

Введение в виртуализацию. Типы и функции гипервизоров. Обзор рынка вендоров и областей применения.

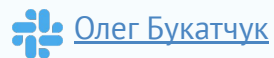


Олег
Букатчук



Олег Букатчук

Software Architect DevOps, crif.com



План занятия

1. [Что такое виртуализация?](#)
2. [Что такое гипервизор?](#)
3. [Типы виртуализации](#)
4. [Коммерческие продукты](#)
5. [Open Source продукты](#)
6. [IaaS продукты](#)
7. [Аппаратная виртуализации vs. Виртуализации уровня ОС](#)
8. [Применение виртуализации: плюсы и минусы](#)
9. [Итоги](#)
10. [Домашнее задание](#)



Что такое виртуализация?

Что такое виртуализация?

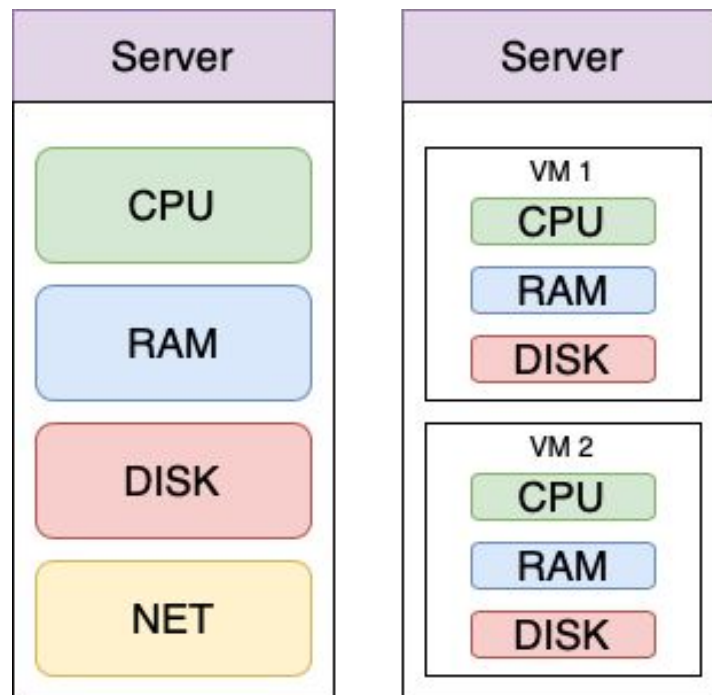
Виртуализация — это создание изолированных окружений в рамках одного физического устройства (сервера).

Каждое окружение при этом выглядит, как отдельный компьютер со своими характеристиками, такими как:

- процессор (количество выделенных ядер),
- оперативная память,
- диски,
- сеть.

Такое окружение называют набором логических ресурсов или виртуальной машиной.

Наборы ресурсов виртуальных машин





Что такое гипервизор?

Что такое гипервизор?

Гипервизор — это монитор виртуальных машин — программа, обеспечивающая одновременное (параллельное) выполнение нескольких операционных систем (виртуальных машин) на одном и том же физическом устройстве (сервере).

Основные задачи гипервизора:

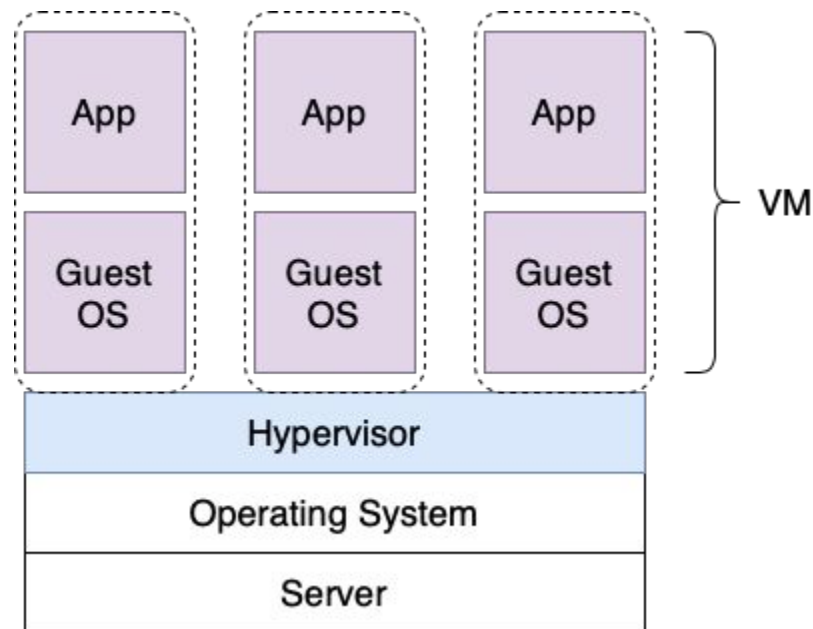
- эмуляция аппаратных ресурсов,
- безопасное выполнение машинных инструкций,
- предотвращение выполнения команд гостевых операционных систем в режиме супервизора на хост-машине (исключение перехвата и анализа команд).

Функции гипервизора

- Гипервизор **обеспечивает изоляцию операционных систем друг от друга**, защиту runtime и безопасность, а также разделение ресурсов между различными запущенными ВМ.
- Гипервизор **предоставляет** работающим под его управлением ВМ **средства связи и взаимодействия** между собой таким образом, как если бы эти ОС выполнялись на разных физических компьютерах.
- Гипервизор гарантирует **независимое «включение», «перезагрузку» и «выключение» каждой ВМ** с той или иной операционной системой установленной на управляемую ВМ.

Как работает гипервизор?

На примере: паравиртуализации.





Типы виртуализации

Типы виртуализации

- Полная (аппаратная) виртуализация.

Гипервизоры первого типа работают на аппаратном уровне без необходимости установки какой-либо ОС на хост. Они сами являются ОС.

- Паравиртуализация.

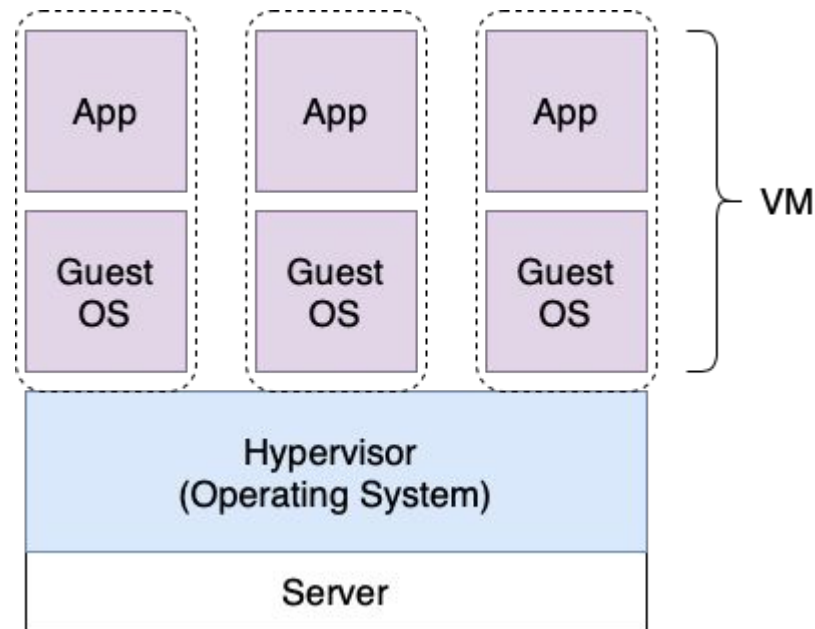
Гипервизорам второго типа необходима ОС для доступа монитора виртуальных машин (гипервизора) к аппаратным ресурсам хоста.

- Виртуализация уровня операционной системы.

