Программное обеспечение «1Stack (1Стек)»

ОПИСАНИЕ ТЕХНИЧЕСКОЙ АРХИТЕКТУРЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

(новая редакция)

2024

Аннотация

В документе приведено описание архитектуры Облачной платформы виртуализации 1Stack (1Стек).

Содержание

Введение

1Stack – это дистрибутив OpenStack, который устанавливается на ОС МСВСфера версии 8.5 и выше, включает в себя дополнительные инструменты администратора. 1Stack можно использовать для развертывания телекоммуникационного облака (Telco Cloud) или частного облака (Private Cloud) и управлять им в собственной среде и на своем оборудовании.

1Stack состоит из набора сервисов, поддерживаемых разработчиком (компанией «Инфосистемы Джет»). Каждый сервис имеет защищенный API, который можно получить через интерфейс командной строки (CLI) или веб-интерфейс. Эти сервисы позволяют создавать инстансы, сети и хранилища, доступ к которым можно ограничить с помощью управления квотами.

OpenStack обладает широким спектром возможностей для построения облачных платформ, а 1Stack – одна из реализаций, где сделан акцент на построении телекоммуникационных платформ.

Для настройки, развертывания и управления продуктивным облаком 1Stack предусмотрен набор инструментов 1Stack TripleO. Архитектура развертывания – гибкая и определяется в файлах формата YAML, поэтому выбранную архитектуру можно сохранять и повторно развертывать одинаково в нескольких ЦОДах.

Платформу 1Stack можно использовать для реализации IaaS («инфраструктуры как услуги») с поддержкой виртуальных экземпляров, сетей и хранилища. При развертывании 1Stack либо по умолчанию, либо с использованием дополнительных доступных функций выполняется развертывание:

- подсистемы управления ресурсами виртуальных экземпляров и образов хранилищ;
- программно-определяемой сети;
- объектного хранилища;
- службы аутентификации и веб-интерфейса для пользователей и администраторов;
- функций безопасности, таких как управление ключами SSH и безопасной аутентификации на основе ролей;
- API-интерфейсы для контроля и управления каждой службой платформы.

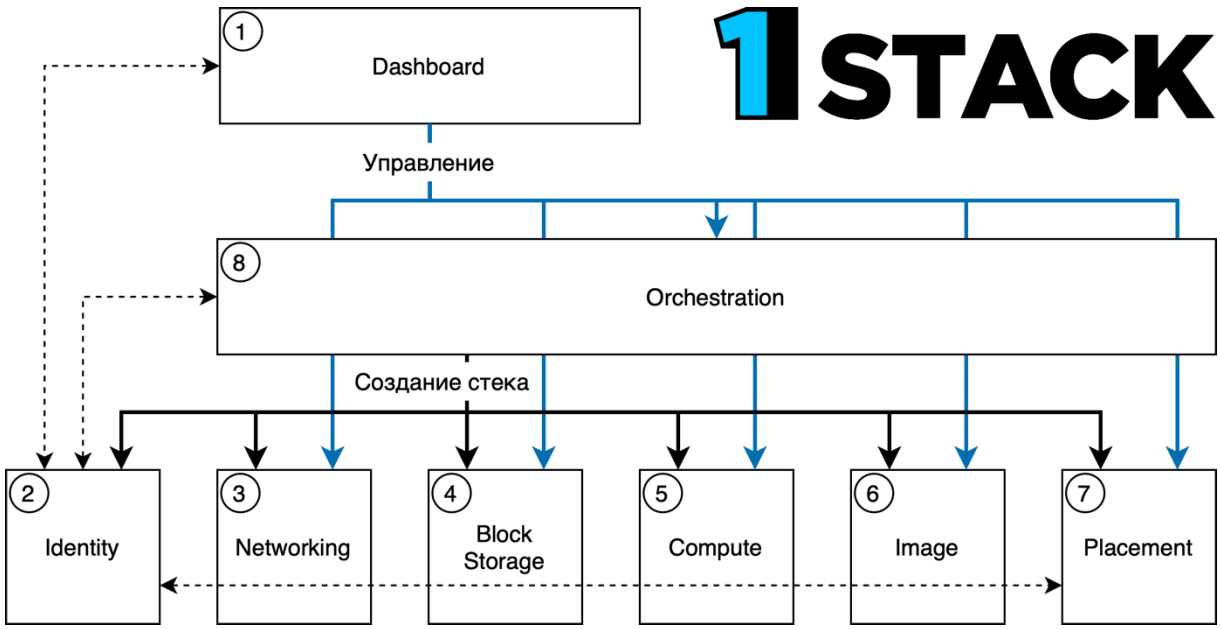
С помощью этих дополнительных функций можно:

- создавать телекоммуникационные или частные облака, которые доступны для масштабирования как в большую, так и в меньшую сторону;
- предоставлять своей организации мультитенантную среду самообслуживания;
- удовлетворять запросы клиентов быстрее, чем в традиционных центрах обработки данных.

1 Общая архитектура

Облачное решение 1Stack реализовано в виде набора взаимодействующих служб, которые управляют вычислительными ресурсами, хранилищем и сетевыми ресурсами. Облаком можно управлять с помощью веб-интерфейса или клиента командной строки, которые позволяют администраторам создавать и управлять ресурсами 1Stack. Кроме того, 1Stack имеет REST-API, который также доступен всем пользователям облака.

На диаграмме представлен общий обзор основных сервисов 1Stack и их взаимосвязи:



В таблице описан каждый компонент, показанный на схеме, и приведены ссылки на раздел документации с описанием компонентов:

№	Название сервиса	Кодовое название	Описание
1	Dashboard	horizon	Панель управления на основе веб-интерфейса. Используется для управления службами 1Stack
2	Identity	keystone	Централизованный сервис для аутентификации и авторизации служб 1Stack, а также для управления пользователями, проектами и ролями
3	Networking	neutron	Обеспечивает связность между интерфейсами служб 1Stack и пользовательскими сетями
4	Block Storage	cinder	Управляет постоянными блочными томами хранилища для инстансов
5	Compute	nova	Управляет виртуальными машинами, которые работают на узлах гипервизора, и обеспечивает их создание
6	Image	glance	Служба реестра. Используется для хранения таких ресурсов как образы виртуальных машин и моментальных снимков томов
7	Placement	placement	Обеспечивает управление ресурсами платформы
8	Orchestration	heat	Механизм оркестровки на основе шаблонов. Поддерживает автоматическое создание стеков ресурсов

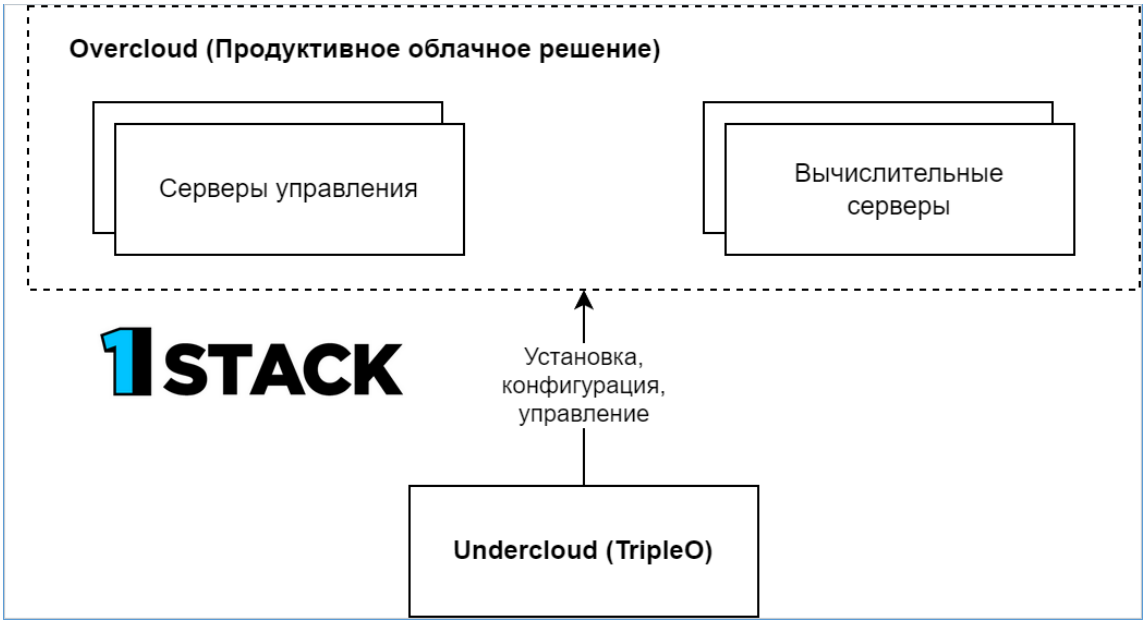
Каждая служба 1Stack содержит функциональную группу служб Linux и других компонентов. Например, службы glance-api и glance-registry Linux вместе с базой данных MariaDB реализуют службу реестра.

