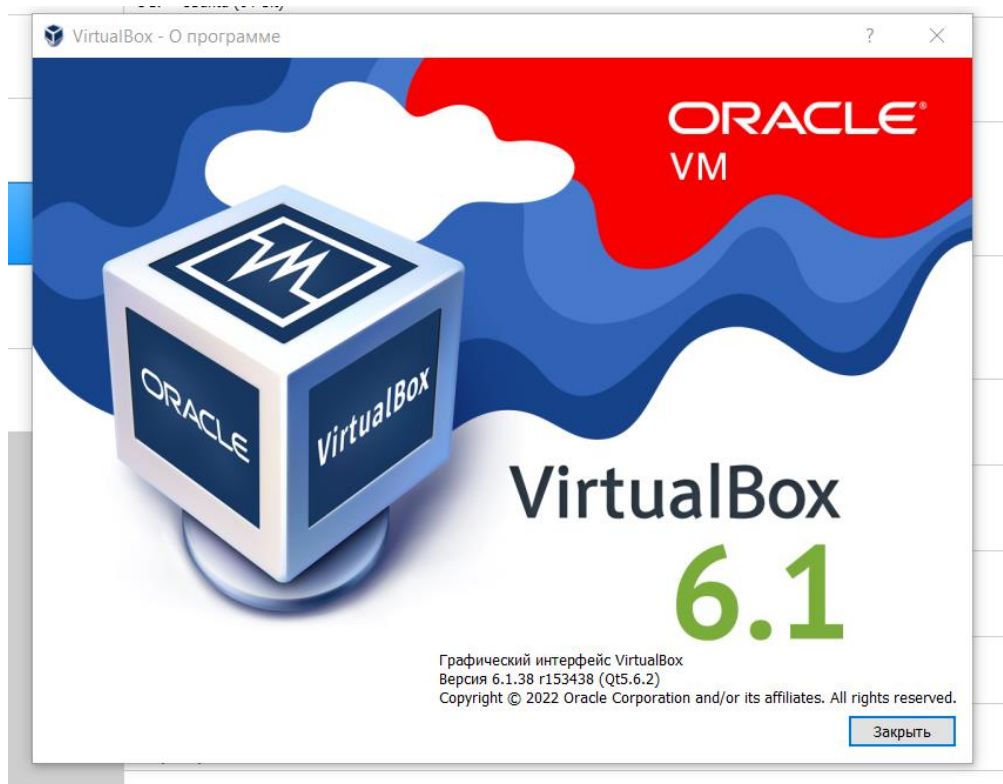


Лаб 2 автоматизация

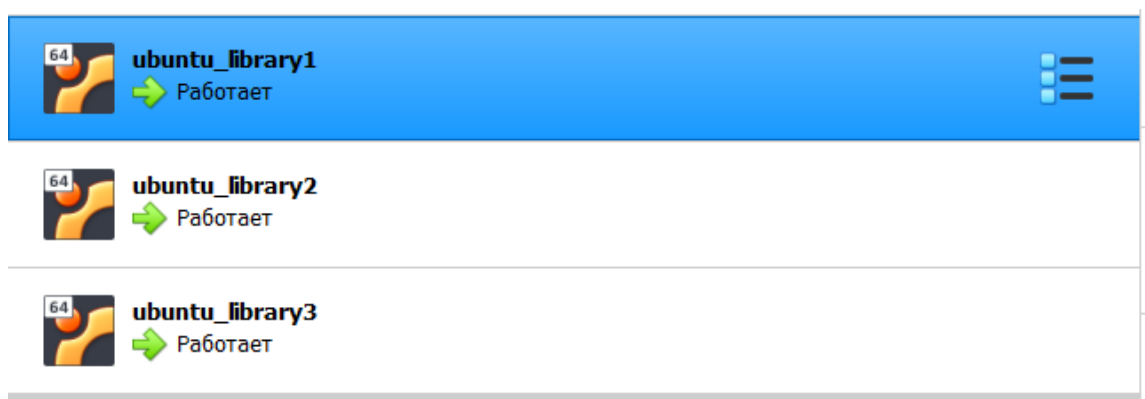
1. Установить гипервизор на выбор студента

Установлен VirtualBox



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

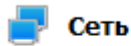
Создано 3 VM на ubuntu



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

Настройки машин:

Аудиоконтроллер: ICH AC97



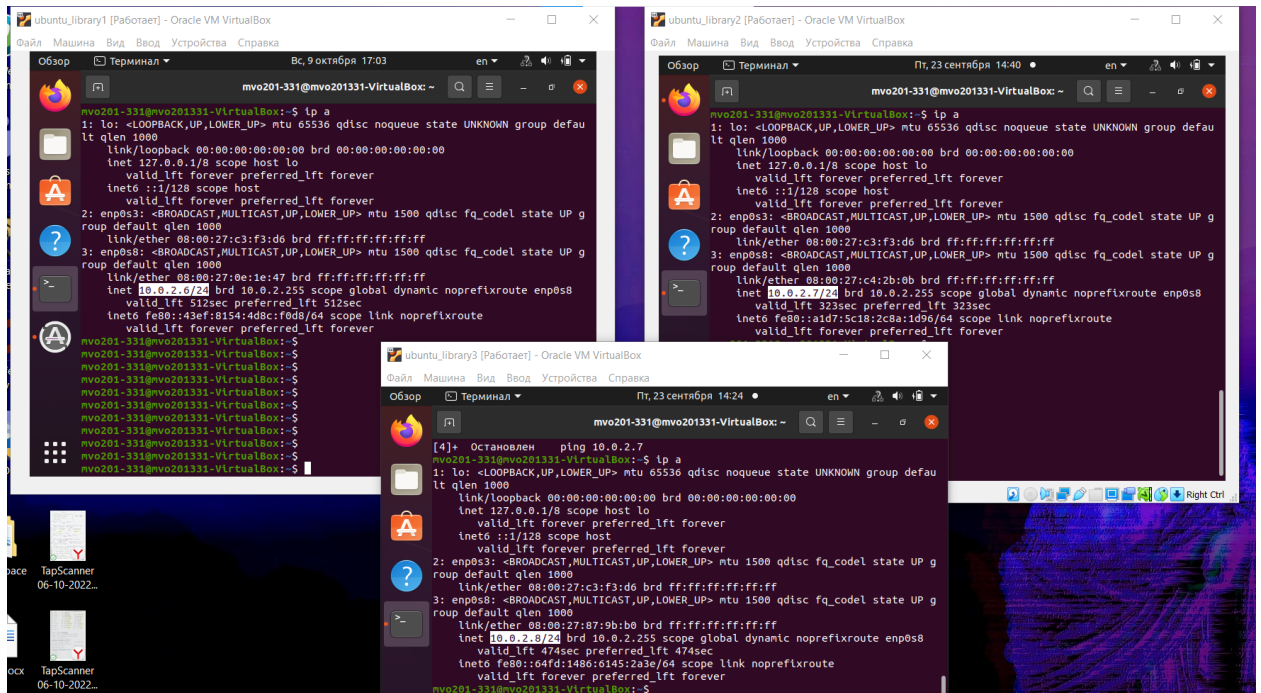
Сеть

Адаптер 1: Intel PRO/1000 MT Desktop (NAT)

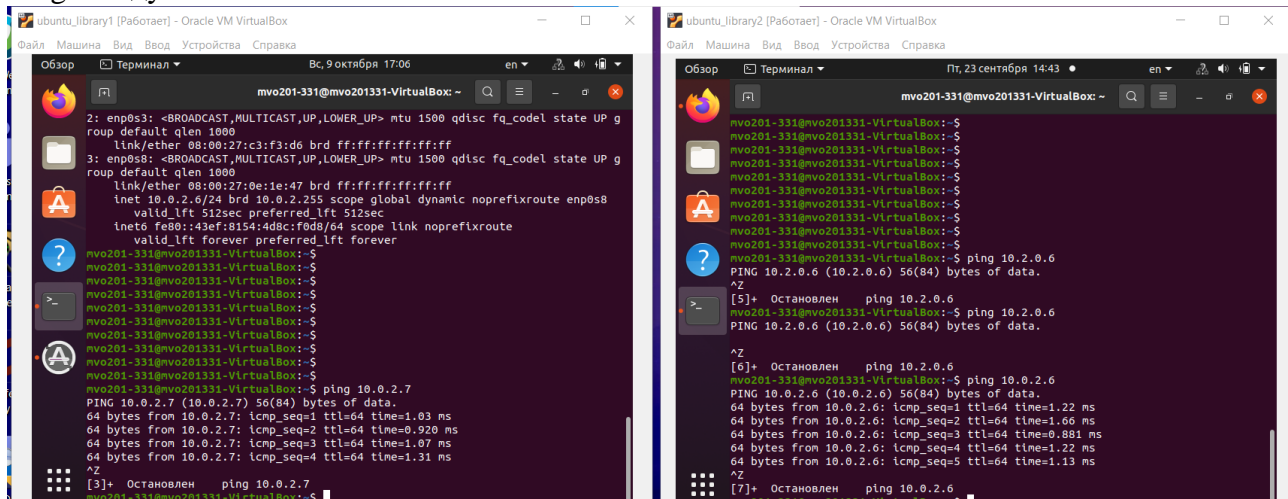
Адаптер 2: Intel PRO/1000 MT Desktop (Сеть NAT, 'NatNetwork')



