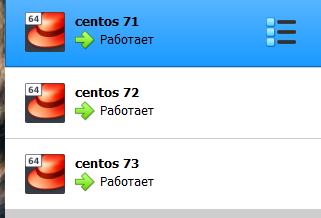
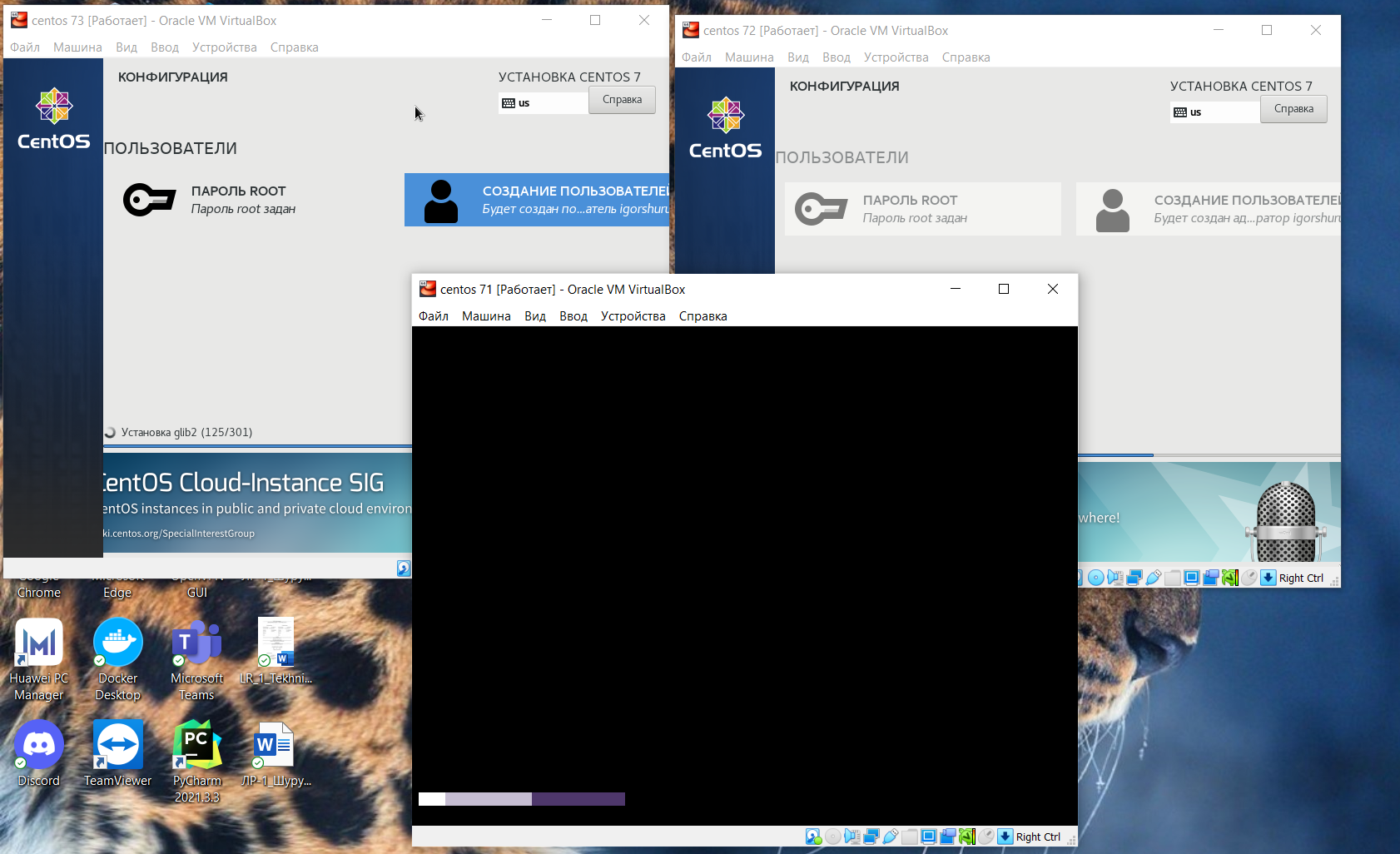
1. Установить гипервизор на выбор студента