2 Подсистемы обслуживания

2.1 TripleO

TripleO – это набор инструментов для установки полноценной облачной платформы 1Stack и её обслуживания. С помощью **TripleO** можно устанавливать ОС на физические серверы и управлять ими, настраивать сервисы управления и в целом – установить полностью работоспособное, отказоустойчивое и надежное облачное решение.

TripleO использует две основные концепции: Undercloud и Overcloud. Сначала устанавливается Undercloud, который затем используется в качестве инструмента для установки и настройки Overcloud.



2.1.1 Undercloud

Undercloud – главный сервер управления, содержащий набор инструментов TripleO. Это односерверная установка 1Stack, включающая компоненты для подготовки и управления продуктивными серверами 1Stack, которые составляют среду 1Stack (overcloud). Формирующие Undercloud компоненты выполняют несколько функций:

1. **Планирование**

В состав Undercloud входят функции планирования, которые можно использовать для создания и назначения определенных ролей вычислительных серверов. Также имеется набор ролей серверов по умолчанию, которые можно назначать конкретным серверам: вычислительным и серверам для управления. Также можно создавать собственные роли. Кроме того, можно выбрать, какие сервисы платформы 1Stack нужно включить в каждую роль сервера. Это позволяет создавать новые типы серверов.

2. **Управление физическими серверами**

Undercloud использует интерфейс управления out-of-band, обычно Intelligent Platform Management Interface (IPMI) или Redfish каждого сервера для управления питанием, и службу на основе PXE для определения характеристик оборудования и установки 1Stack.

3. **Оркестрация**

Undercloud содержит набор шаблонов YAML – это набор планов для облачной платформы. Undercloud загружает эти планы и выполняет их инструкции для создания продуктивного решения 1Stack. Параметры планов можно переопределить, если нужно включить собственные настройки.

4. **Компоненты Undercloud**

Undercloud использует компоненты 1Stack в качестве базового набора инструментов. Каждый компонент работает в отдельном контейнере в Undercloud:

- Identity (keystone) – обеспечивает аутентификацию и авторизацию компонентов TripleO;
- Bare Metal (ironic) и Compute (nova) – управляет физическими серверами;
- Networking (neutron) и Open vSwitch – управляет сетями физических серверов;
- Image (glance) – хранит образы, которые TripleO записывает на физические серверы;
- Orchestration (heat) и Puppet – обеспечивают оркестровку узлов и настройку узлов после того, как TripleO записывает образ Overcloud на диск;
- Workflow (mistral) – предоставляет набор рабочих процессов для определённых действий, специфичных для TripleO, таких как импорт и выполнение планов;
- Messaging (zaqar) – предоставляет службу обмена сообщениями для Workflow Service;
- Object Storage (swift) – предоставляет объектное хранилище для различных компонентов платформы 1Stack, включая:
  - хранилище образов для Image;
  - данные интроспекции для Bare Metal;
  - планы развертывания Workflow.

2.1.2 Overcloud

Overcloud – продуктивное облачное решение 1Stack, которое создает Undercloud.

Overcloud состоит из нескольких серверов с различными ролями, которые определяются на основе конфигурационных файлов TripleO.

Undercloud включает в себя набор ролей серверов Overcloud по умолчанию:

1. **Роль управления**

Серверы управления обеспечивают администрирование, сетевое взаимодействие и высокую доступность среды 1Stack. 1Stack содержит три сервера управления, объединенных в кластер высокой доступности.

Сервер управления по умолчанию состоит из следующих компонентов:

- Dashboard (horizon);

- Identity (keystone);
- Compute (nova);
- Networking (neutron);
- Image (glance);
- Block Storage (cinder);
- Orchestration (heat);
- База данных MariaDB;
- Сервер очередей RabbitMQ;
- Кеширующие сервисы;
- Балансировщик HAproxy;
- Open vSwitch;
- Pacemaker и Galera для обеспечения высокой доступности сервисов.

Не все эти службы включены по умолчанию. Для включения некоторых из них требуется настроенные environment-файлы.

2. **Вычислительная роль**

Вычислительные серверы предоставляют вычислительные ресурсы для среды 1Stack. Добавить дополнительные вычислительные серверы для расширения среды можно в любой момент.

Вычислительный сервер по умолчанию состоит из следующих компонентов:

- Compute (nova);
- QEMU-KVM;
- Telemetry (ceilometer) agent;
- Open vSwitch.

2.1.3 Обеспечение высокой доступности серверов управления

TripleO устанавливает кластер серверов управления для предоставления высокодоступных сервисов в среде платформы 1Stack.

Для каждой службы TripleO устанавливает одни и те же компоненты на всех узлах контроллера и управляет узлами контроллера как единой службой. Такой тип конфигурации кластера обеспечивает переключение на резервный сервер в случае сбоя основного, а пользователям 1Stack – непрерывность работы.

TripleO платформы 1Stack использует специальное программное обеспечение для управления компонентами на сервере управления:

- Pacemaker – менеджер ресурсов кластера. Управляет и контролирует доступность компонентов 1Stack на всех серверах кластера;
- HAProxy – обеспечивает балансировку нагрузки и проксирует на сервисы для кластера;
- Galera – реплицирует базу данных 1Stack по всему кластеру;
- Memcached и Redis – обеспечивают кеширование базы данных.

2.1.4 Сервисы в виде контейнеров

Каждая служба платформы 1Stack в Undercloud и Overcloud работает в отдельном контейнере Linux на соответствующем сервере.

Платформа 1Stack 1.0 поддерживает установку на операционные системы МСВСфера версии 8.5 и выше. Эта ОС обеспечивает управление контейнерами средствами Podman.

Podman

Pod Manager (Podman) – инструмент для управления контейнерами. Он реализует почти все команды Docker CLI, за исключением тех, которые, связаны с Docker Swarm. Podman управляет подами, контейнерами и образами контейнеров. Одно из основных различий между Podman и Docker заключается в том, что Podman может управлять ресурсами без использования демона, работающего в фоновом режиме.

Дополнительная информация об этом инструменте приведена на веб-сайте Podman.

2.2 Сервис Workflow

Сервис Workflow – это сервис для комплексного выполнения задач. Многие вычисления в современных компьютерных системах можно представить в виде процессов, состоящих из множества взаимосвязанных этапов, которые необходимо выполнять в определенном порядке. Эти этапы часто представляют собой взаимодействие с компонентами, распределенными по разным вычислительным серверам. Сервис Workflow предоставляет возможность для автоматизации таких процессов. Выполняется только на Undercloud (TripleO) и используется только для установки Overcloud.

2.3 Сервис Messaging

Сервис Messaging – программное обеспечение для обмена сообщениями. Позволяет обмениваться данными между распределенными компонентами приложения, выполняющими различные задачи, без потери сообщений и без необходимости постоянного доступа к каждому компоненту.

Выполняется только на Undercloud (TripleO) и используется только для установки Overcloud.

3 Подсистема идентификации

3.1 Сервис Identity

Identity обеспечивает аутентификацию и авторизацию пользователей для всех компонентов 1Stack.

Identity поддерживает множество механизмов аутентификации, включая использование локальных имени пользователя и пароля, системы на основе токенов и службы каталогов LDAP.

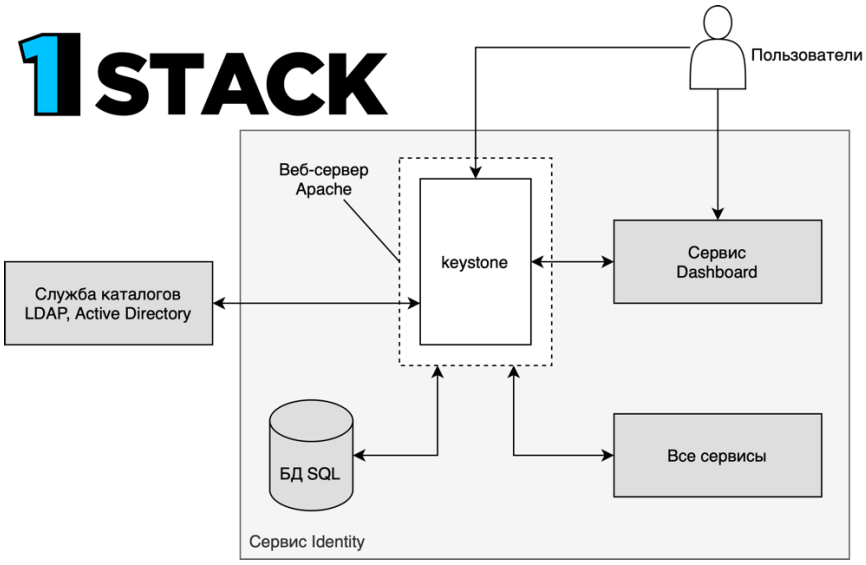
По умолчанию служба идентификации использует базу данных MariaDB для хранения токенов, политик и идентификационной информации. Этот механизм идентификации рекомендуется для сред разработки или для проверки подлинности небольших групп пользователей. Также можно одновременно использовать несколько механизмов идентификации, таких как LDAP и SQL.

