1,0,0], // 6 ST_ANALYSIS
            [0,0,0,0,0,0,0,1,0], // 7 ST_ON_AUDIT
            [0,0,0,0,1,0,0,0,1], // 8 ST_AUDIT
            [0,0,0,0,0,0,0,0,0], // 9 ST_CLOSED
        ],
        'lowCriticalness' => [
            [0,1,0,0,0,0,0,0,0], // 1 ST_NEW
            [0,0,1,1,0,0,0,0,0], // 2 ST_IN_PROGRESS
            [0,0,0,0,0,1,0,0,0], // 3 ST_DECLINED
            [0,0,0,0,0,1,0,0,0], // 4 ST_COMPLETED
            [0,1,0,0,0,0,0,0,0], // 5 ST_RETURNED
            [0,0,0,0,1,0,0,0,1], // 6 ST_ANALYSIS
            [0,0,0,0,0,0,0,0,0], // 7 ST_ON_AUDIT
            [0,0,0,0,0,0,0,0,0], // 8 ST_AUDIT
            [0,0,0,0,0,0,0,0,0], // 9 ST_CLOSED
        ],
    ];
```

В конфигурационном файле описывается поведение для двух уровней критичности (влияющих на сценарий обработки заявки): высокого (**highCriticalness**) и низкого (**lowCriticalness**). Уровни критичности описаны на уровне исходного кода приложения и не редактируются через конфигурацию.

Каждая строка матрицы, например:

```
[0,1,0,0,0,0,0,0,0], // 1 ST_NEW
```

описывает возможности перехода из одного статуса в другой. Ноль означает невозможность перехода, единица – возможность перехода.

На примере строки **ST_NEW**: описывается возможность перехода из статуса **ST_NEW** («Новый») в статус **ST_IN_PROGRESS** («В работе»).

Имеется возможность изменить матрицу переходов. Для этого нужно поменять значения в матрице либо на 0 (переход невозможен), либо на 1 (переход возможен).

Конфигурационный файл перечня действий и условий, которые выполняются при переходе между статусами, имеет следующий вид:

```
<?php
return [
    'status' => [[
        'type' => 'user',
        'new' => [\app\common\logic\orm\Incident::ST_IN_PROGRESS,
\app\common\logic\orm\Incident::ST_ANALYSIS,
\app\common\logic\orm\Incident::ST_AUDIT],
        'relation' => 'owner',
        'override' => true,
        'eventType' => \app\common\db\ChangeAttributeEvent::TYPE_BEFORE,
    ], [
```

```
        'type' => 'department',
        'new' => [\app\common\logic\orm\Incident::ST_IN_PROGRESS,
\app\common\logic\orm\Incident::ST_ANALYSIS,
\app\common\logic\orm\Incident::ST_AUDIT],
        'override' => true,
        'eventType' => \app\common\db\ChangeAttributeEvent::TYPE_BEFORE,
    ], [
        'type' => 'relation',
        'new' => \app\common\logic\orm\Incident::ST_RETURNED,
        'relation' => 'owner',
        'relationValue' => null, // сброс ответственного
        'eventType' => \app\common\db\ChangeAttributeEvent::TYPE_BEFORE,
    ]],
];
```

В этом конфигурационном файле верхнеуровневое поле ‘**status**’ выступает в качестве анализируемого параметра.

Уровнем ниже у каждой записи указаны следующие параметры:

- **type –** обозначает тип зависимости. Доступны типы:
  - **department** (работа с подразделением),
  - **callback** (вызов функции-замыкания),
  - **integration** (вызов интеграции),
  - **method** (вызов метода класса),
  - **timestamp** (работа с меткой времени),
  - **user** (работа с пользователем).
- **old –** описывает набор состояний, при переходе из которых срабатывает условие (в примере не указаны). Содержит массив текстовых значений (в примере текстовые значения берутся из констант класса **Incident**).
- **new** – описывает набор состояний, при переходе в которые срабатывает условие. Содержит массив текстовых значений.
- **relation –** описывает наименование отношения, в которое будет записан результат.
- **override** – указывает, можно ли перезаписать установленное значение либо только установить новое, если еще не установлено. Принимает значения true и false.
- **relationValue** – описывает строго устанавливаемое значение.
- **eventType** – описывает событие, при котором срабатывает данная зависимость, в соответствии с константами **TYPE_*** класса **\app\common\db\ChangeAttributeEvent**.

Возможна следующая настройка:

1) Добавление новых зависимостей при переходе между статусами инцидента.

2) Изменение набора статусов, на которые оказывает влияние добавление зависимости.

Устанавливаемые зависимости должны соответствовать матрице переходов между статусами.

3.6.2 Настройка однонаправленного сетевого шлюза

Имеется возможность настройки однонаправленного сетевого шлюза между узлами Системы, работающей в режиме распределённой информационной Системы.

Предварительно должны быть обеспечены следующие условия:

- Отсутствие возможности для двустороннего взаимодействия между настраиваемыми узлами по другим каналам связи.
- Отсутствие потребности в двустороннем обмене информации с обязательным подтверждением доставки данных.
- Наличие компьютера с операционной системой Windows не ниже версии 7 для обеспечения формирования файла конфигурации шлюза.

Для настройки шлюза требуется:

1) Ввести конфигурацию в шлюз с SD-карты.

Конфигурация представляет собой бинарный файл с расширением *.cfg, который должен быть расположен «в корне» SD-карты. Администратор должен составить конфигурацию в текстовом виде с использованием языка сценария. Преобразование из тестового вида в бинарный с расширением *.cfg выполняется с помощью утилиты zc.exe.

Требования к SD-карте.

a) Тип файловой системы: FAT32.

b) Бинарный конфигурационный файл должен:

- иметь расширение *.cfg
- располагаться «в корне» SD-карты.

Команды языка сценария должны вводиться в нижнем регистре, так как большие буквы не распознаются.

2) Указать команду установки MAC-адреса шлюза в открытой сети:

```
# wan mac xx:xx:xx:xx:xx:xx
```

Пример:

```
wan mac 01:aa:bb:45:00:ff
```

Внимание! Стоит избегать ввода широковещательных и многоадресных MAC-адресов для исключения некорректной работы в сети.

3) Указать команду установки разрешённого IP-адреса открытой сети:

```
# ip permit IPsrc
```

где IPsrc – IP-адрес.

Необходимо иметь в виду следующие положения:

– Пакеты с указанного IP-адреса могут поступать в закрытую сеть.

– Разрешенных IP-адресов может быть несколько.

– Несколько разрешенных IP-адресов образуют список разрешенных IP-адресов.

– Максимальное количество IP-адресов – 511.

– IP-адреса могут быть из разных подсетей.

Пример:

```
ip permit 192.168.1.1
ip permit 192.168.1.2
ip permit 195.0.0.1
```

4) Указать команду установки правила маршрутизации в закрытой сети:

```
# route IPdst_open IPdst_private MACdst_private
```

Необходимо иметь в виду следующие положения:

– IP-пакет из внешней сети с IP-адреса назначения IPdst_open может попасть в закрытую сеть с IP назначения IPdst_private на MAC-адрес MACdst_private.

– Правил может быть несколько.

– Несколько правил образуют таблицу маршрутизации.

– Максимальное количество записей в таблице – 511.

Пример:

```
route 192.168.1.5 10.8.0.1 00:00:00:00:00:01
route 192.168.1.6 10.8.0.1 00:00:00:00:00:02
```

– Пакет попадет в закрытую сеть только в том случае, если IP-адрес источника пакета присутствует в списке разрешенных IP-адресов, а IP-адрес назначения присутствует в таблице маршрутизации.

3.6.3 Настройка журнала приложения

Конфигурация файлового журнала приложения находится в файлах:

– **`config/web.php`** – конфигурация приложения для веб-сервера;

– **config/console.php** – конфигурация приложения для консольных вызовов (в том числе для интеграции),

и описана в блоке **log** раздела **components** (пример рекомендуемой конфигурации журнала):

```
'log' => [
    'traceLevel' => YII_DEBUG ? 3 : 0,
    'targets' => [
        [
            'class' => 'yii\log\FileTarget',
            'levels' => ['error', 'warning'],
        ],
        [
            'class' => 'app\log\DbTarget',
            'levels' => [
                'error',
                'warning',
                'info',
            ],
            'except' => [
                'yii\db\Command::query',
                'yii\db\Connection::open',
                'yii\web\Session::open',
            ],
        ],
    ],
],
```

Журнал может записываться в несколько адресов. В указанном примере настроена запись журнала в два адреса:

– **yii\log\FileTarget** – запись журнала приложения в файл;

– **app\log\DbTarget** – запись журнала приложения в базу данных.

При настройке можно оперировать следующими параметрами:

– **levels** – записываемые в журнал уровни;

– **categories** – записываемые в журнал категории сообщений (если не указано строго, то записываются все категории указанных уровней);

– **except** – исключения из категорий (если категория находится в этом списке, она не будет попадать в журнал).

Уровень – признак критичности сообщения. Имеются следующие уровни:

– **error** – критичная ошибка, оказывающая влияние на работоспособность приложения и потенциально блокирующая выполнение отдельных функций либо всего приложения.

– **warning** – некритичная ошибка, оказывающая влияние на работоспособность приложения, но не блокирующая выполнение функций приложения.

- **info –** информационное сообщение.
- **trace** – отладочный режим. Содержит в себе сообщения обо всех стеках вызова функций; может использоваться для диагностики приложения.
- **profile** – режим профилирования приложения. Используется для оценки времени выполнения отдельных функций и поиска «узких мест».

Категория – условный признак типа сообщения в свободной форме. Часто в качестве категории используется имя класса, формирующего ошибку.

Наиболее часто встречающиеся категории указаны в разделе 6.2.

3.6.4 Сброс мандатных меток пользователя

Для сброса мандатного уровня и категории следует использовать команду:

```
# pdpl-user –z admin
```

Пример корректного вывода данной команды:

```
root@astra:~# pdpl-user -z admin
minimal level:  Уровень_0(0)
maximal level:  Уровень_0(0)
maximal integrity level:        Low(0)
minimal category:       0x0(0)
maximal category:       0x0(0)
```

4 Проверка программы

4.1 Проверка корректности установки программы

4.1.1 Проверка корректности установки и настройки PHP

4.1.1.1 Проверка корректности версии PHP

Для проверки:

1) Ввести команду проверки версии PHP:

```
# php –v
```

2) Нажать клавишу **Enter**.

Должна появиться следующая информация о версии:

```
root@astra:/var/www/jetsignal # php -v
PHP 5.4.4-2astra2 (cli) (built: Mar  1 2016 21:32:20)
Copyright (c) 1997-2012 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2012 Zend Technologies
    with XCache v2.0.0, Copyright (c) 2005-2012, by mOo
```

4.1.1.2 Проверка корректности состава модулей PHP

Для проверки:

1) Ввести команду просмотра состава модулей PHP:

```
# php –m
```

2) Нажать клавишу **Enter**.

Должен быть выведен следующий набор модулей PHP:

```
root@astra:/var/www/jetsignal# php -m
[PHP Modules]
bcmath
bz2
calendar
Core
ctype
curl
date
dba
dom
ereg
exif
fileinfo
filter
ftp
gd
gettext
hash
```

```
iconv
imap
json
libxml
mbstring
mhash
openssl
pcntl
pcre
PDO
pdo_pgsql
pgsql
Phar
posix
Reflection
session
shmop
SimpleXML
soap
sockets
SPL
standard
sysvmsg
sysvsem
sysvshm
tokenizer
wddx
XCache
xml
xmlreader
xmlwriter
zip
zlib

[Zend Modules]
XCache
```

Следует обратить внимание, что среди установленных модулей нет модуля intl; его использование вызывает ошибки при работе с датой и временем в PHP 5.4.4.

4.1.2 Проверка корректности установки и настройки Apache2

4.1.2.1 Проверка корректности версии Apache2

Для проверки:

1) Ввести команду проверки версии Apache2:

```
# apachectl –v
```

2) Нажать клавишу **Enter**.

Должна отобразиться информация о версии Apache2:

```
root@astra:~# apachectl -v
Server version: Apache/2.2.22 (Debian)
Server built:   Mar  1 2016 23:53:10
```

4.1.2.2 Проверка корректности состава модулей Apache2

Для проверки:

1) Ввести команду просмотра состава модулей Apache2:

```
# apachectl –M
```

2) Нажать клавишу **Enter**.

Должен отобразиться следующий набор модулей Apache2:

```
root@astra:~# apachectl -M
Loaded Modules:
 core_module (static)
 log_config_module (static)
 logio_module (static)
 version_module (static)
 mpm_prefork_module (static)
 http_module (static)
 so_module (static)
 alias_module (shared)
 auth_basic_module (shared)
 auth_pam_module (shared)
 authn_file_module (shared)
 authz_default_module (shared)
 authz_groupfile_module (shared)
 authz_host_module (shared)
 authz_user_module (shared)
 autoindex_module (shared)
 cache_module (shared)
 cgi_module (shared)
 deflate_module (shared)
 dir_module (shared)
 env_module (shared)
 file_cache_module (shared)
 mem_cache_module (shared)
 mime_module (shared)
 negotiation_module (shared)
 php5_module (shared)
 reqtimeout_module (shared)
 rewrite_module (shared)
 setenvif_module (shared)
 status_module (shared)
Syntax OK
```

4.1.2.3 Проверка конфигурации виртуальных хостов

Для проверки:

1) Ввести команду тестирования конфигурации Apache2:

```
# apachectl configtest
```

2) Нажать клавишу **Enter**.

Должен отобразиться следующий набор атрибутов:

```
root@astra:~# apachectl configtest
Syntax OK
```

Если в конфигурации Apache2 имеются какие-либо ошибки, появится список обнаруженных ошибок.

4.1.2.4 Проверка корректности используемых портов Apache2

Для проверки запустить команду:

```
# netstat -anp | grep apache
```

Отобразится перечень портов, которые использует Apache2. Для корректного функционирования Apache2 требуются как минимум порты 80 (без шифрования SSL) и 443 (с шифрованием SSL).

```
root@astra:/# netstat -anp | grep apache
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
8447/apache2
tcp        0      0 ***.***.***.***:80      ***.***.***.***:51153
ESTABLISHED 11091/apache2
tcp        0    835 ***.***.***.***:80      ***.***.***.***:5432
ESTABLISHED 13295/apache2
unix  2      [ ]         DGRAM                    5246645  11091/apache2
unix  2      [ ]         DGRAM                    5251730  12366/apache2
unix  2      [ ]         DGRAM                    5255941  13126/apache2
unix  2      [ ]         DGRAM                    5253931  13295/apache2
```

4.1.3 Проверка настройки учётных записей пользователя

4.1.3.1 Проверка мандатного уровня доступа пользователя

Для проверки существующего уровня мандатного доступа пользователя:

1) Ввести команду без дополнительных атрибутов:

```
# usermac <имя пользователя>
```

2) Нажать клавишу **Enter**.

Отобразится перечень мандатных разрешений пользователя, например, для пользователя admin:

```
root@astra:~# usermac admin
minimal level:  Уровень_0(0)
maximal level:  Уровень_0(0)
maximal integrity level:        Low(0)
minimal category:       0x0(0)
maximal category:       0x0(0)
```

4.1.3.2 Проверка наличия учётной записи в базе данных

Для проверки:

1) В командной строке переключиться в сеанс пользователя postgres:

```
# su postgres
```

2) Запустить программу psql с параметром базы данных приложения:

```
# psql -d jetsignal
```

3) Запустить выполнение запроса на получение списка ролей:

```
# SELECT rolname FROM pg_roles;
```

Должен появиться перечень имеющихся ролей:

```
jetsignal =# SELECT rolname FROM pg_roles;
   rolname
-------------
 postgres
 test
 backup
 root
 admin
 (5 rows)
```

4) Выйти из программы и из сеанса пользователя postgres:

```
# \q
# exit
```

4.2 Проверка опциональных возможностей системы

4.2.1 Проверка работоспособности однонаправленного шлюза

Для проверки:

1) На сервере №1 с IP 192.168.1.1 ввести команду:

```
# ping 192.168.1.5
```

Если всё правильно, на шлюзе будет увеличиваться счетчик пакетов. Это значит, что пакеты проходят через шлюз во внутреннюю сеть.

2) Если счетчик пакетов не увеличивается, то с консоли посмотреть arp-таблицу:

```
# arp –a
```

В таблице должна присутствовать запись:

```
192.168.1.5 -------------- 11-12-12-01-02-03
```

Для этой записи необходим wan MAC-адрес в конфигурационном файле.

Если такой записи нет, то, возможно, проблема связана с сетью и требуется её повторная настройка и диагностика.

4.2.2 Включение и выключение режима отладки приложения

Режим отладки приложения позволяет выполнять следующие функции:

1) Создание учётной записи пользователя в веб-интерфейсе без входа в систему.
2) Вывод сообщений об ошибке на экран (источник ошибок «Браузер»).

Внимание! Включение режима отладки потенциально позволяет обойти встроенную в приложение систему аутентификации (путём создания новой учётной записи пользователя) и получить доступ к текстам ошибок приложения с выводом на экран имён каталогов, файлов и прочей информации, предназначенной для системного программиста. В связи с этим при переводе приложения в этот режим должны быть предприняты должные меры защиты системы организационного характера.

1) Для включения режима отладки ввести команду:

   ```
   # nano /var/www/jetsignal/web/index.php
   ```

   Файл будет открыт на редактирование.

2) В строке:

   ```
   defined('YII_ENV') or define('YII_ENV', ‘prod’);
   ```

   поменять prod на dev.

   Строка примет следующий вид:

   ```
   defined('YII_ENV') or define('YII_ENV', ‘dev’);
   ```

3) В строке:

   ```
   defined('YII_DEBUG') or define('YII_DEBUG', false);
   ```

   поменять false на true.

   Строка примет следующий вид:

```
defined('YII_DEBUG') or define('YII_DEBUG', true);
```

Режим отладки включён, если после прохождения базовой аутентификации HTTP Basic отображается веб-форма входа с кнопкой «Добавить пользователя».

Для отключения режима отладки следует проделать шаги 1–3 с возвращением указанных параметров к исходным значениям.

5 Дополнительные возможности

5.1 Механизм установки наборов данных

Механизм установки наборов данных позволяет установить в базу данных приложения данные, необходимые для его корректной работы. Этот механизм следует использовать в следующих случаях:

1) Первичное развёртывание приложения с использованием «чистой» базы данных.
2) Актуализация данных при развёртывании системы с использованием базы данных с уже имеющимися данными (случаи восстановления системы из резервной копии, перенос системы на другое техническое средство и т. д.).
3) Актуализация данных при поставке обновления приложения. В этом случае сведения о необходимости применения операции установки наборов данных будут указаны в инструкции по установке обновления.
4) Восстановление повреждённых наборов данных. Может потребоваться для восстановления учётной записи администратора системы, восстановления конфигурации ролей и операций, повреждённых в процессе ручной настройки приложения из пользовательского интерфейса.
5) Экспорт наборов данных переноса вручную в другую систему.

Для применения механизма установки наборов данных следует использовать команду **`php yii seed`**. Для использования механизма должен быть корректно установлен и настроен пакет **`php5-cli`**.

Чтобы просмотреть список доступных команд, требуется:

1) Для корневого каталога приложения (для примера – **`/var/www/jetsignal`**) ввести команду:

   ```
   # php yii seed –help
   ```

2) Нажать клавишу **Enter**.

   Отобразится список доступных операций:

   ```
   root@astra:/var/www/jetsignal # php yii seed --help

   Class SeedController

   SUB-COMMANDS

   - seed/apply (default)  @param string       $className
   - seed/create           @param string       $className
   - seed/export           @param string             $models
   - seed/list             @param null|int           $limit
   ```

```
To see the detailed information about individual sub-commands, enter:

  yii help <sub-command>
```

Наборы данных расположены в каталоге **seeds/***. В нем расположены как минимум следующие подкаталоги (источники наборов данных):

- **export** – каталог, содержащий экспортируемые наборы данных;
- **install –** каталог, содержащий наборы данных, поставляемые с приложением.

5.1.1 Установка наборов данных

Для установки наборов данных ввести команду:

```
# php yii seed/install <имена_таблиц> <источник>
```

При этом будет выведена информация по установке наборов данных.

Источник по умолчанию – **install.**

5.1.2 Экспорт наборов данных для переноса на другую систему

Для экспорта наборов данных ввести команду:

```
# php yii seed/export <имена_таблиц>
```

Например, для **auth_item** команда выведет следующую информацию:

```
root@astra:/var/www/jetsignal # php yii seed/export auth_item
Export seed `/var/www/jetsignal/seeds/export/AuthItem.php` successfully
created
```

При переносе наборов данных для установки на другую систему необходимо просмотреть все созданные автоматически файлы. При переносе файлов в другой каталог внутри каталога **seeds**, например, **install2**, в каждом созданном файле требуется поменять параметр **namespace**:

```
namespace app\seeds\export;
```

на

```
namespace app\seeds\install2;
```

После этого файлы должны быть перенесены в указанный каталог на целевой машине и должна быть запущена их установка в соответствии с пунктом по установке наборов данных (см. 5.1.1).

5.2 Резервное копирование и восстановление

Резервное копирование должно выполняться в следующих случаях:

1) Обновление приложения.

2) Внесение изменений в конфигурационные файлы приложения.
3) Выполнение автоматических скриптов по загрузке данных в приложение.

5.2.1 Резервное копирование

Для полного резервного копирования базы данных ввести команды:

```
# sudo -u postgres pg_dumpall > /var/backups/postgresql/pg_dumpall.sql 2>&1
# export exp_date=$(date +"%h.%M_%d.%m.%Y")
# /bin/tar -zcp -C / var/backups/postgresql/ >
/var/backups/jetsignal/jetsignal_db_$exp_date.tar.gz
```

Для полного резервного копирования файлов приложения ввести команды:

```
# export exp_date=$(date +"%h.%M_%d.%m.%Y")
# tar –zcvf /var/backups/jetsignal/jetsignal_files_$exp_date.tar.gz –find
/var/www/jetsignal ! -type l
```

5.2.2 Восстановление из резервной копии

Для восстановления:

1) Найти в каталоге **/var/backups/jetsignal** файлы резервной копии файлов приложения и базы данных.
2) Ввести команды:

   ```
   # mv /var/www/jetsignal /var/www/jetsignal-copy
   # tar –zxvf /var/backups/jetsignal/jetsignal_files_*.tar.gz
   /var/www/jetsignal
   ```

   В зависимости от ситуации каталог **/var/www/jetsignal-copy** либо запаковать в отдельную «битую» резервную копию, либо удалить командой:

   ```
   rm –rf /var/www/jetsignal-copy
   ```

3) Для восстановления базы данных выполнить команду:

   ```
   # psql jetsignal < tar –zxv /var/backups/jetsignal/jetsignal_db_*.tar.gz
   ```

6 Сообщения системному программисту

6.1 Типы сообщений и способы их анализа

Основными источниками сообщений для системного программиста являются:

- Работа с приложением через браузер (сокращённо «Браузер»).
- Файловый журнал веб-сервера Apache2 (сокращённо «Журнал Apache2»).
- Файловый журнал виртуального хоста приложения (сокращённо «Журнал хоста»).
- Файловый журнал кластера PostgreSQL (сокращённо «Журнал БД»).
- Файловый журнал ошибок приложения (сокращённо «Журнал приложения»).
- Файловый журнал ошибок модуля интеграции приложения (сокращённо «Журнал ошибок интеграции»).
- Журнал ошибок приложения в базе данных (сокращённо «Журнал действий пользователя»).
- Журнал ошибок модуля интеграции в базе данных (сокращённо «Журнал действий интеграции»).
- Журнал интеграционных сообщений в базе данных (сокращённо «Журнал сверки интеграции»).

6.1.1 Работа с приложением через браузер

При работе с веб-интерфейсом приложения через браузер сообщения об ошибках отображаются в небольшом красном прямоугольнике в правой верхней части интерфейса приложения.

Сообщения об ошибке хранятся в оперативной памяти до последующего обновления страницы, а также исчезают с экрана в течение 20 секунд.

В случае переключения приложения в режим отладки критичные ошибки приложения могут выводиться на весь экран.

6.1.2 Файловый журнал веб-сервера Apache2

В журнал включаются сообщения о критичных низкоуровневых ошибках функционирования веб-сервера, такие как некорректность работы отдельных модулей, внутренних конфигурационных файлов.

Журнал расположен в двух файлах:

- **`/var/log/apache2/error.log`** – журнал ошибок веб-сервера;
- **`/var/log/apache2/access.log`** – журнал запросов к веб-серверу.

Для просмотра:

а) журнала ошибок ввести команду:

```
# tail -f /var/log/apache2/error.log
```

Будет выведен поток сообщений об ошибках, которые записал веб-сервер.

б) журнала запросов ввести команду:

```
# tail -f /var/log/apache2/access.log
```

Будет выведен поток сообщений обо всех поступающих на виртуальный хост запросах.

Для выхода из режима просмотра журнала следует нажать клавишу **[q].**

6.1.3 Файловый журнал виртуального хоста приложения

В журнал включаются сообщения о критичных низкоуровневых ошибках работы приложения и информация о запросах к виртуальному хосту.

Журнал расположен в двух файлах:

– **/var/log/apache2/jetsignal.error.log** – журнал ошибок виртуального хоста;

– **/var/log/apache2/jetsignal.access.log** – журнал запросов к виртуальному хосту.

Для просмотра:

а) журнала ошибок ввести команду:

```
# tail -f /var/log/apache2/jetsignal.error.log
```

Будет выведен поток сообщений об ошибках, записываемых в журнал веб-сервером.

б) журнала запросов ввести команду:

```
# tail -f /var/log/apache2/jetsignal.access.log
```

Будет выведен поток сообщений обо всех поступающих на виртуальный хост запросах.

Для выхода из режима просмотра журнала следует нажать клавишу **[q].**

6.1.4 Файловый журнал кластера PostgreSQL

В журнал включаются сообщения об ошибках в работе кластера базы данных (в случае создания базы данных приложения в кластере, создаваемом по умолчанию при установке PostgreSQL).

Журнал расположен в файле **/var/log/postgresql/postgresql-9.4-main.log.**

Для просмотра нужно ввести команду:

```
# tail -f /var/log/postgresql/postgresql-9.4-main.log
```

Будет выведен поток сообщений, включаемых в журнал, если они создаются в процессе работы кластера.

Для выхода из режима просмотра журнала следует нажать клавишу **[q]**.

6.1.5 Файловый журнал ошибок приложения

В журнал включаются сообщения об ошибках в работе приложения на уровне выше веб-сервера.

Журнал расположен в файле **/var/www/jetsignal/runtime/<maclevel>/logs/app.log**.

Для просмотра журнала нужно ввести команду:

```
# tail -f /var/www/jetsignal/runtime/<maclevel>/logs/app.log
```

Если приложение работает в режиме поддержки мандатных меток, то под каждую мандатную метку будет создан отдельный каталог с числовым номером мандатной метки <maclevel>. Если режим работы мандатных меток выключен, то запись будет производиться в каталог /var/www/jetsignal/runtime/logs/app.log

Будет выведен поток сообщений, включаемых в журнал, если они создаются в процессе работы приложения.

Для выхода из режима просмотра журнала следует нажать клавишу **[q]**.

6.1.6 Файловый журнал ошибок модуля интеграции приложения

В журнал включаются сообщения об ошибках в работе интеграционного модуля приложения (если используется режим работы распределённой информационной системы).

Журнал расположен в файле **/var/www/jetsignal/integration/fs/<maclevel>/logs/system.log,** где **<maclevel>** - мандатная метка, журнал которой требуется просмотреть. Местоположение файла можно изменить путём изменения параметров в файле **unions.php** (см.Приложение Б).

Для просмотра журнала нужно ввести команду:

```
# tail -f /var/www/jetsignal/integration/fs/<maclevel>/logs/system.log.
```

Будет выведен поток сообщений об ошибках интеграции, записываемых в журнал, если такие сообщения создаются в процессе работы приложения.

Для выхода из режима просмотра журнала следует нажать клавишу **[q]**.

6.1.7 Журнал ошибок приложения в базе данных

В журнал включаются сообщения об ошибках и информационные сообщения, которые записываются в базу данных и доступны из веб-интерфейса.

Журнал доступен в разделе «Настройка» -> «Журнал» веб-интерфейса приложения.

Для просмотра журнала нужно войти под учётной записью администратора системы с ролью admin, у которой должна быть установлена операция **admin_log_view** «Просмотр журнала действий пользователей».

6.1.8 Журнал ошибок модуля интеграции в базе данных

В журнал включаются сообщения об ошибках при выполнении операций отправки и получения интеграционных сообщений.

Журнал доступен в разделе «Настройка» -> «Ошибки интеграции» веб-интерфейса приложения.

Для просмотра журнала нужно войти под учётной записью пользователя с ролью admin, у которой должна быть установлена операция **admin_integration_error-view**.

6.1.9 Журнал интеграционных сообщений в базе данных

Журнал доступен в разделах «Настройка» -> «Входящие сообщения» и «Настройка» -> «Исходящие сообщения» веб-интерфейса приложения.

Для просмотра журнала нужно войти под учётной записью пользователя с ролью admin, у которой должны быть установлены операции **admin_integration_incoming-view** и **admin_integration_outgoing-view**.

6.2 Типовые сообщения об ошибках

№	Источники	Текст ошибки	Возможные причины	Возможные решения
1.	Браузер, Журнал хоста	**PHP Warning – yii\base\ErrorException copy(/var/www/jetsignal/web/assets/52b3d1ff/css/interr.css): failed to open stream: Permission denied**	Нет прав на запись файла, нет прав на каталог. Отсутствует каталог. Нет прав на запись в каталог в ACL Не созданы каталоги под мандатные метки в режиме работы приложения с мандатными метками	Исправить права на файл/ каталог, создать каталог: **# mkdir /var/www/jetsignal/web/assets** **# chmod –R 777 /var/www/jetsignal/web/assets** **# chown –R www-data:www-data /var/www/jetsignal/web/assets** **# setfacl –R –m u:<пользователь>:rwx /var/www/jetsignal/web/assets** **# setfacl –d –m u:<пользователь>:rwx /var/www/jetsignal/web/assets** Выполнить скрипт исправления каталогов под мандатные метки: **# bash /var/www/jetsignal/fixdirs.sh**
2.	Браузер, Журнал хоста	**PHP Warning – move_uploaded_file(/var/www/jetsignal/storage/task/instruction/4503599627370599/etJyRDelX4zVmsqY.pdf): failed to open stream: Permission denied (Тип ошибки не был определен)**	Нет прав на запись файла. Нет прав на каталог файлового хранилища. Отсутствует каталог, нет прав на запись в каталог в ACL. Не хватает прав на запись в каталог у пользователя Не созданы каталоги под мандатные метки в режиме работы приложения с мандатными метками	Исправить права на файл/ каталог, создать каталог: **# mkdir /var/www/jetsignal/storage** **# chmod –R 777 /var/www/jetsignal/storage** **# chown –R www-data:www-data /var/www/jetsignal/storage** Исправить права на каталоги для пользователя, у которого проявляется ошибка: **# setfacl –R –m u:<пользователь>:rwx /var/www/jetsignal/storage** **# setfacl –d –m u:<пользователь>:rwx /var/www/jetsignal/storage** Выполнить скрипт исправления каталогов под мандатные метки: **# bash /var/www/jetsignal/fixdirs.sh**
3.	Браузер, Журнал хоста, Журнал Apache2	**[Sun Mar 19 15:55:10 2017] [error] [client ***.***.***.***] PHP Fatal error: Allowed memory size of 536870912 bytes exhausted (tried to allocate 83 bytes) in**	Нехватка оперативной памяти при выполнении процесса. Некорректная конфигурация php.ini. Некорректная настройка Apache2 MPM (Prefork либо Worker)	Отследить использование оперативной памяти процессами приложения с помощью команд: **# top** **# ps -aux | grep php** **# ps -aux | grep apache2** Проследить параметр **memory_limit** в файлах

№	Источники	Текст ошибки	Возможные причины	Возможные решения
		/var/www/jetsignal/vendor/yiisoft/yii2/db/BaseActiveRecord.php on line 1130, referer: http://*.***.***.***:80/integration/integration-message/index**		**/etc/php5/apache2/php.ini** и **/etc/php5/cli/php.ini:** **# nano /etc/php5/apache2/php.ini** **# nano /etc/php5/cli/php.ini** Проследить параметры **mpm_*** в файле **/etc/apache2/apache2.conf** на соответствие документации веб-сервера Apache2 и документации Astra Linux 1.5 SE. **# nano /etc/apache2/apache2.conf** При необходимости увеличить объём оперативной памяти сервера
4.	Журнал хоста, Журнал Apache2	[**Mon Mar 20 09:41:58 2017] [error] [client ***.***.***.***] 2017-03-20 09:41:58 ERROR [Logger] Logger could not write log error_log(C:/JetSignal/project/integration/fs/logs/system.log): failed to open stream: No such file or directory, referer: http://***.***.***.***/directory/directory/view?key=threat**	Некорректно указан путь файлов журнала интеграции в **unions.php**. Отсутствует каталог, указанный в пути. Некорректно настроены права на запись каталогов Не созданы каталоги под мандатные метки в режиме работы приложения с мандатными метками	Исправить в файле **config/unions.php** параметры узла с **'type' = 'self'**: **# nano config/unions.php** Параметры: **'folder' => '../integration/fs'** **'fileServerBuffer' => '../integration/fs/upload',** Создать каталоги: **# mkdir integration/fs** **# mkdir integration/fs/upload** Поправить права на каталоги: **# chmod –R 777 integration/fs** **# chown –R www-data:www-data integration/fs** Поправить ACL на каталоги для пользователя, у которого проявляется ошибка: **# setfacl –R –m u:<пользователь>:rwx integration** **# setfacl –d –m u:<пользователь>:rwx integration** Выполнить скрипт исправления каталогов под мандатные метки: **# bash /var/www/jetsignal/fixdirs.sh**
5.	Журнал хоста, Журнал Apache2,	**[Mon Mar 20 13:01:00 2017] [error] [client ***.***.***.***] PHP Warning:**	Отсутствует файл **config/unions.php**. Нет прав на чтение файла	Создать файл из прототипа: **# cp config/unions.php.sample**

№	Источники	Текст ошибки	Возможные причины	Возможные решения
	Журнал приложения	**require(/var/www/jetsignal/config/unions.php): failed to open stream: No such file or directory in /var/www/jetsignal/config/params.php on line 6** **[Mon Mar 20 13:01:00 2017] [error] [client 10.31.249.116] PHP Fatal error: require(): Failed opening required '/var/www/jetsignal/config/unions.php' (include_path='.:/usr/share/php:/usr/share/pear') in /var/www/jetsignal/config/params.php on line 6**	**config/unions.php**.	**config/unions.php** Отредактировать файл в соответствии с инструкцией по настройке системы: **# nano config/unions.php** Исправить права на чтение файла: **# chmod 777 config/unions.php** Исправить ACL-права на файл для пользователя, у которого проявляется ошибка: **# setfacl –d –m u:<пользователь>:rx config/unions.php** **# setfacl –d –m u:<пользователь>:rx config/unions.php**
6.	Журнал приложения	**2017-03-20 13:06:31 [***.***.***.***][4503599627370511][-][warning][yii\debug\Module::checkAccess] Access to debugger is denied due to IP address restriction. The requesting IP address is ***.***.***.***	Нет доступа к отладчику (debugger) для указанного пользователя	Отключить режим отладки приложения в соответствии с данным руководством. Проверить корректность прав пользователя, идентификатор в БД которого указан вторым параметром в квадратных скобках: **[4503599627370511]** Проверить корректность сетевых разрешений в ОС пользователя, IP-адрес которого указан первым параметром в квадратных скобках: **[***.***.***.***]**
7.	Журнал ошибок интеграции, Журнал действий интеграции	**2017-03-20 12:14:30 ERROR [Broker] Exception 'ErrorException' with message 'stream_socket_client(): unable to connect to tcp://***.***.***.***:5672**	Нет доступа к серверу очередей. Отсутствует доступ к серверу очередей, локальному или удаленному. Не запущен сервер RabbitMQ на локальном или удаленном сервере. В файле настроек unions.php указаны неверные адреса подключения к серверам очередей	Проверить сетевые доступы к удаленному серверу очередей. Восстановить доступ, если доступ отсутствует. Проверить, запущен ли локальный или удаленный сервер очередей RabbitMQ. Запустить, если не запущен. Указать верные адреса в файле настроек unions.php
8.	Журнал ошибок интеграции, Журнал действий интеграции	**2017-03-15 08:38:45 ERROR [Replicator] Error replicating file: File /var/www/jetsignal-83/integration/upload/IntegrationFile_hwqJxGXl7gUpetpV.jpg20**	Не получен файл-вложение. Файл не был доставлен механизмом доставки файлов	По предыдущей записи лога определить поле RecordId для записи по имени файла. В данном примере "IntegrationFile_hwqJxGXl7gUpetpV". Проверить размер файла, не превышает ли он максимальный размер файла, передаваемого по

№	Источники	Текст ошибки	Возможные причины	Возможные решения
		17-03-13_09-55-24_824 does not exist on the local server		интеграции. Проверить, достаточно ли место в каталогах, в которые сохраняются буферные файлы. Отправить файл повторно из ТО, из которого он был отправлен. ТО-отправитель можно определить по полю SourceSystem интеграционного сообщения
9.	Журнал ошибок интеграции, Журнал действий интеграции	**2017-03-20 12:04:30 WARNING [Replicator] In exception block. We have some message: SQLSTATE[23505]: Unique violation: 7 ERROR: duplicate key value violates unique constraint "user_pkey"**	Повторно получено интеграционное сообщение	Настроенные интеграционные связи приводят к двойному приходу сообщений. Одно сообщение отправлено дважды

7 Перечень принятых сокращений

БД	База данных
АРМ	Автоматизированное рабочее место
«Джет Сигнал», Система	Информационная система управления инцидентами информационной безопасности «Джет Сигнал»
ИБ	Информационная безопасность
ПО	Программное обеспечение
СУБД	Система управления базами данных
ТС	Технические средства
ТО	Территориальное объединение
Узел Системы	Часть разворачиваемой Системы, в состав которой входит аппаратное обеспечение, операционная система, СУБД и база данных, сервер очередей и экземпляр приложения
Экземпляр приложения	Копия приложения, входящая в состав каждого узла Системы

Приложение А

Соответствие мандатных меток уровням допуска

Уровень допуска	**Level (параметр -m)**	**Category (параметр -c)**	**MAC LEVEL (БД)**	**MAC CCR (БД)**	**Команды установки на учётную запись в ОС**	**Команды установки на объект БД: типы объектов CLUSTER, DATABASE, SCHEMA, TABLE**
О (открытый)	0:0	0:0	**{0,0}**	**OFF**	**# usermac -m 0:0 <пользователь>**	**# MAC LEVEL ON <тип> <объект> IS '{0,0}';** **# MAC CCR ON <тип> <объект> IS OFF;**
1Г (конфиденциально)	0:1	0:0	**{1,0}**	**OFF**	**# usermac -m 0:1 <пользователь>**	**# MAC LEVEL ON <тип> <объект> IS '{1,0}';** **# MAC CCR ON <тип> <объект> IS OFF;**
1Б (совершенно секретно)	0:3	0:0	**{3,0}**	**OFF**	**# usermac -m 0:3 <пользователь>**	**# MAC LEVEL ON <тип> <объект> IS '{3,0}';** **# MAC CCR ON <тип> <объект> IS OFF;**
Максимальный уровень (администратор)	0:3	0:0	**{3,0}**	**OFF**	**# usermac -m 0:3 <пользователь>** **# usercaps -m +2,+3,+4,+5 <пользователь>**	**# MAC LEVEL ON <тип> <объект> IS '{3,0}';** **# MAC CCR ON <тип> <объект> IS OFF;**

Приложение Б Настройка конфигурационного файла unions.php

