



Общие сведения о руководстве по администрированию

Содержание

Условные обозначения	3
Оформление информационных блоков.....	3
Оформление блоков с кодом	3
Оформление таблиц конфигурации	4
Архитектура решения.....	6
Общие сведения.....	6
Архитектурная схема	6
Модули TIONIX	7
Модули OpenStack.....	7

Условные обозначения

Для упрощения восприятия информации и лучшего вида Руководство содержит несколько правил по оформлению материала.

Оформление информационных блоков



Примечание

Комментарий с подробной информацией, объясняющей часть текста.



Важно

Информация, с которой необходимо ознакомиться, прежде чем продолжить работу с документом.



Внимание

Полезная информация, которая не дает пользователю ошибиться.



Информация

Дополнительная информация.

Оформление блоков с кодом

Блок кода с подсветкой синтаксиса используются для выделения типов данных, которые необходимо передать в исходном виде. К таким типам данных относятся: файлы конфигурации, файлы журналирования и исполняемые команды. Пример:

```
systemctl enable openstack-ceilometer-notification.service
```

Блоки с большим содержанием информации оформляются в свернутом виде:

```

openstack endpoint create --region RegionOne \
  metric public http://controller:8041
+-----+
| Field      | Value                                     |
+-----+
| enabled    | True                                     |
| id         | b808b67b848d443e9eaaa5e5d796970c      |
| interface  | public                                   |
| region     | RegionOne                               |
| region_id  | RegionOne                               |
| service_id | 205978b411674e5a9990428f81d69384      |
| service_name | gnocchi                                  |
| service_type | metric                                   |
| url        | http://controller:8041                  |
+-----+

$ openstack endpoint create --region RegionOne \
  metric internal http://controller:8041
+-----+
| Field      | Value                                     |
+-----+
| enabled    | True                                     |
| id         | c7009b1c2ee54b71b771fa3d0ae4f948      |
| interface  | internal                                 |
| region     | RegionOne                               |
| region_id  | RegionOne                               |
| service_id | 205978b411674e5a9990428f81d69384      |
| service_name | gnocchi                                  |
| service_type | metric                                   |
| url        | http://controller:8041                  |
+-----+

$ openstack endpoint create --region RegionOne \
  metric admin http://controller:8041
+-----+
| Field      | Value                                     |
+-----+
| enabled    | True                                     |
| id         | b2c00566d0604551b5fe1540c699db3d      |
| interface  | admin                                    |
| region     | RegionOne                               |
| region_id  | RegionOne                               |
| service_id | 205978b411674e5a9990428f81d69384      |
| service_name | gnocchi                                  |
| service_type | metric                                   |
| url        | http://controller:8041                  |
+-----+


```

Оформление таблиц конфигурации

К каждой конфигурации сервиса прилагается описание настроек в виде таблицы. Она имеет следующий вид:

Параметр	Описание	Примечания
Название обязательного параметра	Описание этого параметра. • По умолчанию: X	<ul style="list-style-type: none"> Примечания в виде списка. Обычно указываются нюансы настройки.
Название необязательного параметра.	Описание этого параметра. • По умолчанию: Y	<ul style="list-style-type: none"> Примечания в виде списка. Обычно указываются нюансы настройки.

Параметр	Описание	Примечания
Название обязательного или необязательного параметра при указании другого параметра.	Описание этого параметра. <ul style="list-style-type: none"> По умолчанию: Z Зависит от: название параметра. 	<ul style="list-style-type: none"> Примечания в виде списка. Обычно указываются нюансы настройки.
Название необязательного параметра оптимизации.	Описание этого параметра. <ul style="list-style-type: none"> По умолчанию: A 	<ul style="list-style-type: none"> Для параметров оптимизации здесь нужно указать, что на что именно параметр влияет.

 Допускается включение отдельных методов описания в зависимости от типа конфигурации, они описываются в самой статье с конфигурацией.

Архитектура решения

Общие сведения

В рамках технического решения в составе облачной платформы TIONIX OpenStack используется для реализации сервисной модели облачной услуги «**Инфраструктура как сервис**» (IaaS), которая разворачивается в частном аппаратном окружении. Платформа OpenStack состоит из ряда отдельных компонентов, выполняющие свои функции, необходимы для реализации облачной платформы. Конкретная инсталляция OpenStack в зависимости от задачи может использовать только некоторые или все доступные сервисы. Каждый компонент имеет свой набор REST API, клиента для командной строки и интерфейсы для различных языков программирования. Все сервисы аутентифицируются через единый сервис идентификации OpenStack Keystone. Все сервисы взаимодействуют друг с другом через API-интерфейсы или используют общую системную шину, которая реализуется протоколом AMQP. Состояние служб и метаданные сервисов хранятся в реляционной базе данных.