Виртуализация уровня ОС позволяет запускать изолированные и безопасные ВМ на одном хосте, но не позволяет запускать ОС с ядрами, отличными от типа ядра хостовой ОС.

Полная (аппаратная) виртуализация

Использует менеджер виртуальных машин (гипервизор), который осуществляет связь между гостевой операционной системой и аппаратными средствами физического сервера.

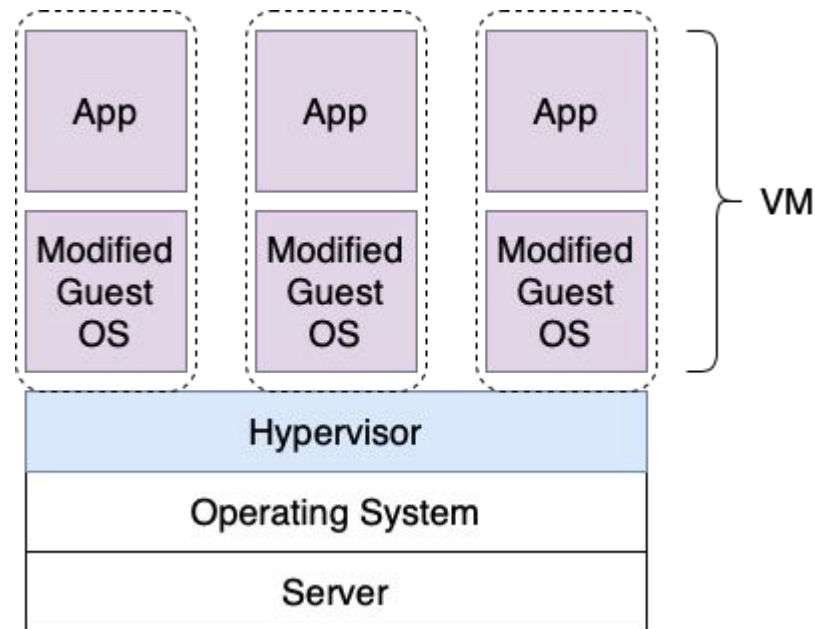


Примеры решений использующих полную (аппаратную) виртуализацию



Паравиртуализация

Паравиртуализация разделяет процесс с гостевой ОС.
Гипервизор модифицирует ядро гостевой ВМ для разделения доступа к аппаратным средствам физического сервера.