Identity поддерживает федерацию с SAML. Федеративная идентификация устанавливает доверие между поставщиками идентификационных данных (IdP) и службами, которые Identity предоставляет конечному пользователю.

Преимущества Identity:

1. Управление учетными записями пользователей (user, group), включая связанную с ними информацию, такую как имя и пароль.
2. Управление проектами (project). В проект можно включить группу пользователей или организацию.
3. Управление ролями (role). Роли определяют права пользователя. Например, роли могут иметь разные права для торгового представителя и для менеджера.
4. Управление доменами (domain). Домены определяют административные границы объектов службы идентификации и поддерживают многопользовательский режим, при котором домен представляет собой объединение пользователей, групп и проектов.

На диаграмме показан базовый процесс аутентификации, который Identity использует для аутентификации пользователей с помощью других компонентов 1Stack:



Компонент	Описание
keystone	Предоставляет идентификацию, а также административный и общедоступный API. Поддерживаются версии Identity API v2 и API v3.

4 Подсистема управления ресурсами

4.1 Сервис Compute

Сервис Compute служит ядром облака 1Stack, предоставляя виртуальные машины по запросу. Сервис выбирает физический сервер и запускает там инстансы, которые взаимодействуют с базовыми механизмами виртуализации, и предоставляет доступ другим компонентам 1Stack.

Compute поддерживает программное обеспечение libvirt, которое использует QEMU-KVM в качестве гипервизора. Гипервизор создает виртуальные машины и обеспечивает миграцию инстанса с узла на узел.

Compute взаимодействует со службами:

- идентификации Keystone для проверки подлинности службы и доступа к базе данных;
- реестра для доступа к образам инстансов и запуска экземпляров, а также взаимодействует со службой панели управления на основе веб-интерфейса для предоставления пользовательского и административного интерфейса.

Администратор может:

- ограничить доступ к образам инстансов по проектам и пользователям;
- указать квоту проекта и пользователя – например, количество экземпляров, которые может создать один пользователь.

При развертывании облако 1Stack можно разбить на различные области:

1. **Регионы (Regions)**

Каждая служба, указанная в службе идентификации Keystone, идентифицируется по региону обслуживания, который обычно представляет собой географическое местоположение.

Регионы используются для построения высокодоступной виртуальной инфраструктуры.

2. **Агрегаты серверов (Host Aggregates) и зоны доступности (Availability Zones)**

Облачное решение можно разделить на логические группы. Можно создать:

- несколько групп узлов, которые совместно используют общие ресурсы, такие как хранилище и сеть,

  или

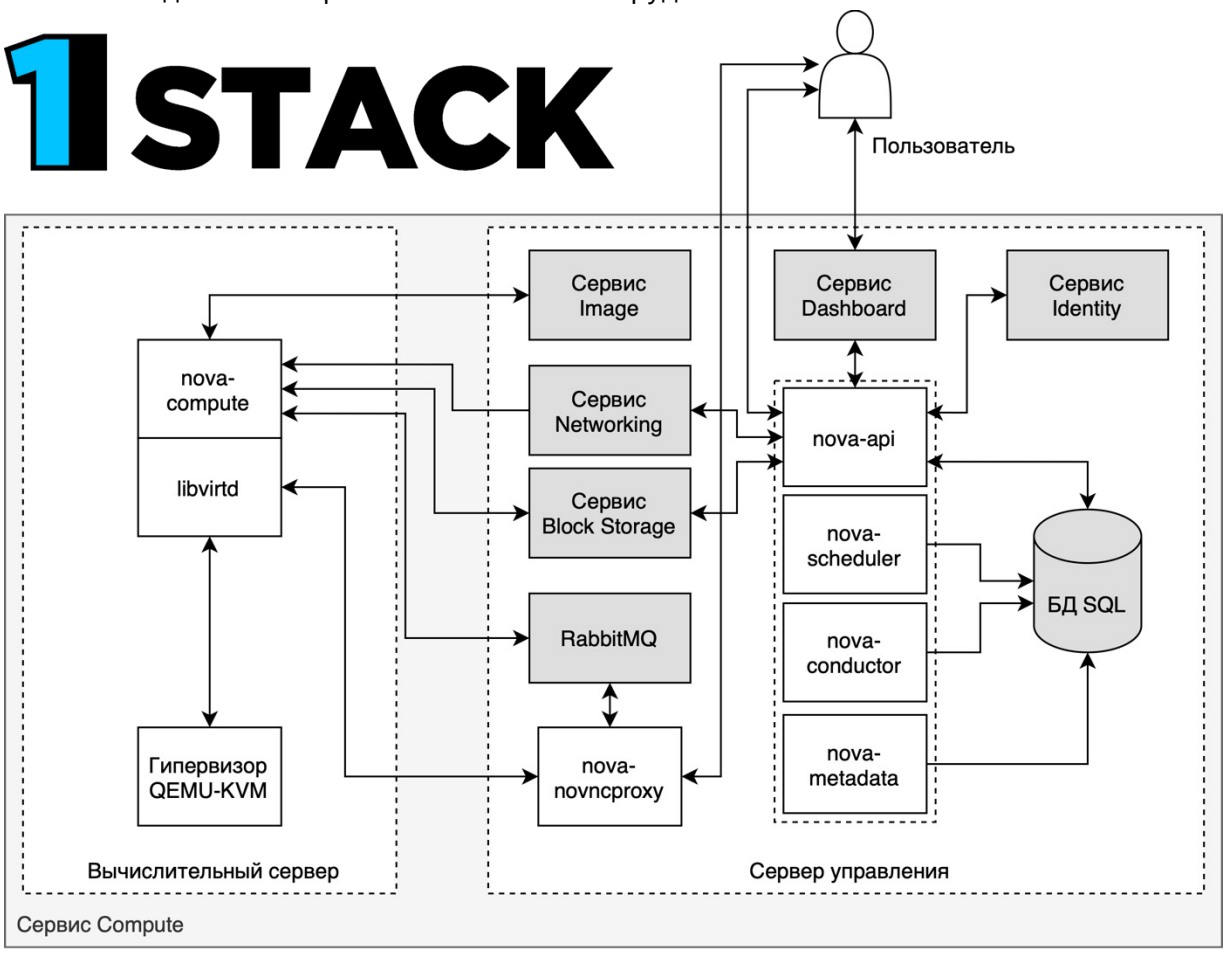
- группы, которые совместно используют определенные свойства, такие как тип CPU.

Для администраторов группа представлена в виде агрегатов серверов с выделенными вычислительными серверами и связанными с ними метаданными. Метаданные агрегатов серверов обычно используются для предоставления информации планировщику ресурсов (openstack-nova-scheduler), таких как ограничение определенных шаблонов серверов или образов инстансов на подмножество серверов.

Для пользователей эта группа представлена как зона доступности. Пользователь не может просматривать метаданные группы или список серверов в зоне доступности.

Преимущества агрегатов и зон доступности:

1. Балансировка нагрузки и распределение инстансов.
2. Физическая изоляция и резервирование между зонами (обычно используются отдельные источники питания или сетевое оборудование).
3. Группировка серверов, имеющих общие атрибуты.
4. Разделение на различные классы оборудования.



Компонент	Описание
nova-api	Обрабатывает запросы и предоставляет доступ к службам, таким как загрузка инстанса
nova-compute	Запускается на каждом вычислительном сервере для создания и завершения работы инстансов. Взаимодействует с гипервизором для запуска новых инстансов и записывает состояния инстансов в базу данных
nova-conductor	Обеспечивает поддержку доступа вычислительных серверов к базе данных
nova-novncproxy	Предоставляет VNC-доступ к инстансу через браузер
nova-scheduler	На основе настроенных весов и фильтров вычисляет, на каком физическом сервере создать или запустить инстанс
nova-metadata	Управляет конфигурацией инстансов

4.2 Сервис Image

Сервис Image выполняет функцию реестра образов виртуальных дисков. Пользователи могут добавлять новые образы или делать мгновенные снимки существующего инстанса. Мгновенные снимки можно использовать для резервного копирования или в качестве шаблонов для новых серверов.