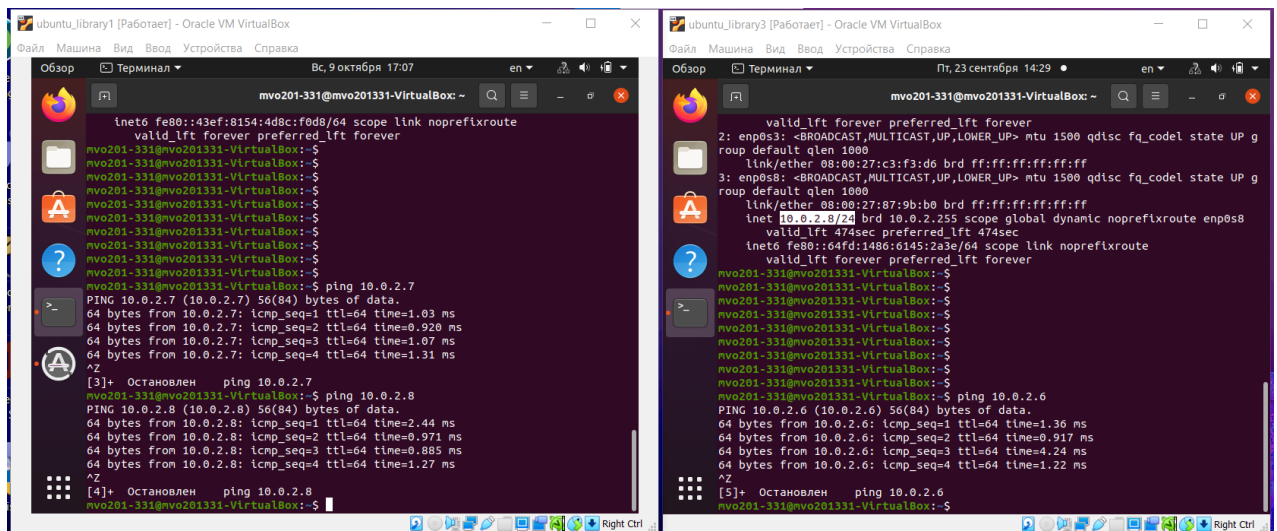
IISR



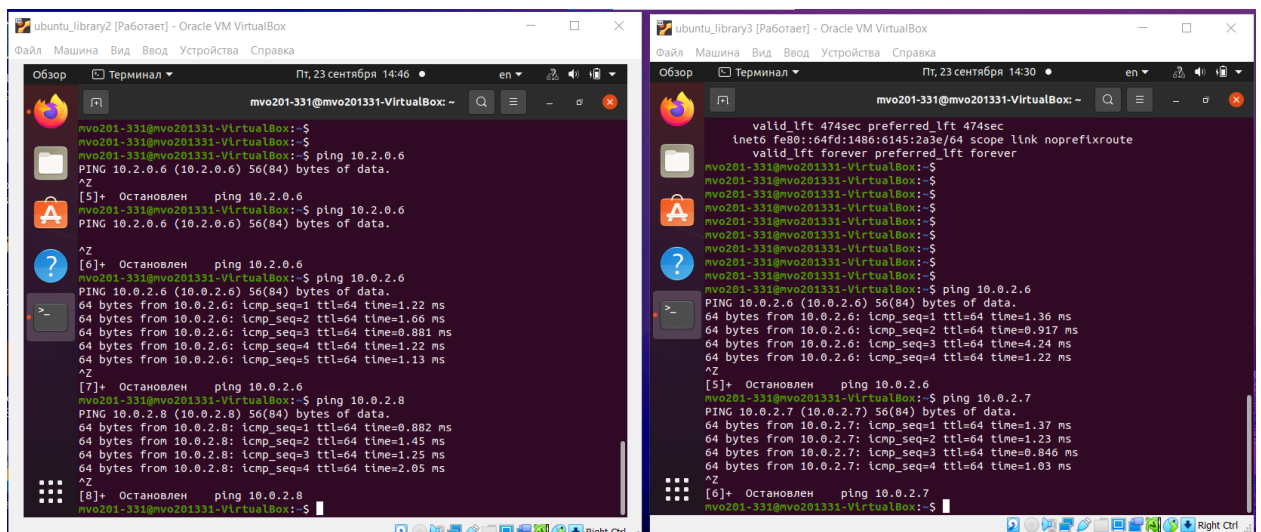
Пинг между 1 и 2 машинами



Пинг между 1 и 3 машинами



Ping между 2 и 3 машинами



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

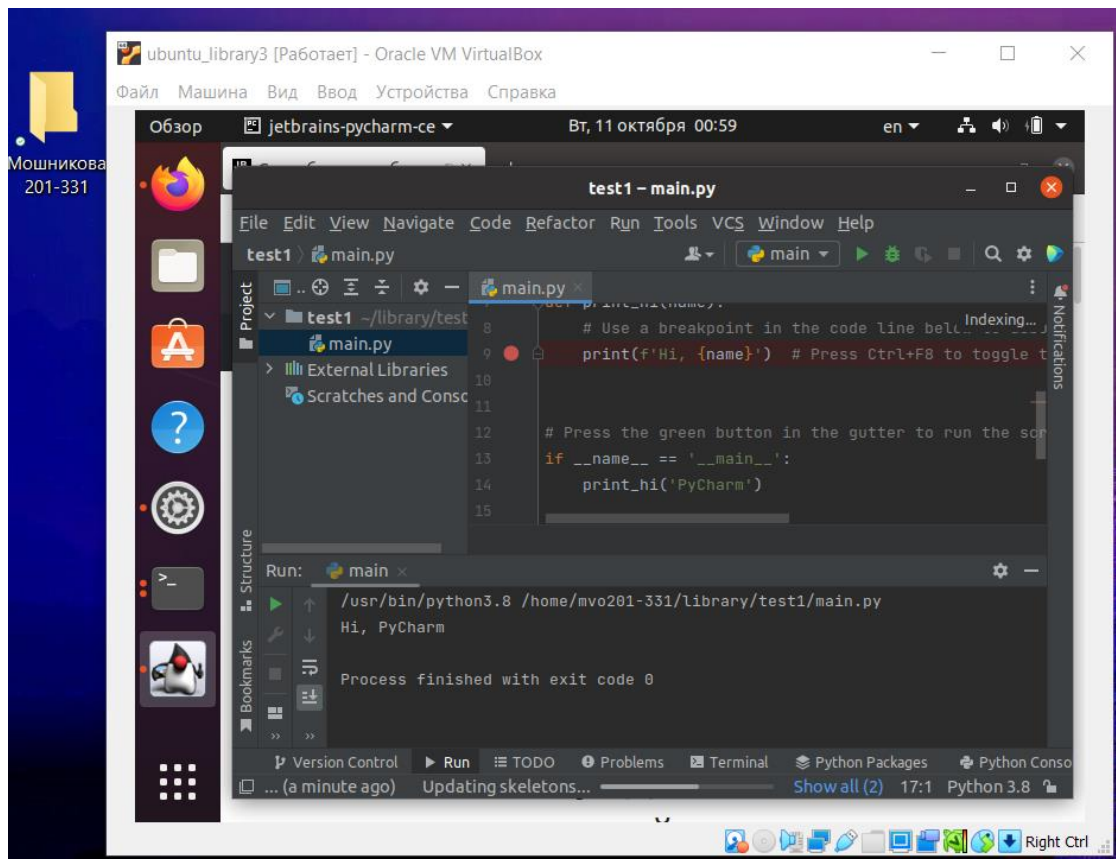
Тестовой машиной является **ubuntu_library3**

```
ubuntu_library3 [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Обзор Терминал Вс, 9 октября 17:51 en
mvo201-331@mvo201331-VirtualBox: ~
mvo201-331@mvo201331-VirtualBox:~$ sudo apt install git
[sudo] пароль для mvo201-331:
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Будут установлены следующие дополнительные пакеты:
git-man liberror-perl
Предлагаемые пакеты:
git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
gitweb git-cvs git-mediawiki git-svn
Следующие НОВЫЕ пакеты будут установлены:
git git-man liberror-perl
Обновлено 0 пакетов, установлено 3 новых пакетов, для удаления отмечено 0 пакет
ов, и 337 пакетов не обновлено.
Необходимо скачать 0 B/5 470 kB архивов.
После данной операции объём занятого дискового пространства возрастёт на 38,5 М
В.
```

```
Обзор Терминал Вт, 11 октября 01:08 en
mvo201-331@mvo201331-VirtualBox: ~
mvo201-331@mvo201331-VirtualBox:~$ git config --global user.name "stand1"
mvo201-331@mvo201331-VirtualBox:~$ git config --global user.email "stand1@mail.
ru"
mvo201-331@mvo201331-VirtualBox:~$ git config --list
user.name=stand1
user.email=stand1@mail.ru
mvo201-331@mvo201331-VirtualBox:~$
```

```
mvo201-331@mvo201331-VirtualBox:~$ git clone https://github.com/Dybashka/Librar
y-MB.git
Клонирование в «Library-MB»...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Распаковка объектов: 100% (6/6), 47.21 КиБ | 863.00 КиБ/с, готово.
mvo201-331@mvo201331-VirtualBox:~$
```