В общей архитектуре Tionix Cloud Platform принято следующее:

- В качестве системной шины на базе AMQP используется брокер сообщений RabbitMQ;
- В качестве реляционной базы данных используется MariaDB.

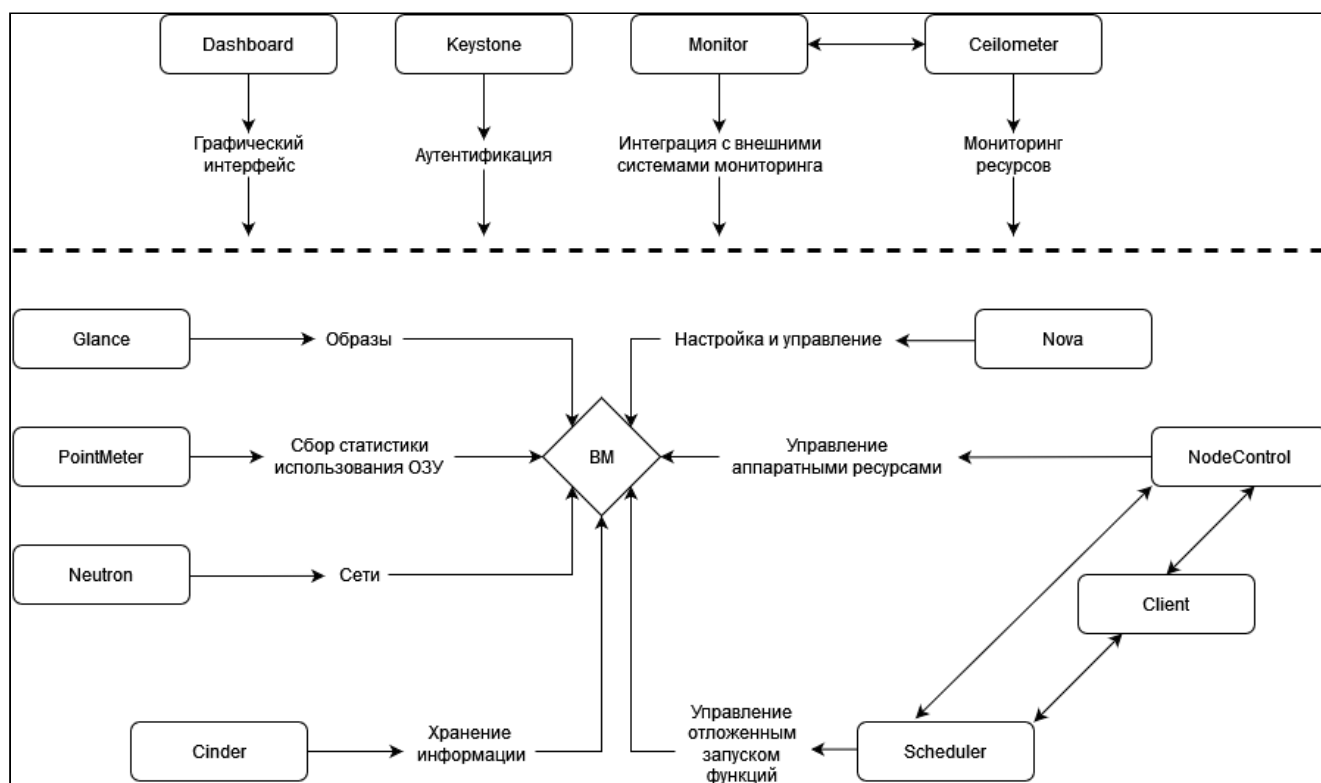
Администратор может получить доступ к OpenStack через веб-интерфейс, реализуемый через сервис Horizon Dashboard, CLI-интерфейс или API-запросы, применяя плагины для браузеров или утилиту curl.