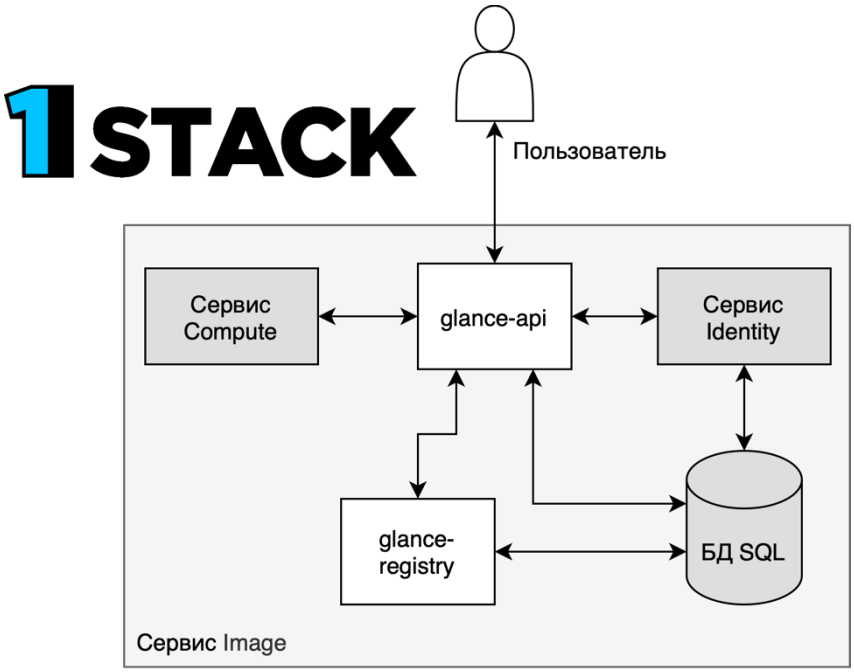
Поддерживаются следующие форматы дисков:

- aki/ami/ari (Amazon kernel, ramdisk или образ инстанса);
- iso (формат архива для оптических дисков, таких как CD);
- qcow2 (Qemu/KVM, поддерживает технология CoW);
- raw (неструктурированный формат).

Форматы контейнеров также можно загрузить в реестр образов. Формат контейнера определяет формат метаданных инстанса, который будет сохранен в образе.

Поддерживаются следующие форматы:

- bare (без метаданных);
- ova (tar-архив OVA);
- ovf (формат OVF);
- aki/ami/ari (ядро Amazon, ramdisk или образ инстанса).

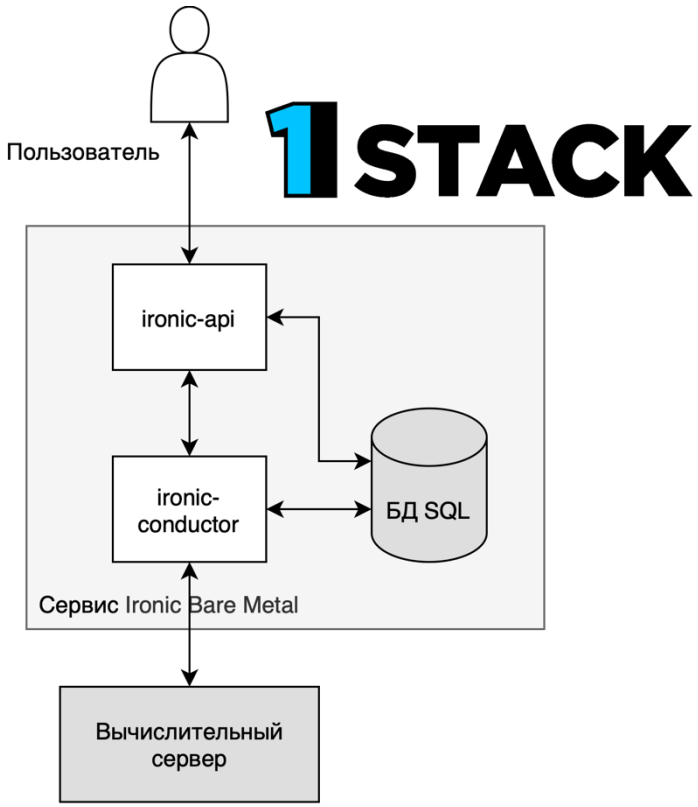


Компонент	Описание
glance-api	Взаимодействует с серверной частью хранилища для обработки запросов на создание, изменение и удаление образов инстансов. API использует glance-registry для получения информации об образах инстансов
glance-registry	Управляет всеми метаданными для каждого образа инстанса

4.3 Сервис Bare Metal

Сервис Bare Metal позволяет пользователю устанавливать ОС и настраивать физические серверы. Сервис запускается только на TripleO.

Для планирования ресурсами и управления квотами Bare Metal использует сервис Compute, а для аутентификации сервиса – службу Identity. Образы инстансов должны быть настроены для поддержки установки на физический сервер вместо KVM.



Компонент	Описание
ironic-api	Обрабатывает запросы и предоставляет доступ к физическим вычислительным ресурсам
ironic-conductor	Взаимодействует с физическими серверами и базой данных, а также выполняет пользовательские и служебные действия

4.4 Сервис Orchestration

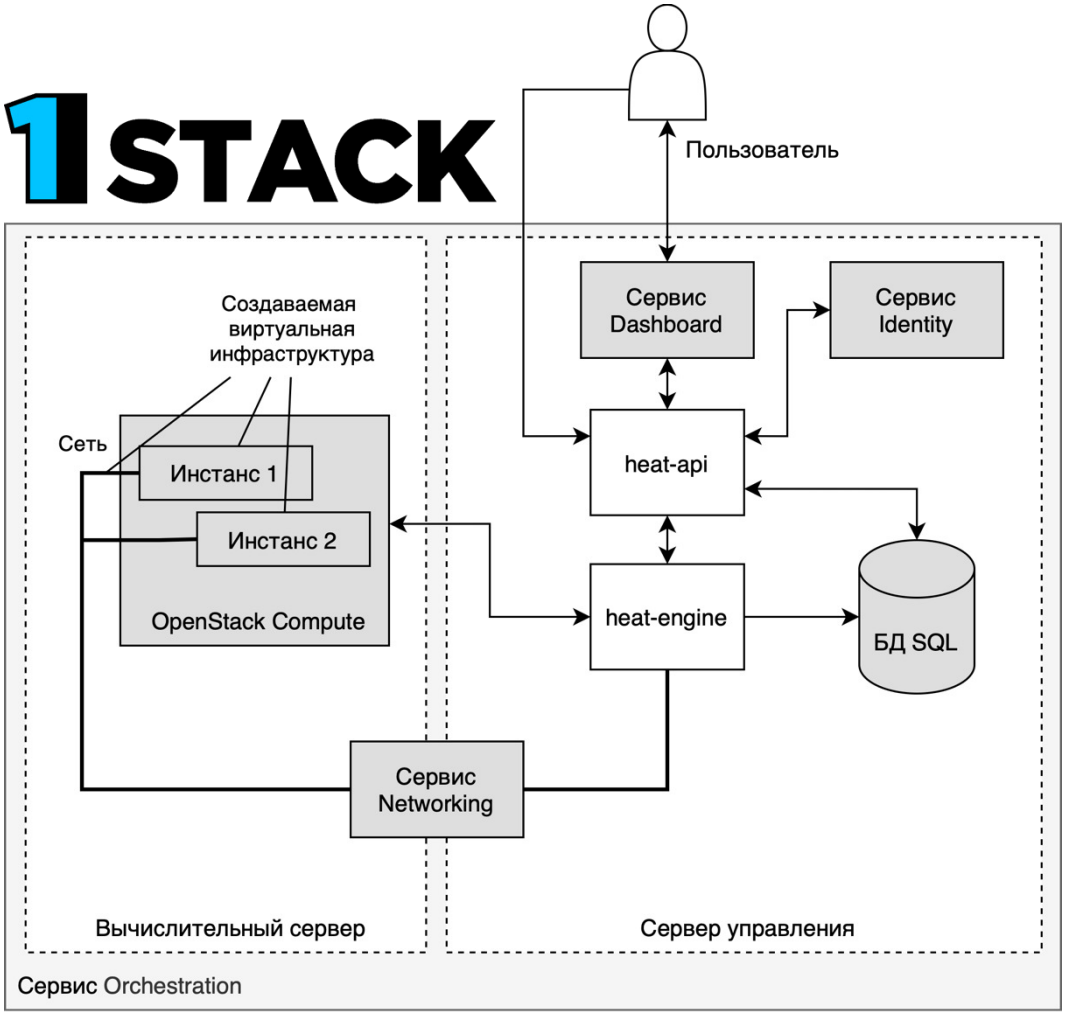
Orchestration предоставляет шаблоны для создания облачных ресурсов, таких как хранилища, сети, инстансы или приложения, и для управления ими.