```
<?php

/**
 * Пункты назначения
 */
return [
   [
                'title' => 'Центральный. Уровень 1Г', // Название узла
                'unionId' => '2',
      // Настройки интеграции с собственной системой
      'name' => 'CA_1G',
      'type' => 'self', // параметр, признак собственной системы
      'mqInbound' => 'CA_1G_Inbound', // имя входящей очереди
      // Настройка очередей RabbitMQ с разными мандатными метками
      'mqConnections' => [
         [
            'macLevelLabel' => '0', // Уровень мандатной метки
            'mqHost' => '127.0.0.1', // IP адрес сервера RabbitMQ
            'mqPort' => '5672', // Порт
            'mqUsername' => 'int0', // Логин
            'mqPassword' => 'P@ssw0rd', // Пароль
            'mqVirtualHost' => 'o', // Виртуальные хост
         ],
         [
            'macLevelLabel' => '1', // Уровень мандатной метки
            'mqHost' => '127.0.0.1', // IP адрес сервера RabbitMQ
            'mqPort' => '5672', // Порт
            'mqUsername' => 'int1', // Логин
            'mqPassword' => 'P@ssw0rd', // Пароль
            'mqVirtualHost' => '1g', // Виртуальные хост
         ],
         [
            'macLevelLabel' => '3', // Уровень мандатной метки
            'mqHost' => '127.0.0.1', // IP адрес сервера RabbitMQ
```

```
            'mqPort' => '5672', // Порт
            'mqUsername' => 'int3', // Логин
            'mqPassword' => 'P@ssw0rd', // Пароль
            'mqVirtualHost' => '1b', // Виртуальные хост
        ],
    ],

    'folder' => '/var/www/jetsignal/integration',// каталог интеграции, относительно которого будет расположена папка /logs
            'fileServerBuffer' => '/var/www/jetsignal/integration/upload',// каталог временного хранения файлов-вложений
            'integrationSourceFolder' => '/var/www/internal/upload',// каталог входящих сообещний
            'integrationFolder' => '/var/www/jetsignal/integration',// основной каталог интеграции

    'dbBufferReadRows' => '100',
    'filePollingInterval' => '5',
            'minimumFileAge' => '10', // возраст файла в секуднах необходимый для того, чтобы он попал в обработку
            'minimumAttachmentAge' => '5', // возраст файла-вложения в секуднах необходимый для того, чтобы он попал в обработку. Должен быть не менее в чем 2 раза меньше, чем minimumFileAge
            'fileWaitTimeout' => '10', // Ожидание того, что файл-вложение будет перемещен шлюзом и перенаправлен в буферную папку ожидания

            'relation' => 'self',
            //'instructionAllowed' => 'false', // признак того что в юнион можно отпралвять распоряжения
            'integrationMode' => 'queue', //тип интеграции очереди или файлы
            'chatIntegrationMode' => 'queue', //тип интеграции для сообщений чата очереди или http
            'pollingEmailInterval' => '5',

    'sourceSiem' => 'SIEM',
```

```
            'emailServer' => 'XX.XX.XX.XX', //адрес почтового сервера
                'emailUser'   => 'imap', //имя почтового ящика
            'emailPass'   => 'imap', //пароль
            'emailPort'   => '143', //порт
        ],