Уже был установлен

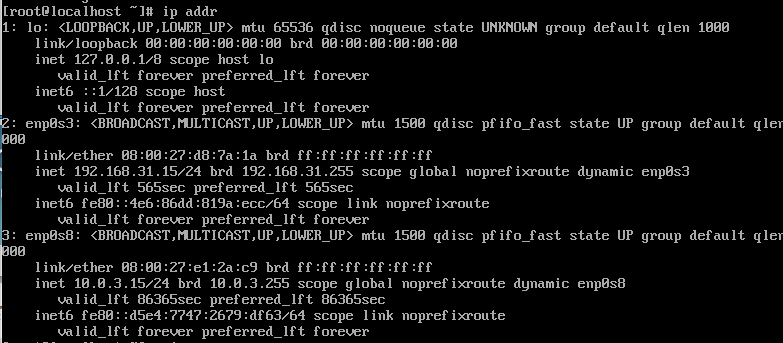


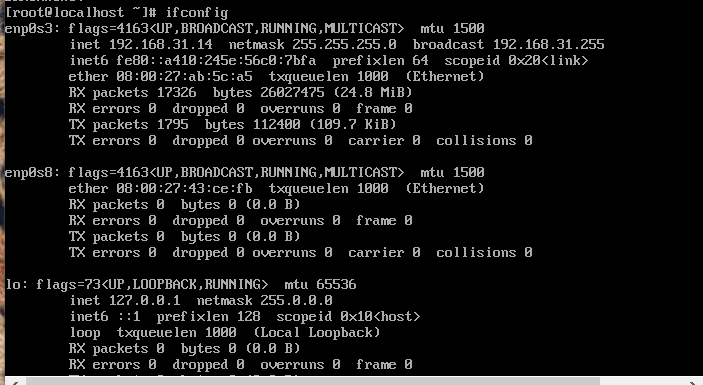
1. Создать несколько виртуальных машин (от 3х ВМ) с Unix-подобными ОС для стендов test, stage и prod соответственно

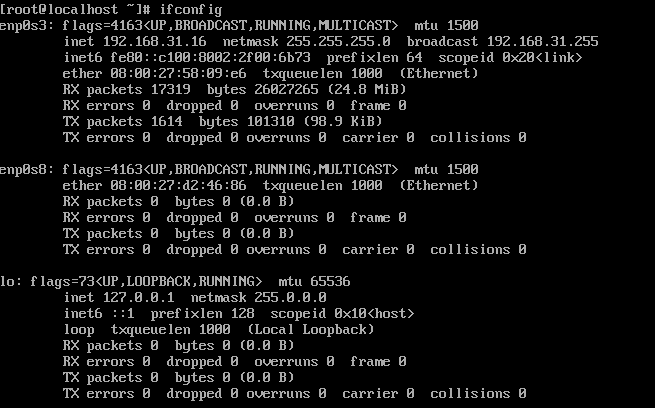




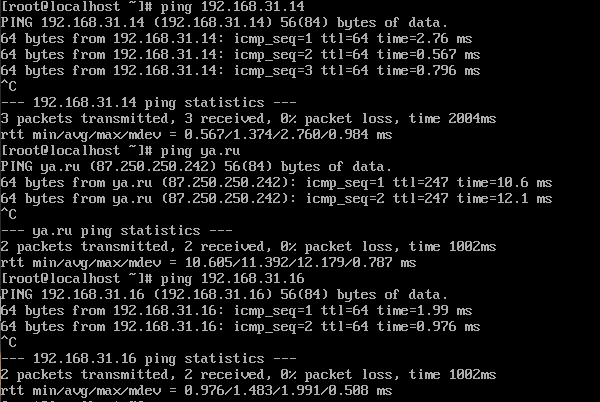
1. Объединить созданные виртуальные машину в единую локальную сеть (ping должен работать по принципу “от всех ко всем”)