Шаблоны используются для создания стеков, которые представляют собой коллекции ресурсов.

Например, можно создать шаблоны для экземпляров, плавающих IP-адресов, томов, групп безопасности или пользователей. Orchestration предоставляет доступ ко всем основным сервисам 1Stack с помощью единого модульного шаблона.

Преимущества Orchestration:

1. Наличие единого шаблона, который предоставляет доступ ко всем базовым API-интерфейсам сервисов.
2. Шаблоны являются модульными и ресурсоориентированными.
3. Шаблоны можно переопределить и использовать повторно (например, в виде вложенных стеков).
4. Подключаемая реализация ресурсов, поэтому можно включать пользовательские ресурсы.
5. Используется подход Infrastructure-as-a-Code (IaaC).



Компонент	Описание
heat-api	Собственный REST-API, который обрабатывает запросы API путем отправки запросов в службу heat-engine через RPC
heat-api-cfn	Дополнительный AWS-Query API, совместимый с AWS CloudFormation, который обрабатывает запросы API, отправляя запросы в службу heat-engine через RPC
heat-engine	Управляет запуском шаблона и генерирует события для API пользователя

4.5 Сервис Placement

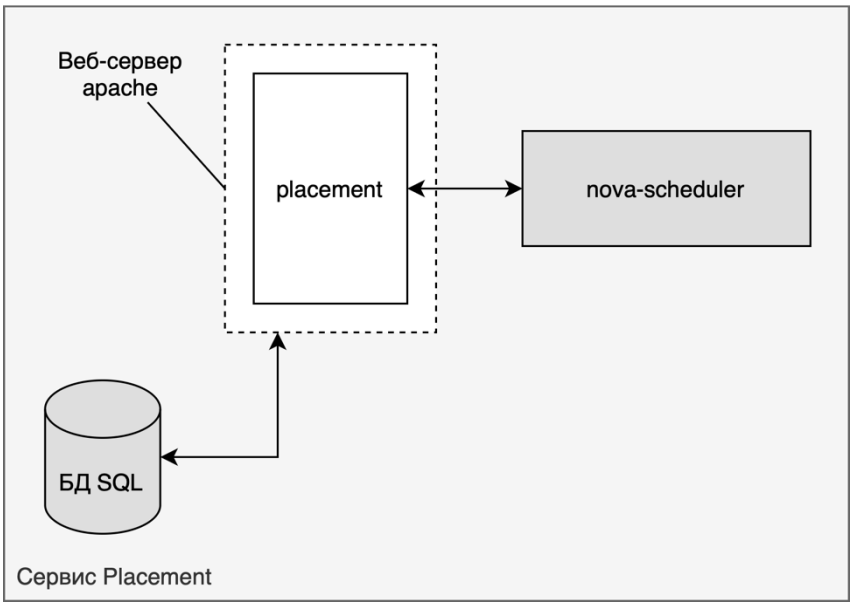
Сервис Placement обеспечивает управление ресурсами платформы. Применяется для ведения учета ресурсов и использования поставщиков ресурсов. Например: поставщиком ресурсов может быть вычислительный сервер, общий пул хранилища или пул распределения IP-адресов.

Типы потребляемых ресурсов отслеживаются в виде классов. Сервис предоставляет:

- набор стандартных классов ресурсов (например, DISK_GB, MEMORY_MB и VCPU);
- возможность определять пользовательские классы ресурсов по мере необходимости.

У каждого поставщика ресурсов также может быть набор характеристик, которые описывают дополнительные возможности или ограничения поставщика ресурсов.

1STACK



Компонент	Описание
placement	Обеспечивает управление ресурсами платформы, взаимодействует с компонентом планировщика ресурсов nova-scheduler сервиса Compute

5 Подсистема хранения

5.1 Сервис Block Storage

Сервис Block Storage управляет постоянным блочным хранилищем для виртуальных жестких дисков. Блочное хранилище позволяет пользователю создавать и удалять блочные устройства, а также управлять подключением блочных устройств к инстансам.

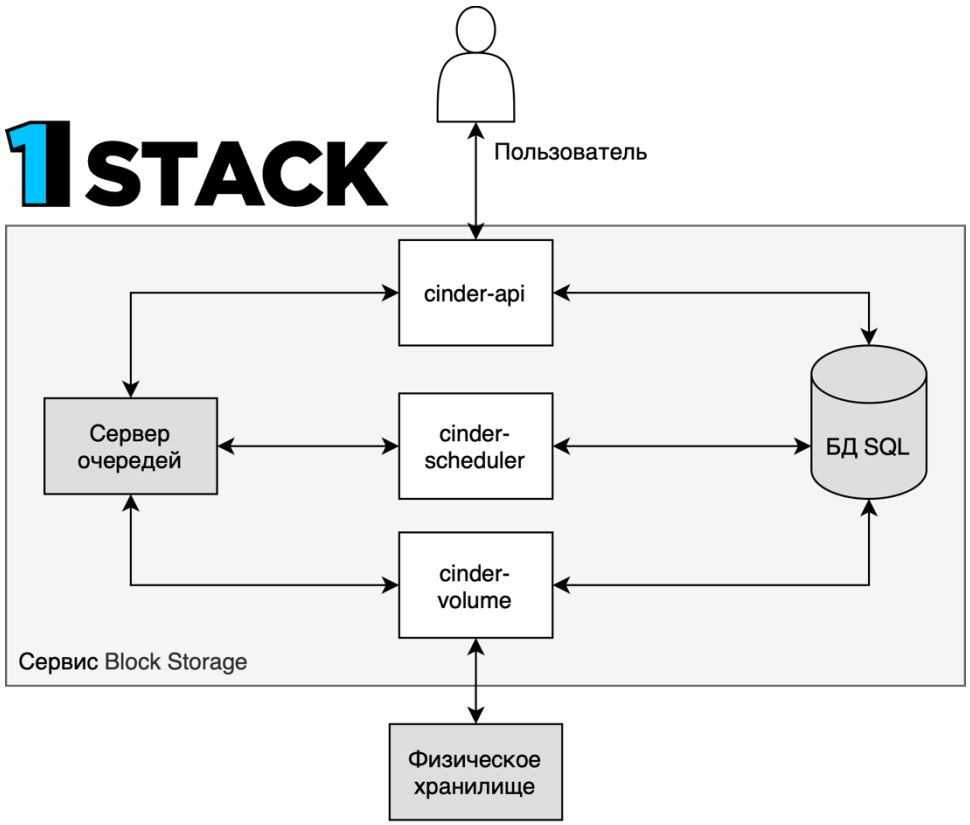
Фактическое подключение и отсоединение устройств выполняется за счёт интеграции с сервисом Compute. Для управления распределенным блочным хранилищем можно использовать регионы и зоны доступности.

В свою очередь, блочное хранилище можно использовать в сценариях, требующих высокой производительности, таких как хранилище баз данных или расширяемые файловые системы.

Также сервис предоставляет инстансам прямой блочный доступ к дискам. Кроме того, для сохранения данных или создания новых томов можно делать моментальные снимки томов.

Преимущества блочного хранилища 1Stack:

1. Создание, просмотр и удаление томов и моментальных снимков.
2. Подключение и отсоединение томов от запущенных инстансов.



Компонент	Описание
cinder-api	Отвечает на запросы пользователей и сервисов и помещает их в очередь сообщений. При получении запроса служба API проверяет соответствие требованиям к идентификации и преобразует запрос в сообщение, содержащее требуемое действие с блочным хранилищем
cinder-scheduler	Назначает задачи из очереди сообщений и планирует выделение ресурсов. Служба планировщика считывает запросы из очереди сообщений и определяет, на каком хранилище следует выполнить запрошенное действие. Затем планировщик связывается со службой cinder-volume на выбранном сервере для обработки запроса
cinder-volume	Определяет хранилище для инстансов. Служба томов взаимодействует непосредственно с устройствами блочного хранения. Когда от планировщика поступают запросы, служба томов может создавать, изменять или удалять тома. Служба томов включает в себя несколько драйверов для взаимодействия с устройствами блочного хранения, такими как NFS

5.2 Сервис Object Storage

Сервис Object Storage предоставляет доступную по протоколу HTTP систему хранения больших объемов данных, включая статические объекты, такие как видео, изображения, сообщения электронной почты, файлы или образы инстансов.

Также в объектном хранилище сохраняются данные интроспекции для Bare Metal и планы развертывания Workflow на этапе установки 1Stack. Объекты хранятся в базовой файловой системе в виде двоичных файлов, а метаданные – в расширенных атрибутах каждого файла.