Примеры решений использующих паравиртуализацию



VirtualBox

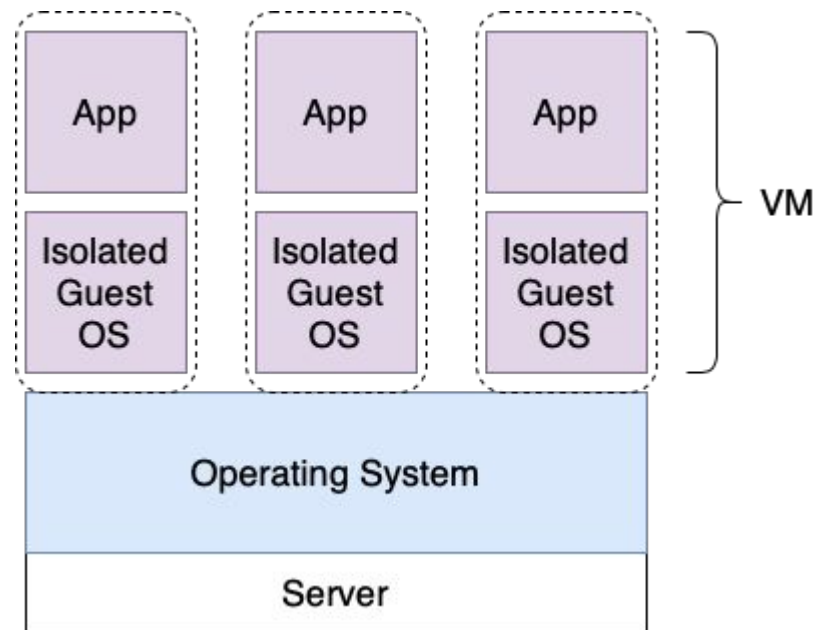


Microsoft
Hyper-V

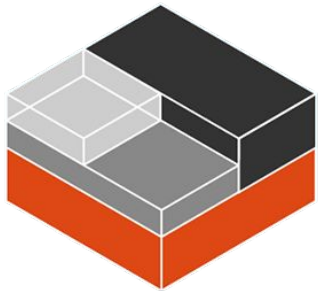


Виртуализация уровня ОС

Позволяет запускать изолированные и безопасные ВМ на одном хосте, но не позволяет запускать ОС с ядрами, отличными от типа ядра базовой операционной системы.



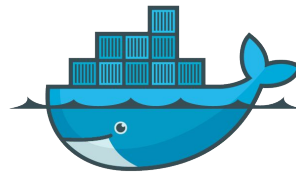
Примеры решений использующих виртуализацию уровня ОС



LXC



OpenVZ



docker

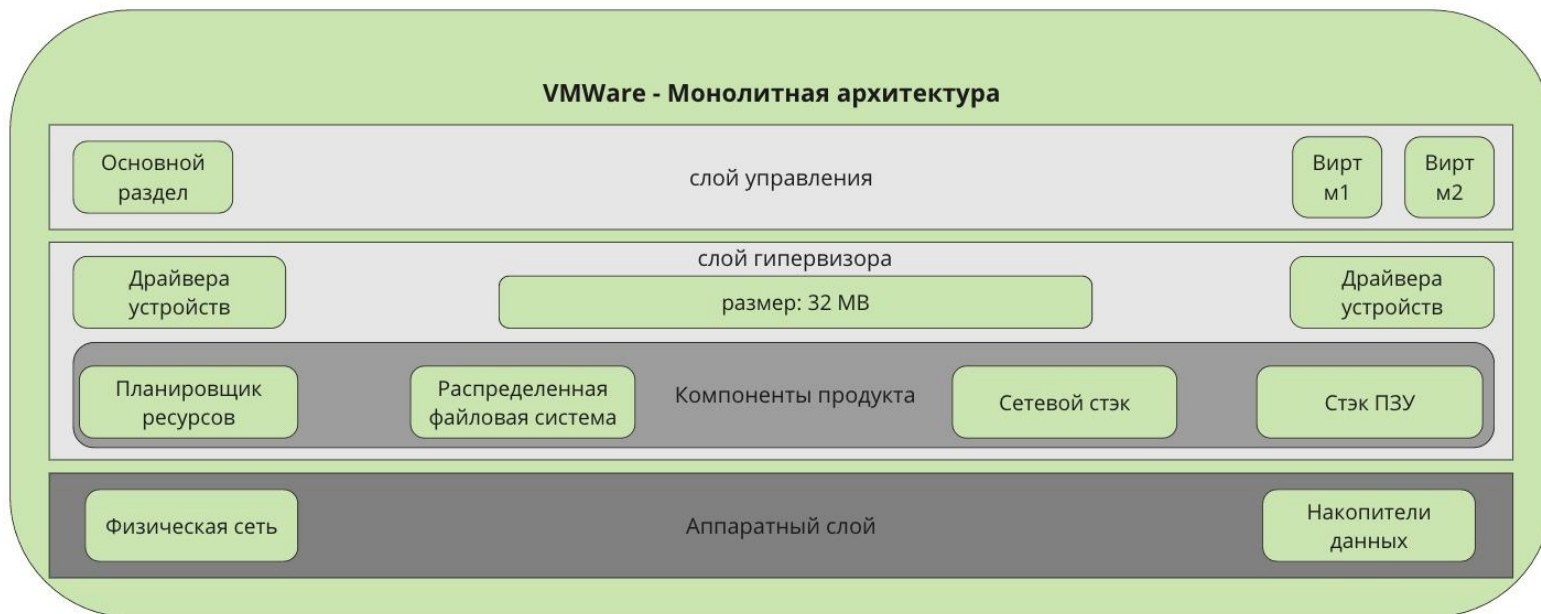


* Важно понимать, что Docker не является гипервизором в чистом виде, это технология со своей собственной архитектурой, но подход к управлению ресурсами хостовой машины у Docker очень похож на LXC-контейнеры.