Для обеспечения надежности и масштабируемости управляющей части облачной платформы используются средства кластеризации. В кластеризации применяются следующие технологии:

- **Pacemaker** – основной компонент кластерного окружения, отвечающий за регистрацию и настройку всех необходимых системных объектов, таких как, виртуальные IP-адреса, сервисы облачной платформы Tionix, вспомогательные инфраструктурные сервисы, в виде ресурсов кластера для которых Pacemaker обеспечивает доступность и работоспособность. В случае сбоя узла управления или функционирования сервиса платформы, Pacemaker может перезапустить сервис на работоспособном узле, вывести сбойный узел из кластера, выключить или перезагрузить сервер и т.п.
- **HAProxy** – компонент, отвечающий за балансировку нагрузки управляющих функций между узлами кластера управления.
- **Galera** – компонент, который отвечает за репликацию данных СУБД MariaDB между узлами кластера управления.
- **Кластер RabbitMQ** – для кластеризации шины используются встроенные средства объединения сервисов RabbitMQ в один кластер с репликацией сообщений между собой.

Архитектурная схема

На рисунке ниже приведена схема типовой архитектуры платформы, с указанием потоков информационного взаимодействия между сервисами и их внутренними компонентами:



Типовая архитектура платформы

Модули TIONIX

Модули Tionix – это компоненты для OpenStack, расширяющие ряд функций облачной платформы, которые в основном связаны с управлением жизненного цикла VM, задачами мониторинга и лучшей поддержки внешних аппаратных ресурсов.

TIONIX.NodeControl – модуль, расширяющий функционал платформы OpenStack, связанный с управлением аппаратными ресурсами облачной инфраструктуры – вычислительными узлами.

TIONIX.Client – служебный модуль, необходим для предоставления доступа к функциональности модулей TIONIX.

TIONIX.Dashboard – модуль, расширяющий функции OpenStack Horizon. Модуль дополняет стандартный функционал графическими инструментами использования остальных модулей TIONIX.

TIONIX.Monitor – модуль, расширяющий функции платформы OpenStack, связанные с получением статистических данных об основных характеристиках производительности виртуальных машин. Также модуль предоставляет возможности для интеграции с системами мониторинга, в частности Zabbix, и резервного копирования, в частности Bareos.

TIONIX.Scheduler – модуль, обеспечивающий отложенный запуск функций модулей TIONIX или сервисов платформы OpenStack.

Модули OpenStack

Horizon – это панель управления с графическим интерфейсом, предназначенная для отслеживания всех действий, происходящих внутри OpenStack. Для пользователей, Horizon – это инструмент позволяющий отслеживать все, что происходит внутри платформы.

Nova – это контроллер, управляющий работой виртуальных машин OpenStack. Данный компонент предоставляет услуги в виде платформы, а также дает вам возможность управлять собственными гостевыми ОС. Nova отвечает за модуль управления и подготовку к работе, обеспечивающую корректную визуализацию. Помимо этого, Nova выступает в роли центра управления всей системой и, в частности, гипервизорами.

Cinder – это реализация блочного хранилища, принцип работы которого является схожим с принципом работы жесткого диска. Cinder является важным компонентом в платформе OpenStack, выступающей в роли предоставляемой услуги, ввиду определенных сценариев, требующих постоянного и быстрого доступа к данным. Использование Cinder позволяет беспрепятственно достичь этого. Помимо этого, тот факт, что в случае кратковременного завершения работы данные не будут потеряны позволяет разработчикам хранить необходимую информацию, используя блочное хранилище Cinder.

Neutron – это сетевой компонент OpenStack, который обеспечивает надлежащее межсетевое взаимодействие между различными компонентами.

Keystone – играет роль идентификатора и картографического сервиса для приложений и пользователей. Благодаря данному сервису разработчик может отобразить в виде карты необходимые пользователю сервисы и приложения в облаке, и, кроме того, системные администраторы имеют возможность установить права доступа и управлять ими.

Glance – это инструмент, предназначенный для хранения образов виртуальных машин. Glance позволяет легко обрабатывать образы или настраивать виртуальные машины и шаблоны для постоянного использования.

Ceilometer – отвечает за всю информацию, связанную с метрикой и/или счетами. Эта услуга имеет решающее значение для формирования счетов и обработки ресурсов.

Gnocchi – предназначен для хранения агрегированных измерений в очень больших масштабах и управления ими. Включает в себя многопользовательскую базу данных временных рядов, метрик и ресурсов. В качестве источника измеренных данных может использоваться Ceilometer.

Placement – предназначен для учёта данных о доступных ресурсах облачной платформы. Таким ресурсом может быть вычислительный узел, распределенная система хранения данных или пул доступных IP-адресов. Placement хранит не только информации о доступных ресурсах, но и хранит данные их потребления. Каждый инстанс, созданный в Nova, будет регистрировать и связывать различные типы ресурсов с ним в Placement.