Распределенная архитектура Object Storage поддерживает горизонтальное масштабирование, а также отказоустойчивое резервирование с помощью программной репликации данных. Поскольку сервис поддерживает асинхронную репликацию и, в конечном итоге, последовательную репликацию, ее можно использовать при развертывании в нескольких центрах обработки данных.

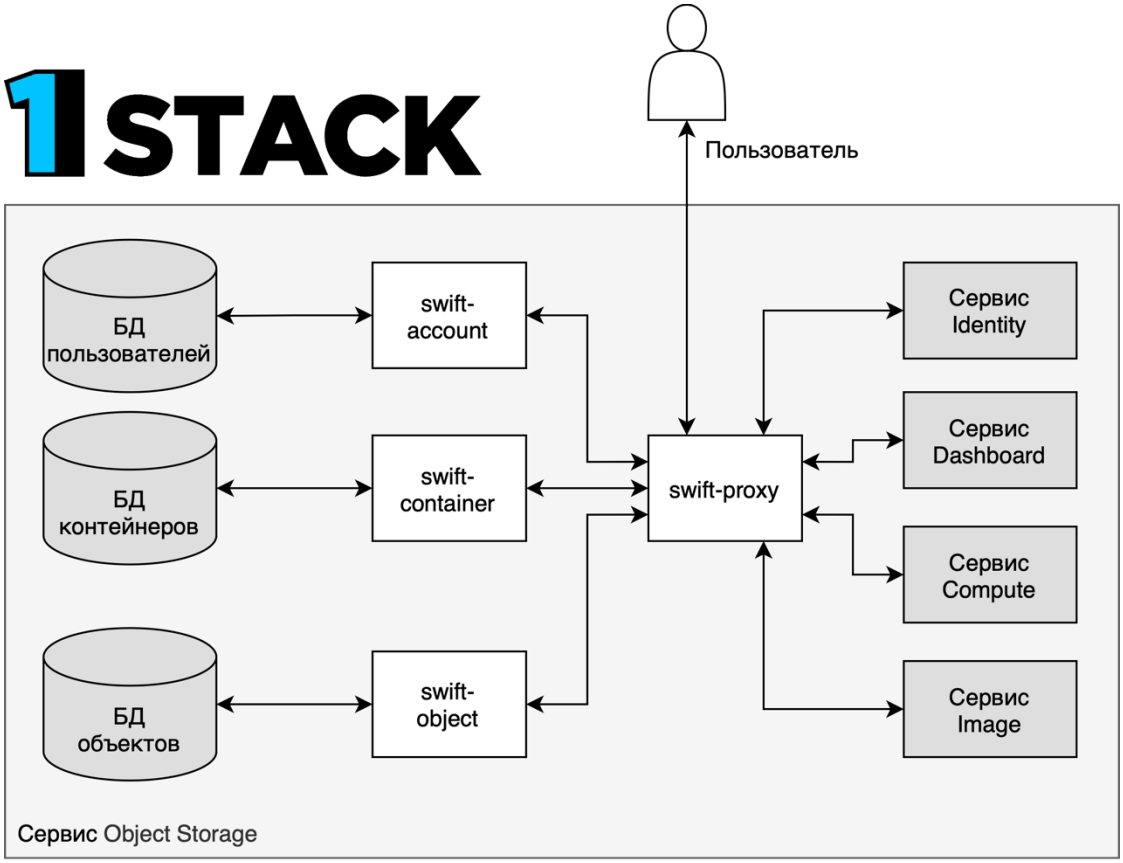
Преимущества хранилища объектов 1Stack:

1. Реплики хранилища поддерживают состояние объектов в случае сбоя. Рекомендуется использовать как минимум три реплики.
2. Реплика зон хранения. Зоны гарантируют, что каждая реплика объекта может храниться отдельно. Зона может представлять собой отдельный диск, массив данных, сервер, серверную стойку или даже целый центр обработки данных.
3. Регионы хранения могут: группировать зоны по местоположению; включать серверы или серверные фермы, которые обычно расположены в одной и той же географической области. Регионы имеют отдельное API для каждой установки службы Object Storage, что позволяет обеспечить разделение служб.

Объектное хранилище использует кольцевые файлы .gz, которые служат в качестве файлов базы данных и конфигурации. Они содержат подробную информацию обо всех устройствах хранения и сопоставлениях сохраненных объектов с физическим местоположением каждого файла. Таким образом, эти файлы можно использовать для определения местоположения конкретных данных. У каждого объекта, учетной записи и контейнерного сервера есть уникальный кольцевой файл.

Сервис хранения объектов использует другие подсистемы и компоненты 1Stack для выполнения действий. Например, требуется служба Identity, демон rsync и балансировщик нагрузки.

На диаграмме показаны основные интерфейсы, которые сервис Object Storage использует для взаимодействия с другими службами 1Stack, базами данных и брокерами:



Компонент	Описание
swift-account	Обрабатывает списки контейнеров в базе данных
swift-container	Обрабатывает списки объектов, которые включены в определенный контейнер с помощью базы данных
swift-object	Сохраняет, извлекает и удаляет объекты
swift-proxy	Предоставляет общедоступный API, обеспечивает аутентификацию и маршрутизацию запросов. Объекты передаются пользователю через прокси-сервер без буферизации

6 Сетевая подсистема

6.1 Сервис Networking

Сервис Networking обеспечивает создание и управление виртуальной сетевой инфраструктуры в облаке 1Stack. Элементы инфраструктуры включают в себя сети, подсети и маршрутизаторы.

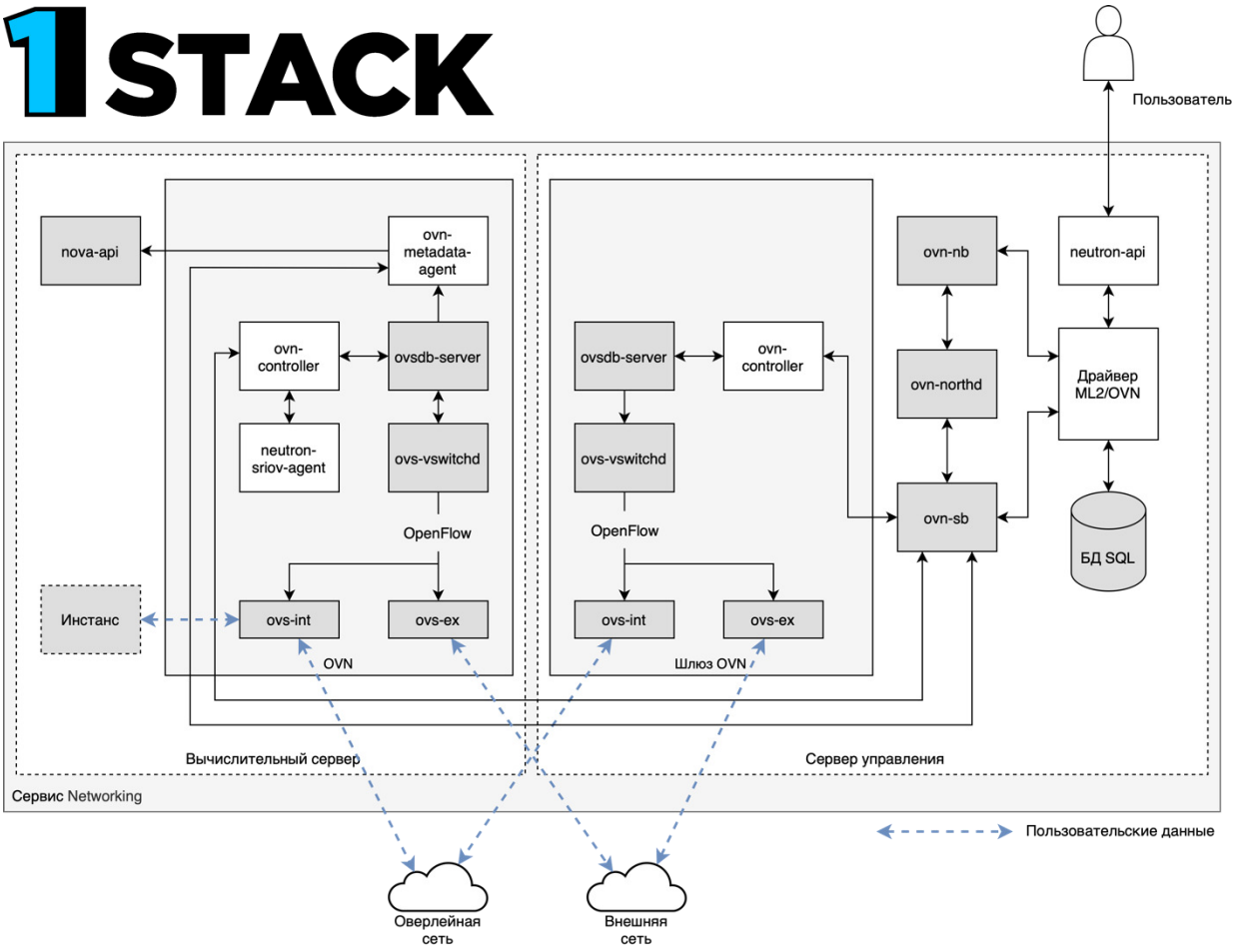
Networking предоставляет администраторам возможность гибко выбирать, какие сетевые службы запускать и на каких физических системах они будут действовать. Сетевые сервисы можно запускать либо на одном физическом сервере, либо на разных серверах, в том числе в отказоустойчивой конфигурации.