        // Настройки интегарции с удаленными системами
            [
                'title' => 'ТО 1. Уровень 1Г', // Название узла
                'unionId' => '4',
            'isBroadcast' => 'true', // признак того, что в систему нужно передавать данные при широковещательной рассылке "для всех"
            'name' => 'TO_1G',
            'type' => 'queue', // параметр, тип ваимодейтсвия очередь
            'folder' => '/var/www/external/upload48',

                'relation' => 'child',
    //          'instructionAllowed' => 'true', // признак того что в юнион можно отпралвять распоряжения

            'mqInbound' => 'TO1_1G_Inbound', // имя входящей очереди
            // Настройка очередей RabbitMQ с разными мандатными метками
            'mqConnections' => [
              [
                'macLevelLabel' => '0', // Уровень мандатной метки
                'mqHost' => 'YY.YY.YY.YY', // IP адрес сервера RabbitMQ
                'mqPort' => '5672', // Порт
                'mqUsername' => 'int0', // Логин
                'mqPassword' => 'P@ssw0rd', // Пароль
                'mqVirtualHost' => 'o', // Виртуальные хост
              ],
              [
                'macLevelLabel' => '1', // Уровень мандатной метки
                'mqHost' => 'YY.YY.YY.YY', // IP адрес сервера RabbitMQ
                'mqPort' => '5672', // Порт
```

```
            'mqUsername' => 'int1', // Логин
            'mqPassword' => 'P@ssw0rd', // Пароль
            'mqVirtualHost' => '1g', // Виртуальные хост
        ],
        [
            'macLevelLabel' => '3', // Уровень мандатной метки
            'mqHost' => 'YY.YY.YY.YY', // IP адрес сервера RabbitMQ
            'mqPort' => '5672', // Порт
            'mqUsername' => 'int3', // Логин
            'mqPassword' => 'P@ssw0rd', // Пароль
            'mqVirtualHost' => '1b', // Виртуальные хост
        ],
            ],

    'fileServerAddress' => 'YY.YY.YY.YY', // IP адрес удаленной машины, куда необходиом передавать файлы
    'fileServerBuffer' => 'integration/upload',
    // Настройка передачи файлов с разными мандатными метками
    'fileServerUsers' => [
        [
          'macLevelLabel' => '0', // Уровень мандатной метки
          'username' => 'int0', // Логин
          'password' => 'P@ssw0rd', // Пароль
        ],
        [
          'macLevelLabel' => '1', // Уровень мандатной метки
          'username' => 'int1', // Логин
          'password' => 'P@ssw0rd', // Пароль
        ],
        [
            'macLevelLabel' => '3', // Уровень мандатной метки
            'username' => 'int3', // Логин
            'password' => 'P@ssw0rd', // Пароль
        ],
    ],
```

```
        ],

            // Настройка интеграции с удаленным узлом. Однонаправленный шлюз.
            [
                'title' => 'Центральный. Уровень 1Б', // Название узла
                'unionId' => '2',
            'isBroadcast' => 'true', // признак того, что в систему нужно передавать данные при широковещательной рассылке "для всех"
            'name' => 'CA_1B',
            'type' => 'file', // параметр, тип ваимодейтсвия Файл
            'folder' => '/var/www/external/upload', // каталог, в который помещаются файлы для передачи файлов
                'relation' => 'parent',
            //      'instructionAllowed' => 'false', // признак того что в юнион можно отправлять распоряджения
            ],
        ];
```

Приложение В Ручное создание учётной записи пользователя

Ручной механизм создания пользователя может использоваться в тех случаях, когда автоматизированное создание учётной записи пользователя невозможно.

1) Выполнить команду создания учётной записи пользователя admin:

```
# adduser admin
```

2) Ввести требуемый пароль и нажать **Enter**.

3) В ответ на запрос Системы ввести пароль ещё раз и нажать **Enter**.

Система запросит дополнительную информацию, ввод которой можно пропустить, нажав нужное количество раз клавишу **Enter**.

Пример вывода в случае корректного выполнения данной команды:

```
root@astra:~# adduser admin
Adding user `admin' ...
Adding new group `admin (1017) ...
Adding new user `admin' (1016) with group `admin ...
Creating home directory `/home/admin' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for admin
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
Adding new user `admin' to extra groups ...
Adding user `admin' to group `fuse' ...
Adding user `admin' to group `dialout' ...
Adding user `admin' to group `cdrom' ...
Adding user `admin' to group `floppy' ...
Adding user `admin' to group `audio' ...
Adding user `admin' to group `video' ...
Adding user `admin' to group `plugdev' ...
Adding user `admin' to group `users' ...
```

4) Ввести команду присвоения мандатного уровня доступа создаваемой учётной записи:

в) Если пользователь должен иметь доступ только к несекретным сведениям (открытый контур безопасности):

– ввести команду:

```
# usermac -m 0:0 admin
```

– нажать клавишу **Enter.**

В результате пользователю будет установлен минимальный уровень мандатного доступа.

г) Если пользователь должен иметь доступ к конфиденциальным сведениям (конфиденциальный контур безопасности, 1Г):

– ввести команду:

```
# usermac -m 0:1 admin
```

– нажать клавишу **Enter.**

д) Если пользователь должен иметь доступ к совершенно секретным сведениям (совершенно секретный контур безопасности, 1Б):

– ввести команду:

```
# usermac -m 0:3 admin
```

– нажать клавишу **Enter.**

е) Если пользователь должен иметь служебные привилегии на доступ к любым данным узла Системы, включая обрабатываемые в рамках узла данные с максимальными степенями секретности (такой вариант рекомендуется для учётной записи системного администратора, обеспечивающего работоспособность узла системы), то последовательно ввести команды установки максимальной мандатной метки и сброса для пользователя строгих правил проверки:

```
# usermac -m 0:3 admin
# usercaps -m +2,+3,+4,+5 admin
```

5) Перевести приложение в режим dev и включить YII_DEBUG = true (см. 4.2.2)
6) В веб-интерфейсе приложения нажать кнопку «Добавить пользователя»
7) Заполнить логин, пароль, подразделение, роль и другие обязательные параметры, нажать кнопку «Добавить пользователя»
8) Перевести приложение в режим prod и выключить YII_DEBUG.

После выполнения действий учётная запись пользователя будет создана.

Аналогичной схемы настройки учётных записей следует придерживаться при создании других учётных записей пользователей узла Системы.