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



Вопросы:

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

Протокол NAT позволяет гостевой операционной системе выходить в Интернет, используя при этом частный IP, который не доступен со стороны внешней сети или же для всех машин локальной физической сети. Однако извне невозможно напрямую соединиться с такой системой, если она использует NAT.

В соединении типа "Сетевой мост" 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. Адаптер в режиме "Сетевой мост" подключается, минуя хост, к устройству, которое распределяет IP-адреса внутри локальной сети для всех физических сетевых карт.

При подключении типа "Виртуальный адаптер хоста" гостевые ОС могут взаимодействовать между собой, а также с хостом. Но все это только внутри самой виртуальной машины VirtualBox. В этом режиме адаптер хоста использует свое собственное, специально для этого предназначенное устройство, которое

называется vboxnet0. Также им 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. Невозможен прямой доступ извне к машинам.

Если необходимо настроить взаимосвязь между несколькими гостевыми операционными системами, работающими на одном хосте и могущими общаться только между собой, тогда можно воспользоваться режимом "Внутренняя сеть". Этот режим схож с режимом "Сетевой мост", но обладает большей безопасностью. В режиме "Сетевой мост" все пакеты отправляются и получаются через адаптер физической сети, установленный на машине-хосте. В этом случае весь трафик может быть перехвачен.

2. Каким образом осуществляется обработка системных вызовов Unix в виртуальных машинах при их развертке в ОС Windows?

Системный вызов является механизмом, который обеспечивает интерфейс между процессом и операционной системой. Это программный метод, при котором компьютерная программа запрашивает сервис у ядра ОС. Системный вызов предлагает услуги операционной системы пользовательским программам через API (интерфейс прикладного программирования). Системные вызовы являются единственными точками входа в систему ядра.

Системные вызовы нужны для:

1. Чтение и запись из файлов требуют системных вызовов.
2. Если файловая система хочет создать или удалить файлы, требуются системные вызовы.
3. Системные вызовы используются для создания и управления новыми процессами.
4. Сетевые подключения требуют системных вызовов для отправки и получения пакетов.

Есть варианты работы гипервизора, при получении системного вызова:

1. Гипервизор перехватывает системные вызовы от гостя: Гипервизор проверяет, поступил ли системный вызов от самой гостевой ОС к хосту или от программы в гостевой ОС к ней самой. Если это первый случай, то гипервизор фактически перенаправит вызов на оборудование, хотя и с помощью инструкций

виртуализации. Если это последнее, гипервизор перенаправит вызов на гостевую ОС, а затем продолжит.

2. Паравиртуализация: здесь сама гостевая ОС модифицируется таким образом, что вместо выполнения вызовов к оборудованию у нее есть API для вызова гипервизора для выполнения аппаратного ввода-вывода.

3. Двоичная трансляция: здесь гипервизор проверяет код гостевой ОС в так называемых «базовых блоках», сканируя привилегированные инструкции. Везде, где он их находит, он заменяет их вызовами своих собственных процедур на системные вызовы. Затем он приступает к кэшированию этих блоков и, в конечном итоге, создает целый набор таких блоков.

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

Файл образа диска виртуальной машины находится на хост-системе и воспринимается гостевой системой, как жёсткий диск определённой геометрии. Когда гостевая ОС читает с диска или записывает на него, гипервизор перенаправляет запрос в файл образа. Как и физический диск, виртуальный носитель имеет размер и ёмкость, которые необходимо указать при создании диска. Только в отличие от физического носителя его можно расширять.

Все настройки виртуальной машины хранятся в файлах с расширением .vmx в папке, заданной пользователем, а файлы виртуальных дисков имеют расширение .vmdk и также хранятся в этой папке.