Поскольку сеть 1Stack создается на программном уровне, то она может в режиме реального времени реагировать на меняющиеся сетевые потребности, такие как создание и назначение новых IP-адресов.

Преимущества сети 1Stack:

1. Пользователи могут создавать сети, управлять трафиком и подключать серверы и устройства к одной или нескольким сетям.
2. IP-адреса могут быть выделенными или плавающими. Плавающие IP-адреса могут использоваться для динамического перенаправления трафика.
3. Для сети VLAN можно использовать 4094 VLAN, т. е. максимум 4094 сети. Для сетей на основе туннелей VXLAN или GENEVE VNI (идентификатор виртуальной сети) можно использовать около 16 миллионов уникальных адресов/сетей.

Ядро сетевой службы – подключаемый модуль Modular Layer 2 (ML2), а драйвером служит Open Virtual Networking (OVN). Для платформы 1Stack драйвер OVN предоставляет сетевые сервисы.



Компонент	Описание
neutron-api	Центральный управляющий компонент
Драйвер ML2/OVN	Модуль ML2 преобразует сетевую конфигурацию, специфичную для 1Stack, в конфигурацию логической сети OVN
ovn-nb	База данных OVN northbound. В ней хранится логическая конфигурация сети OVN из подключаемого модуля OVN ML2. Запускается на сервере управления на TCP-порту 6641
ovn-northd	Служба OVN northbound. Эта служба преобразует конфигурацию логической сети из базы данных OVN NB в логические правила путей передачи данных и заполняет их в базе данных OVN Southbound
ovn-sb	База данных OVN southbound. В ней хранятся преобразованные логические правила путей передачи данных. Запускается на сервере управления на TCP-порту 6642
ovn-controller	Контроллер OVN. Взаимодействует с базой данных OVN SB и действует как контроллер Open vSwitch для управления и мониторинга сетевым трафиком
ovn-metadata-agent	Агент метаданных OVN. Создает экземпляры haproxy для управления интерфейсами OVS, сетевыми пространствами имен и процессами HAProxy
OVSDB	Сервер базы данных OVS. Размещает базы данных OVN Northbound и Southbound. Также взаимодействует с ovs-vswitchd для размещения базы данных OVS conf.db
neutron-sriov-agent	Агент, отвечающий за работу с SR-IOV портами

ovn-controller, запущенный на серверах управления, выполняет следующие функции:

- маршрутизацию north-south;
- SNAT.

ovn-controller, запущенный на вычислительных серверах, выполняет:

- функции DHCP-сервера;
- функции IGMP;
- функции внутреннего DNS;
- распределение Floating IP;
- работу с трафиком Geneve;
- маршрутизацию east-west.

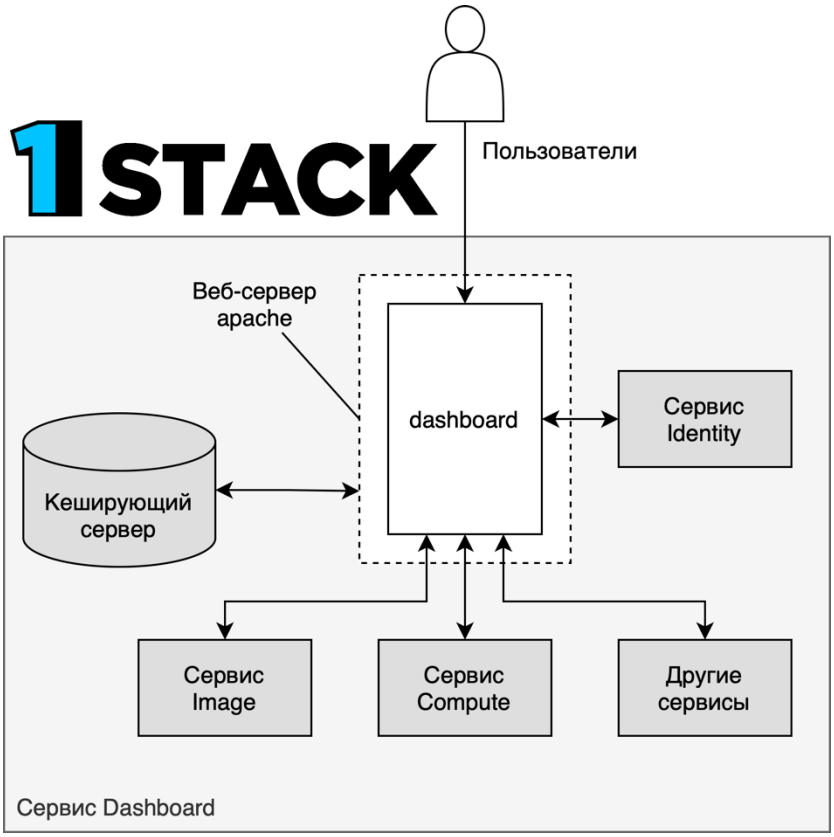
7 Пользовательский интерфейс

7.1 Сервис Dashboard

Сервис Dashboard предоставляет пользователям и администраторам графический пользовательский интерфейс для выполнения операций создания и запуска экземпляров, управления сетью и настройки контроля доступа.

Модульная архитектура сервиса позволяет сервису взаимодействовать с другими подсистемами, такими как биллинг, мониторинг, а также с дополнительными инструментами управления.

Роль пользователя, который входит в веб-интерфейс, определяет, какие информационные панели будут ему доступны.



Компонент	Описание
dashboard	Веб-приложение Django. Предоставляет доступ к веб-интерфейсу управления из любого браузера

В этом примере показано следующее взаимодействие:

- сервис идентификации Identity выполняет аутентификацию и авторизацию пользователей;
- кеширующий сервер Memcached хранит сессии аутентифицированных пользователей;
- сервис Dashboard взаимодействует с другими службами 1Stack по API.

8 Вспомогательные сервисы

8.1 База данных MariaDB

MariaDB – база данных, которая используется по умолчанию, и поставляется с ОС. Для каждого компонента 1Stack требуется чтобы служба MariaDB была запущена. Каждый компонент подсистем хранит свои данные в отдельной БД.

Для построения отказоустойчивой конфигурации используется ПО Galera.

Galera – программное обеспечение репликации данных для создания синхронного кластера MariaDB Galera, состоящего из нескольких серверов MariaDB. В отличие от традиционной настройки основного сервера/реплики, где реплики обычно доступны только для чтения, все узлы в кластере MariaDB Galera могут быть доступны для записи.

Основные функции MariaDB Galera Cluster:

1. Синхронная репликация.
2. Топология active-active с несколькими источниками данных.
3. Чтение и запись на любой сервер кластера
4. Автоматический контроль кластера и удаление отказавших серверов из кластера.
5. Автоматическое присоединение узлов.
6. Параллельная репликация на уровне строк.
7. Прямые клиентские подключения: пользователи могут входить на серверы кластера и работать с узлами напрямую во время выполнения репликации.

Синхронная репликация означает, что сервер реплицирует транзакцию во время фиксации и передает связанный с транзакцией набор данных для записи на каждый сервер кластера. Клиент (пользовательское приложение) подключается напрямую к системе управления базами данных (СУБД) и работает аналогично одной БД MariaDB.

Синхронная репликация гарантирует, что изменение, произошедшее на одном узле кластера, одновременно произойдет и на других узлах кластера.

MariaDB Galera задействована только на серверах управления.

8.2 Сервер очередей RabbitMQ

RabbitMQ – надежная система обмена сообщениями, основанная на стандарте AMQP. RabbitMQ – это высокопроизводительный сервер очередей, который используется во многих корпоративных системах с широкой коммерческой поддержкой.

RabbitMQ управляет транзакциями 1Stack, включая постановку в очередь, распространение сообщений, безопасность, управление и кластеризацию.

Кластер сервера очередей RabbitMQ выполняется только на серверах управления.

8.3 Кеширующие сервисы

Внешние приложения для кеширования, такие как Memcached или Redis, ускоряют работу динамических веб-приложений за счет снижения нагрузки на базу данных.