Коммерческие продукты

VMWare



VMWare

VMWare предлагает множество решений для виртуализации, в частности vSphere, который используется, как основа для кластеризации и обеспечения отказоустойчивости.

VMware's vSphere использует монолитный дизайн гипервизора, который требует, чтобы драйверы устройств были включены в слой Hypervisor.

В слое гипервизора находятся следующие компоненты:

- планировщик ресурсов,
- распределенная файловая система,
- сетевой стек,
- стек хранения данных.

VMWare

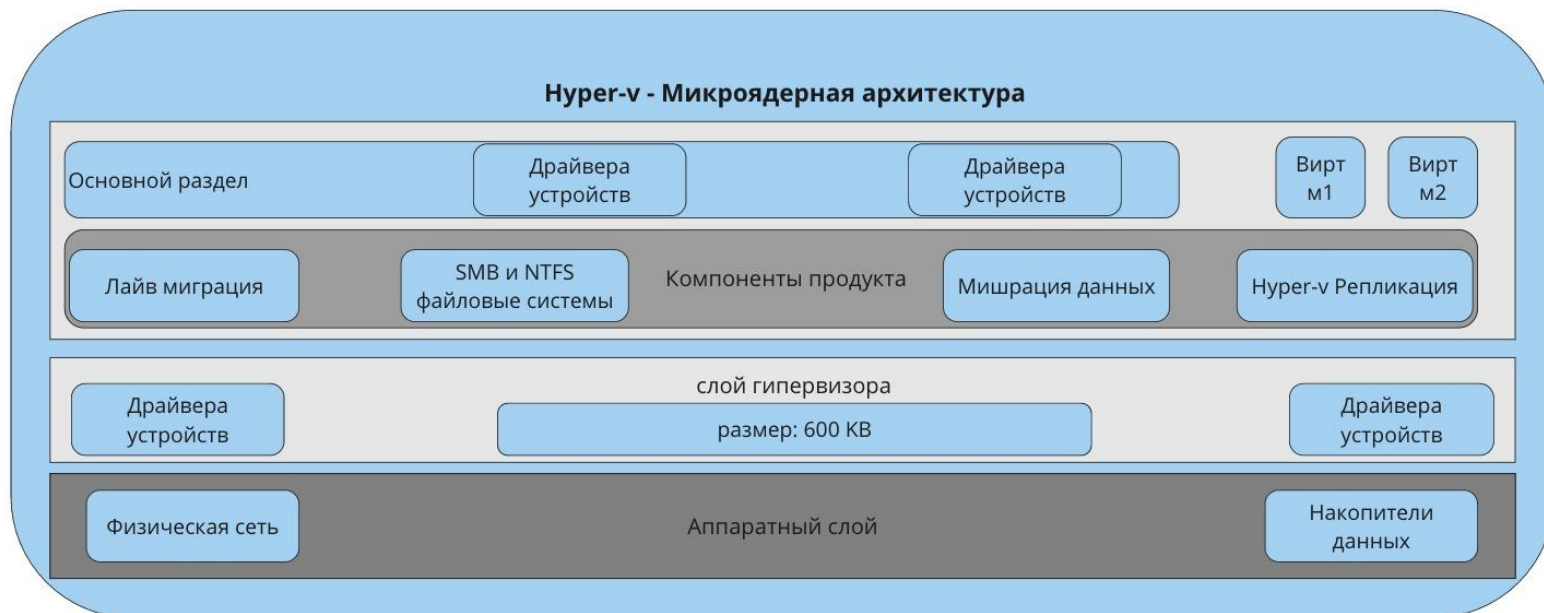
Преимущества:

- Нет ограничений по выбору операционной системы, необходимой для управления всеми компонентами.
- Нет необходимости в патчах безопасности, для слоя управления.
- Высокий уровень безопасности.
- Хорошая поддержка.

