Создание виртуальной машины – во вкладке compute claud

Подключение к виртуальной машине виндовс – если через винду - в пуске подключение к удаленному рабочему столу, вводим новая учетная запись Administrator и пароль, в линуксе через отдельное приложение – Vinagre

Практическая работа №3

К виртуальной машине линукс – через терминал линукс - при создании ВМ втавляем ssh ключ, генерируем его ssh-keygen -t rsa -b 2048

Подключение к вирт машине ssh <имя\_пользователя>@<публичный\_IP-адрес\_виртуальной машины>

Вход в виртуальную машину через линукс

ssh -t -p 9600 -o IdentitiesOnly=yes -i ~/.ssh/<имя закрытого ключа> <ID виртуальной машины>.<имя пользователя>@serialssh.cloud.yandex.net

ssh -t -p 9600 -o IdentitiesOnly=yes -i ~/.ssh/id\_rsa [fhm0b28lgfp4tkoa3jl6.yc-user@serialssh.cloud.yandex.net](mailto:fhm0b28lgfp4tkoa3jl6.yc-user@serialssh.cloud.yandex.net)

практическа работа 5

Слева вкладка мониторинг – там нагрузка цпу, сетевая активность, нагрузка диска и тп\

Диски, снимки и образы

Ключевое понятие здесь — **блок размещения**, т. е. единица выделения дискового пространства. Чем больше блоков размещения в вашем диске, тем более производительный диск в итоге вы получаете. У сетевых SSD размер блока размещения — 32 ГБ, у сетевых HDD он равен 256 ГБ. Поэтому при одинаковом объёме у SSD больше блоков размещения, а значит, выше IOPS и пропускная способность.

Данные ВМ в Compute Cloud хранятся на загрузочном и дополнительных дисках. На загрузочном диске находится операционная система и приложения, поэтому его нельзя отключить от ВМ. На дополнительном диске вы можете хранить любые данные, его можно отключить от одной ВМ и подключить к другой.

Поменять пароль 1. Войти ssh [bjiaduk@84.201.152.78](mailto:bjiaduk@84.201.152.78)

2. sudo passwd bjiaduk

Затем в веб-консоли откройте страницу ВМ и через меню слева перейдите на страницу **Серийная консоль**.

Создание снимка диска На странице **Compute Cloud Сервис** откройте **Диски**. Справа от диска нажмите значок ... и выберите **Создать снимок**.

Снимки – бекап диска на определнный момент времени

Восстановление из диска - При создании загрузочного диска машины выберите готовый снимок диска. Для этого в разделе **Выбор образа/загрузочного диска** перейдите на вкладку **Пользовательские** и нажмите кнопку **Выбрать**.

Кароче похлже образ – это тот же снимок,только это готовый образ не наш, то есть не создпнный пользовактелем, а готовый + с каким-то пприложением

# Практическая работа 7. Создание новой сети с подсетями и ВМ

В каталоге перейдите в раздел **Сервисы**, выберите **Virtual Private Cloud** и нажмите кнопку **Создать сеть**. Сеть облаков связывает облака друг с другом

# Балнсировщик - Yandex Network Load Balancer

Целевая группа – 5 вирт машин, по которым распределяется загрузка от сайта

Обрабочик – распределяет ресурсы на эти 5 машин

# Практическая работа 10. Создание балансировщика

В **Network Load Balancer** в разделе **Целевые группы** нажмите кнопку **Создать целевую группу**. На открывшейся странице введите имя целевой группы (например, demo-web), выберите обе ВМ, созданные на предыдущем уроке, и нажмите кнопку **Создать**.

# Практическая работа 11. Создание группы виртуальных машин

На странице сервиса **Compute Cloud** откройте **Группы виртуальных машин** и нажмите кнопку **Создать группу**.

# Практическая работа 12. Автоматическое масштабирование под нагрузкой

На странице **Создание группы виртуальных машин** сервиса **Compute Cloud** перейдите к блоку **Масштабирование** и выберите тип **Автоматический**.

# Модуль 2

**Способ 2. Управляемые базы данных**

При использовании этих сервисов вы будете настраивать БД через консоль управления в браузере, CLI Yandex.Cloud или Yandex.Cloud API, а подключаться к БД из приложений (точно так же, как к обычным БД).

# Кластер — это одна или несколько виртуальных машин (или хостов), где разворачивается БД. Вирт машин несколько, так как это позволяет разделить нагрузку, если резкий скачок, то можно добавить вирт машины, если выход из строя, то заменить одну из них.

# **Реляционная БД** — это набор таблиц и связей между ними. Если мы разрабатываем мессенджер, то нам понадобится хранить в БД сведения обо всех пользователях, чатах и сообщениях. Данные нужно четко структурировать и свести в связанные между собой таблицы. В одной будут данные о пользователях (ID, ник, ссылка на аватарку, номер телефона и город проживания), в другой — о сообщениях (номер, текст, кто, кому и когда написал сообщение, в каком чате, прочитано ли оно).

# Нереляционные (NoSQL) БД обеспечивают высокую доступность, быстро работают и хорошо масштабируются. Однако зачастую это достигается за счет ограничения транзакционных возможностей, которыми славятся реляционные базы. Транзакцией называется последовательность операций над данными имеющая начало и конец – то есть количество воозможных действий с данными, хорошо когда высокая транзакционная возможность и высокая скорость транзакций.

# Всем этим требованиям отвечает **объектное хранилище** (англ. object storage) — система хранения данных произвольного формата. В хранилище записываются **объекты** — по сути, привычные нам файлы. У каждого объекта есть уникальный идентификатор (аналог имени файла) и метаданные (дополнительные сведения). Идентификатор помогает различать объекты, а в метаданных хранятся, например, дата создания и MD5-хеш.

# Внутри хранилища объекты группируются в **бакеты**. Это позволяет разделять данные разных проектов или пользователей. У каждого бакета в Yandex.Cloud уникальное имя. Переименовать бакет нельзя, поэтому выбирайте имя для бакета с умом.

# Управление доступом

В объектном хранилище Yandex.Cloud для контроля доступа используются три независимых механизма:

* сервис управления доступом (IAM, Identity and Access Management) работает на уровне всего облака;
* список управления доступом (ACL, Access Control List) работает на уровне бакетов и объектов;
* политика доступа (Bucket Policy) задает условия доступа к бакетам и объектам.

# Практическая работа. Создание бакетов и загрузка объектов. Работа с утилитой S3cmd

Выберите на стартовой странице консоли управления сервис **Object Storage**.

1. Нажмите кнопку **Создать бакет**. Откроется окно с основными параметрами:

# Оркестрация и Kubernetes

Главное и принципиальное отличие **контейнера** от ВМ в том, что он использует ресурсы и ядро хостовой ОС. Несколько контейнеров, размещенных на одном сервере, используют ресурсы сервера совместно, тем самым экономя их. – кароче более удобная и дешевая – ВМ на которой тоже распологается какой-нибудь сервис то есть сайт и тд.

Оркестрация – управление контейнерами – масштабирование, удаление, прибавление и тд.