Кеширование используется различными компонентами **1Stack**, например:

- сервис Object Storage использует memcached для кеширования аутентифицированных клиентов, вместо того чтобы обращаться на каждый запрос клиента к базе данных;
- по умолчанию пользовательский веб-интерфейс использует memcached для хранения сеансов;
- сервис Identity использует memcached для сохранения токенов.

Кеширующие сервисы Memcached и Redis выполняются только на серверах управления.

8.4 Балансировщик HAproxy

HAProxy – программное решение, которое реализует высокодоступный балансировщик нагрузки и прокси-сервер для приложений на основе TCP и HTTP, распределяющих запросы между несколькими серверами.

В **1Stack** HAProxy используется для следующих компонентов:

1. Подсистема хранения, компонент cinder-api.
2. Подсистема управления ресурсами, компонент glance-api.
3. Подсистема управления ресурсами, компонент heat-api и heat-api-cfn.
4. Подсистема управления ресурсами, компонент nova-api.
5. Подсистема управления ресурсами, компонент nova-novncproxy.
6. Подсистема управления ресурсами, компонент nova-metadata.
7. Подсистема управления ресурсами, компонент placement-api.
8. Пользовательский интерфейс, компонент dashboard.
9. Подсистема идентификации, компонент keystone.
10. База данных MariaDB.
11. Сетевая подсистема, компонент neutron-api.
12. Кеширующий сервис Redis.
13. Сервис очередей для мониторинга средствами Service Telemetry Framework (STF), компонент metrics-qdr

HAProxy выполняется только на серверах управления.

8.5 Pacemaker

Pacemaker – программное обеспечение для построения высокой доступности сервисов, в частности, Pacemaker реализует высокую доступность инфраструктуры **1Stack**.

Pacemaker использует обмен сообщениями средствами CoroSync для надежной связи между серверами кластера. CoroSync реализует для Pacemaker обмен сообщениями на основе протокола UDP, оповещает о кворуме и обеспечивает членство в кластере.

Для построения высокой доступности сервисов Pacemaker использует агенты ресурсов, которые представляют собой сценарии, содержащие информацию о том, как запускать,

останавливать и проверять работоспособность каждого приложения, управляемого кластером. Во избежание неконсистентности данных Pacemaker также посредством агента STONITH перезагружает сервер, который перестал быть доступен по сети.

В 1Stack Pacemaker использует следующие ресурсы:

1. **ocf::heartbeat:IPaddr2** – для плавающих IP-адресов внутренних и внешних API, управляющих сервисов и др.
2. **stonith:fence_ipmilan** – для перезагрузки неисправного сервера.
3. **ocf::heartbeat:podman:haproxy-bundle** – для сервиса HAProxy (active-active).
4. **ocf::heartbeat:galera:galera-bundle** – для базы данных MariaDB (active-active).
5. **ocf::heartbeat:rabbitmq-cluster:rabbitmq-bundle** – для сервера очередей (active-active).
6. **ocf::heartbeat:redis:redis-bundle** – для кеширующего сервиса (active-passive).
7. **ocf::ovn:ovndb-servers:ovn-dbs-bundle** – для базы данных OVN (active-passive).
8. **ocf::heartbeat:podman:openstack-cinder-volume** – для сервиса блочного хранилища.

Pacemaker выполняется только на серверах управления.

Термины, сокращения и определения

Термин	Определение
1Stack (1Стек)	Облачная платформа виртуализации. Представляет собой программное решение для управления виртуализированными ресурсами: ▪ процессорной мощности и оперативной памяти физических серверов; ▪ сети передачи данных; ▪ систем хранения данных. и предоставления пользователю этих ресурсов в пределах выделенной квоты, с поддержкой виртуальной и физической изоляции ресурсов между различными пользователями
Affinity	Правила распределения виртуальных машин на один хост виртуализации с привязкой к определенным ресурсам (CPU, NUMA node)
AMQP	Advanced Message Queuing Protocol. Открытый протокол прикладного уровня для передачи сообщений между компонентами системы
Anti-affinity	Правила распределения виртуальных машин на разные хосты без привязки к ресурсам
API	Application Programming Interface/ Программный интерфейс приложения
AWS	Amazon Web Services. Коммерческое публичное облако, поддерживаемое и развиваемое компанией Amazon с 2006 года
CLI	Command Line Interface. Интерфейс командной строки
CPU	Central Processing Unit. Центральный процессор
DNS	Domain Name System. Распределённая система для получения информации о доменах
GENEVE	Туннельный протокол
HTTP	HyperText Transfer Protocol. Протокол передачи информации в интернете
HugePages	Метод управления памятью. Применяется для повышения производительности за счет использования блоков памяти большего размера, чем размер страницы по умолчанию
IaaC	Infrastructure as a Code. Подход к автоматизации и управлению инфраструктурой через использование кода
IGMP	Internet Group Management Protocol. Протокол управления групповой (multicast) передачей данных в сетях, основанных на протоколе IP
IPMI	Intelligent Platform Management Interface
Jumbo Frame	Фрейм Ethernet, незначительно превышающий размеры, которые разрешены стандартами сети Ethernet
LDAP	Lightweight Directory Access Protocol. Протокол быстрого доступа к каталогам
ML2	Плагин (фреймворк). Позволяет сети 1Stack одновременно использовать различные сетевые технологии уровня 2
NFS	Network File System. Сетевая файловая система

Термин	Определение
NUMA	Non-Uniform Memory Access. Архитектура организации компьютерной памяти. Используется в мультипроцессорных системах
OpenLDAP	OpenLDAP. Протокол облегченного доступа к каталогам с открытым исходным кодом
OVN	Open Virtual Network. Платформа сетевой виртуализации, которая отделяет физическую топологию сети от логической
OVS	Open vSwitch. Многоуровневый программный коммутатор. Используется для работы в качестве виртуального коммутатора в средах виртуальных машин
OVSDB	Протокол управления. Предназначен для обеспечения программного доступа к базе данных Open vSwitch
PCI	Peripheral Component Interconnect. Локальная компьютерная шина для подключения аппаратных устройств к компьютеру
QEMU	Программа с открытым исходным кодом для эмуляции аппаратного обеспечения различных платформ
QEMU-KVM	Kernel-based Virtual Machine – Virtual Machine Manager. Гипервизор, обеспечивающий виртуализацию в среде Linux
QoS	Quality of Service. Набор технологических решений для оптимизации сетевого трафика с помощью назначаемых приоритетов передачи информации
RAW	Формат образа тома данных с непосредственным доступом к данным
REST-API	Representational State Transfer. Способ создания API, использующий протокол передачи данных HTTP
RPC	Remote Procedure Calling. Механизм, исполняющий процедуры на стороне сервера, а не клиента
SAML	Security Assertion Markup Language. Язык разметки декларации безопасности. Открытый стандарт обмена данными аутентификации, основанный на языке XML
SNAT	Source Network Address Translation. Изменение сетевого адреса отправителя в заголовке пакета
SQL	Structured Query Language. Язык структурированных запросов. Декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных
SR-IOV	Single Root Input/Output Virtualization. Виртуализация ввода-вывода, которая применяется для виртуализации ресурсов ввода-вывода для отдельных серверов
SSH	Secure Shell. Безопасная оболочка – сетевой протокол прикладного уровня. Позволяет удалённо управлять операционной системой и туннелировать TCP-соединения
STF	Платформа Service Telemetry Framework. Автоматизирует сбор измерений и данных от удаленных клиентов и передачу информации на централизованную платформу Red Hat OpenShift Container Platform (OCP)
STONITH	Техника ограждения в компьютерных кластерах – изоляция вышедшего из строя узла таким образом, чтобы он не вызывал сбоев в работе компьютерного кластера
Syslog	Системный журнал

Термин	Определение
TCP	Transmission Control Protocol. Протокол управления передачей данных
UDP	User Datagram Protocol. Протокол пользовательских датаграмм
VLAN	Virtual Local Area Network. Виртуальная локальная компьютерная сеть
VNC	Virtual Network Computing. Метод удаленного доступа к рабочему столу компьютера по сети
VNI	Virtual Network Index. Идентификатор сети в рамках VxLAN
VXLAN	Virtual Extensible LAN. Технология сетевой виртуализации для решения проблем масштабируемости в больших системах облачных вычислений
YAML	Yet Another Markup Language. Формат сериализации данных, разработанный для удобства чтения и записи
БД	База данных
Инстанс	Виртуальная машина
ОС	Операционная система
ПО	Программное обеспечение
Сокет	Программный интерфейс для обеспечения обмена данными между процессами
СУБД	Система управления базами данных
Транковый интерфейс	Магистральный интерфейс, канал передачи данных. Совокупность технических средств и правил, обеспечивающих обмен информацией между абонентами интерфейса последовательным кодом по общей информационной магистрали
ЦОД	Центр обработки данных