Недостатки:

- Не работает с оборудованием, которое не поддерживает VMWare.
- Требуется высокий уровень подготовки инженеров.

Hyper-V





Hyper-V

Hyper-V от Microsoft встроен непосредственно в Windows Server, хотя он также может быть установлен как Hyper-V Server, который является автономным.

Hyper-V использует микроядерную архитектуру, поэтому драйверы устройств работают независимо друг от друга в управляемом слое.

Hyper-V

В слое управления находятся следующие **компоненты** продукта:

- Live-миграция виртуальных машин.
- Hyper-V реплики.
- SMB и NTFS (файловые системы).
- Миграция накопителей данных.
- Сетевой стек и стек хранения данных не являются частью компонентов продукта, а являются частью гипервизора.

Hyper-V

Преимущества:

- Упрощенное управления драйверами устройств, широкий диапазон поддерживаемых устройств.
- Простота установки новых ролей сервера.
- Нет прерывания сервиса для обслуживания или обновления безопасности.
- Сервисы могут быть масштабированы;
- Более низкий порог вхождения для инженеров.

Недостатки:

- Необходимость установки ОС для работы слоя гипервизора.
- Ограниченная поддержка версий ОС.



Open source продукты



KVM

KVM является представителем паравиртуализации, вы можете запустить практически любую операционную систему в качестве гостевой — BSD / Windows / Linux и с драйвером virtio вы получите близкую к нативной производительность.

KVM поддерживает установку из ISO образа, а также шаблоны установки, он поставляется с хорошим уровнем реализации безопасности, может иметь проблемы с I/O под тяжелой нагрузкой, которая влияет на гостевые и хостовую операционную системы.



KVM

Каждый гостевая машина работает, как процесс на хостовой машине, что хорошо для обнаружения источника проблемы, но также при проблемах с доступными ресурсами на хостовой машине все гостевые подвержены проблемам с производительностью.



KVM

KVM является нативным для большинства современных ядер Linux.

Это дает преимущество в производительности по сравнению с другими системами виртуализации. При этом данный продукт относительно новый и находится в активном развитии.

Большинство людей выбирают KVM из-за более низкого порога входа, хотя эта система не столь стабильна, как Xen.



Xen

Xen поставляется в двух вариациях, и может работать одновременно на том же физическом хосте в режиме **Xen PV** (паравиртуализация) и **HVM** (полная аппаратная виртуализация).

Гостевые машины в режиме Xen PV обычно основаны на шаблонах для быстрого развертывания и высокой производительности.

Вы можете запускать свое собственное ядро в Xen PV. По умолчанию вы можете запустить только Linux в этом режиме. Запуск BSD возможен с дополнительной конфигурацией.



Xen

Режим Xen HVM работает примерно так же, как KVM. Он имеет лучшие драйверы для Linux-дистрибутивов, однако в NetBSD и Windows, Xen HVM показывает себя хуже по сравнению с KVM, в то время как Xen PV и Windows более совместимы.

Xen очень зрелый продукт, большинство людей выбирают его для хорошей производительности и исключительной стабильности.

Стабильность Xen обеспечивается, за счет, предварительно выделяемой RAM и CPU для гипервизора, а так как он имеет свои собственные выделенные ресурсы, то не него не могут повлиять гостевые ОС, что и дает пресловутую стабильность в работе.

Все против всех: VMWare vs. Hyper-V vs. KVM vs. Xen

- **VMware vSphere** является наиболее сбалансированным и универсальным продуктом для организаций с высокими требованиями к их виртуальной инфраструктуре.
- **Hyper-V** является выбором defacto для окружений с преобладанием технологий Microsoft.
- Если вы хотите сэкономить деньги и ваши окружения представляют собой исключительно Linux, то **KVM** может стать неплохим вариантом платформы виртуализации.
- **Xen** с его надежностью и стабильностью подойдет для большинства пользователей, как универсальный гипервизор.