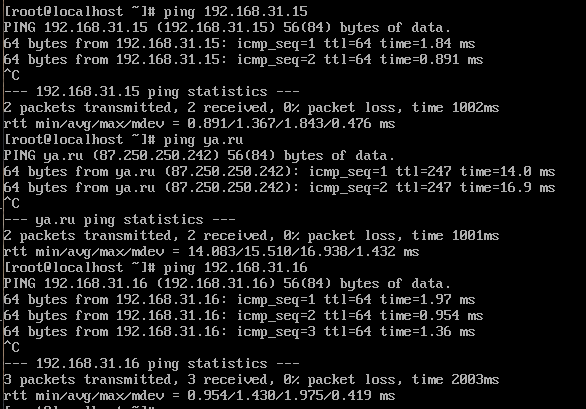


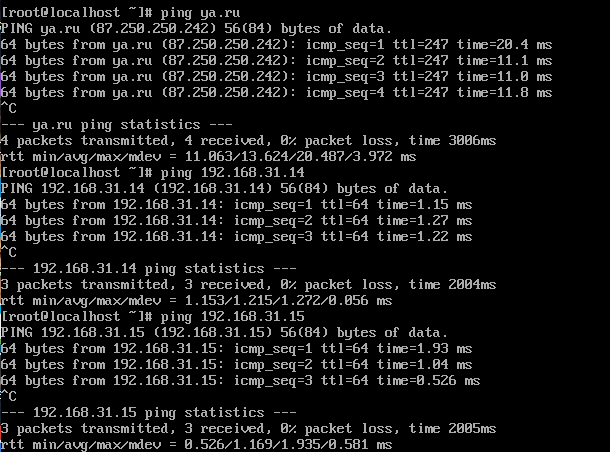




Проверяем ping

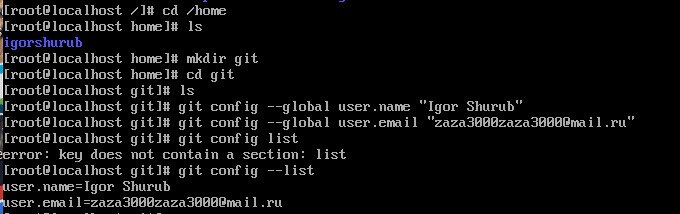


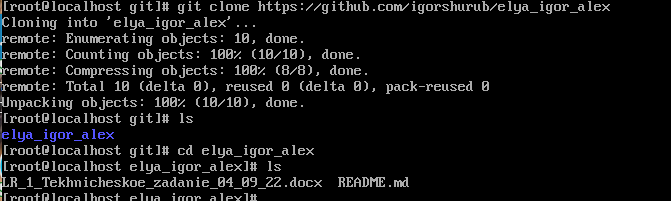




1. Разместить на тестовом стенде файлы из удаленного репозитория проекта (при помощи утилиты Git)







Повторим данное действие на 3 виртуалках

1. Развернуть на стендах средства разработки, согласно выбранному для проекта стеку. Если выбраны скриптовые ЯП, то запустить прототип, в ином случае - собрать приложение.















1. Что общего и чем отличаются различные типы сетей для виртуальных машин (virtual network, bridge, nat)?

Каждый из этих видов сетевого взаимодействия может применяться для различных вариантов использования виртуальных машин и необходимо тщательно подбирать тип сетевого взаимодействия виртуальной машины для более эффективного ее использования совместно с другими компонентами сетевой инфраструктуры.

При подключении типа "Виртуальный адаптер хоста" (virtual network) гостевые ОС могут взаимодействовать между собой, а также с хостом. Но все это только внутри самой виртуальной машины. Также машинам создается подсеть и назначаются IP-адреса сетевым картам гостевых операционных систем. Гостевые ОС не могут взаимодействовать с устройствами, находящимися во внешней сети, так как они не подключены к ней через физический интерфейс.