Необходимо отметить, что оптимальный выбор системы очень зависит от конкретного сценария развертывания.



IaaS продукты

AWS EC2

Сервисы облачных провайдеров — самая современная реализация систем управления виртуализацией. Гипервизоры и прочая реализация системы скрыта от пользователя, при этом есть гарантированные показатели доступности и отказоустойчивости.

Пользователь может создавать инфраструктуру продуктивных окружений и использовать уникальные преимущества такие, как установка в нескольких локациях, построение цепочки с другими сервисами облака, программное API для автоматизации создания и изменения ресурсов с помощью Terraform и Packer.

OpenStack


Приватные облака являются программной надстройкой для организации приватного (частного) облака, и дают часть преимуществ:

- Построение продуктивных цепочек с другими компонентами вашей частной инфраструктуры.
- Программное API для автоматизации создания и изменения ресурсов приватного облака с помощью Terraform и Packer.

Единственный, но очень весомый недостаток:

- Очень высокий порог входа в технологию, вам потребуется выделенная команда инженеров для поддержки продуктовой инсталляции приватного облака.

Приватные облака имеют более высокий уровень безопасности, необходимый для некоторых бизнес проектов и часто используются в банках и других финансовых учреждениях.



Аппаратная виртуализация vs. виртуализации уровня ОС

Аппаратная виртуализация vs. виртуализации уровня ОС

Аппаратная виртуализация	Виртуализация уровня ОС
<ul style="list-style-type: none">● Виртуальные машины содержат собственное виртуальные устройства.● Программное обеспечение может запускаться в виртуальных машинах без необходимости модификации.● Виртуальные машины полностью изолированы друг от друга и используют собственные ядра ОС.	<ul style="list-style-type: none">● Контейнеры монтируют устройства хост машины, но выглядят как обычная VM.● Сторонние приложения могут запускаться в контейнерах без необходимости модификации.● Контейнеры полностью изолированы друг от друга, но используют одно ядро ОС.
<ul style="list-style-type: none">● Возможность создания множества виртуальных машин с различными операционными системами.● Пользователь может устанавливать собственные патчи на ядро.	<ul style="list-style-type: none">● Возможность создавать машины только с Linux или только Windows системами.● Контейнеры разделяют ядро системы, работая как отдельный процесс основной ОС.
<ul style="list-style-type: none">● Жесткое распределение системных ресурсов между виртуальными машинами на уровне гипервизора.	<ul style="list-style-type: none">● Потребление системных ресурсов: CPU, RAM, DISK могут ограничиваться отдельно для каждого контейнера.



Плюсы и минусы виртуализации

Плюсы и минусы виртуализации

Плюсы

- Высокая утилизация вычислительных мощностей.
- Снижение простоев за счет быстрого развёртывания.
- Возможность использовать подход Infrastructure as a Code.

Минусы

- Накладные расходы на обслуживание.
- Увеличение числа абстракций.
- Единая точка отказа, если не используется кластеризация.



Итоги

Итоги

Сегодня мы:

- Рассмотрели, что такое виртуализация и гипервизор;
- Узнали о преимуществах виртуализации;
- Рассмотрели типы виртуализации и сравнили их;
- Рассмотрели плюсы и минусы виртуализации;
- Познакомились с основными системами управления виртуализацией, их классификацией, преимуществами и недостатками каждой из них;
- Также узнали о более комплексном и современном подходе таком, как “инфраструктура как сервис” и его реализациях в виде публичных и частных облаков.



Домашнее задание

Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задавайте **в чате** мессенджера Slack.
- Задачи можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**.

**Задавайте вопросы и
пишите отзыв о лекции!**

Олег Букатчук