В соединении типа "Сетевой мост" (bridge) виртуальная машина работает также, как и все остальные компьютеры в сети. В этом случае адаптер выступает в роли моста между виртуальной и физической сетями. Со стороны внешней сети имеется возможность напрямую соединяться с гостевой операционной системой.

В режиме NAT гостевому сетевому интерфейсу присваивается по умолчанию IPv4 адрес из диапазона 10.0.х.0/24, где х обозначает конкретный адрес NAT-интерфейса, определяемый по формуле +2. Таким образом, х будет равен 2, если имеется только один активный NAT-интерфейс. В этом случае, гостевая операционная система получает IP-адрес 10.0.2.15, сетевому шлюзу назначается адрес 10.0.2.2, серверу имен (DNS) назначается адрес 10.0.2.3. В режиме NAT также недоступны такие возможности, как предоставление общего доступа к папкам и файлам.  
2. Каким образом осуществляется обработка системных вызовов Unix в виртуальных машинах при их развертке в ОС Windows?

Существуют следующие виды виртуализации:

* **Полная виртуализация**
* **Паравиртуализация**
* **Гибридная виртуализация**
* **Виртуализация на уровне ОС**

**Полная виртуализация:**

Виртуальная машина имитирует оборудование, чтобы позволить немодифицированной гостевой ОС работать изолированно. На корпоративном рынке существует два типа полной виртуализации. В обоих типах полной виртуализации исходная информация гостевой операционной системы не будет изменена.

* Полная виртуализация с помощью программного обеспечения
* Полная аппаратная виртуализация

**Паравиртуализация:**

Паравиртуализация работает иначе, чем полная виртуализация. Не нужно имитировать аппаратное обеспечение для виртуальных машин. Гипервизор устанавливается на физический сервер (хост), а в среду устанавливается гостевая ОС. Виртуальные гости знают, что он был виртуализирован, в отличие от полной виртуализации (где гость не знает, что он был виртуализирован), чтобы воспользоваться преимуществами функций. В этом методе виртуализации исходные коды гостей будут изменены с использованием конфиденциальной информации для связи с хостом. Гостевым операционным системам требуются расширения для выполнения вызовов API к гипервизору. При полной виртуализации гости будут выполнять аппаратные вызовы, но при паравиртуализации гости будут напрямую общаться с хостом (гипервизором) с помощью драйверов. Вот список продуктов, поддерживающих паравиртуализацию.

**Гибридная виртуализация: (виртуализация оборудования с драйверами PV)**

При полной виртуализации с аппаратной поддержкой гостевые операционные системы не изменяются и включают множество ловушек виртуальных машин и, следовательно, высокие накладные расходы ЦП, которые ограничивают масштабируемость. Паравиртуализация — это сложный метод, при котором гостевое ядро ​​необходимо модифицировать для внедрения API. Учитывая эти проблемы, инженеры пришли к гибридной паравиртуализации. Это комбинация полной и паравиртуализации. Виртуальная машина использует паравиртуализацию для определенных драйверов оборудования (там, где полная виртуализация является узким местом, особенно при интенсивных рабочих нагрузках ввода-вывода и памяти), а хост использует полную виртуализацию для других функций. Следующие продукты поддерживают гибридную виртуализацию.

* Виртуальная машина Oracle для x86
* Ксен
* VMware ESXi

**Виртуализация на уровне ОС:**

Широко используется виртуализация на уровне операционной системы. Она также известна как «контейнеризация». Ядро операционной системы хоста позволяет использовать несколько пользовательских пространств, также называемых экземпляром. В виртуализации на уровне ОС, в отличие от других технологий виртуализации, будет очень мало или совсем не будет накладных расходов, поскольку для выполнения используется ядро ​​​​операционной системы хоста. Зона Oracle Solaris — один из известных контейнеров на корпоративном рынке.

3. Как размещаются на жестком диске хостовой машины виртуальные жесткие диски виртуальной машины, если их файловые системы различны?

Все настройки виртуальной машины хранятся в файлах с расширением .vmx в папке которую задал пользователь, а файлы виртуальных дисков имеют расширение (.vmdk) и также хранятся в этой папке.