

Лабораторная работа № 1. Введение в Mininet

1.1. Цель работы

Основной целью работы является развёртывание в системе виртуализации (например, в VirtualBox) mininet, знакомство с основными командами для работы с Mininet через командную строку и через графический интерфейс.

1.2. Предварительные сведения

Mininet (<http://mininet.org/>) — это виртуальная среда, которая позволяет разрабатывать и тестировать сетевые инструменты и протоколы. В сетях Mininet работают реальные сетевые приложения Unix/Linux, а также реальное ядро Linux и сетевой стек.

1.3. Последовательность выполнения работы

1.3.1. Настройка стенда виртуальной машины *Mininet*

1.3.1.1. Общая информация

- Репозиторий *Mininet*: <https://github.com/mininet/mininet>
- Рекомендуемый к установке образ виртуальной машины: mininet-2.3.0-210211-ubuntu-20.04.1-legacy-server-amd64-ovf
- В случае выполнения лабораторных работ в дисплейных классах РУДН в качестве системы виртуализации следует использовать VirtualBox с указанием путей и размещением файлов виртуальной машины в каталоге `/var/tmp` в предварительно созданном подкаталоге, совпадающим по названию с вашей учётной записью.

1.3.1.2. Настройка образа VirtualBox

- Перейдите в репозиторий *Mininet*. Скачайте актуальный релиз ovf-образа виртуальной машины.
- При необходимости переместите скачанный образ в каталог для работы, затем распакуйте его.
- Запустите систему виртуализации и импортируйте файл `.ovf`.
- Перейдите в настройки системы виртуализации и уточните параметры настройки виртуальной машины. В частности, для VirtualBox выберите импортированную виртуальную машину и перейдите в меню Машина » Настроить. Перейдите к опции «Система». Если внизу этого окна есть сообщение об обнаружении неправильных настроек, то, следуя рекомендациям, внесите исправления. Например, может потребоваться увеличить видеопамять

виртуальной машины и изменить тип графического контроллера на рекомендуемый. В настройках сети первый адаптер должен иметь подключение типа NAT. Для второго адаптера укажите тип подключения *host-only network adapter* (виртуальный адаптер хоста), который в дальнейшем вы будете использовать для входа в образ виртуальной машины. В этом режиме адаптер хоста использует специальное устройство `vboxnet0`, создает подсеть и назначает IP-адрес сетевой карте гостевой операционной системы. Если данный режим не получается выбрать в настройках, то воспользуйтесь менеджером сетей хоста в VirtualBox для создания сети и настройки адаптера.

- Запустите виртуальную машину с Mininet.

1.3.1.3. Подключение к виртуальной машине

- Залогиньтесь в виртуальной машине:
 - login: mininet
 - password: mininet
- Посмотрите адрес машины:

```
1 ifconfig
```

- Внутренний адрес машины будет иметь вид 192.168.x.y.
- Подключитесь к виртуальной машине (из терминала хостовой машины):

```
1 ssh -Y mininet@192.168.x.y
```

- Для отключения ssh-соединения с виртуальной машиной нажмите `Ctrl` + `d`.
- Настройте ssh-подсоединение по ключу к виртуальной машине, для чего в терминале основной Linux-машины перейдите в каталог `.ssh` своего домашнего каталога и введите (вместо 192.168.x.y укажите внутренний адрес виртуальной машины Mininet):

```
1 ssh-copy-id mininet@192.168.x.y
```

- Вновь подключитесь к виртуальной машине и убедитесь, что подсоединение происходит успешно и без ввода пароля.

1.3.1.4. Настройка доступа к Интернет

- После подключения к виртуальной машине mininet посмотрите IP-адреса машины:

```
1 ifconfig
```

- Для доступа к сети Интернет должен быть активен адрес NAT: 10.0.0.x.

- Если активен только внутренний адрес машины вида `192.168.x.y`, то активируйте второй интерфейс, набрав в командной строке:

```
1 sudo dhclient eth1
2 ifconfig
```

- Для удобства дальнейшей работы установите `mc`:

```
1 sudo apt install mc
```

- Для удобства дальнейшей работы добавьте для `mininet` указание на использование двух адаптеров при запуске. Для этого требуется перейти в режим суперпользователя и внести изменения в файл `/etc/netplan/01-netcfg.yaml` виртуальной машины `mininet`:

```
1 sudo mcedit /etc/netplan/01-netcfg.yaml
```

- В результате файл `/etc/netplan/01-netcfg.yaml` должен иметь следующий вид:

```
1 network:
2   version: 2
3   renderer: networkd
4   ethernet:
5     eth0:
6       dhcp4: yes
7     eth1:
8       dhcp4: yes
```

1.3.1.5. Обновление версии *Mininet*

- В виртуальной машине `mininet` переименуйте предыдущую установку *Mininet*:

```
1 mv ~/mininet ~/mininet.orig
```

- Скачайте новую версию *Mininet*:

```
1 cd ~
2 git clone https://github.com/mininet/mininet.git
```

- Обновите исполняемые файлы:

```
1 cd ~/mininet
2 sudo make install
```

- Проверьте номер установленной версии mininet:

```
1 mn --version
```

1.3.1.6. Настройка параметров *XTerm*

По умолчанию *XTerm* использует растровые шрифты малого кегля. Для увеличения размера шрифта и применения векторных шрифтов вместо растровых необходимо внести изменения в файл `/etc/X11/app-defaults/XTerm`. Для этого можно воспользоваться следующей командой:

```
1 sudo mcedit /etc/X11/app-defaults/XTerm
```

и затем в конце файла добавить строки:

```
1 xterm*faceName: Monospace
2 xterm*faceSize: 12
```

Здесь выбран системный моноширинный шрифт, кегль шрифта — 12 пунктов.

1.3.1.7. Настройка соединения X11 для суперпользователя

При попытке запуска приложения из-под суперпользователя возникает ошибка:

X11 connection rejected because of wrong authentication.

Ошибка возникает из-за того, что X-соединение выполняется от имени пользователя mininet, а приложение запускается от имени пользователя root с использованием sudo. Для исправления этой ситуации необходимо заполнить файл полномочий `/root/.Xauthority`, используя утилиту xauth.

Скопируйте значение куки (MIT magic cookie)¹ пользователя mininet в файл для пользователя root:

```
1 mininet@mininet-vm:~$ xauth list $DISPLAY
2 mininet-vm/unix:10 MIT-MAGIC-COOKIE-1
   ↪ 295acad8e35d17636924c5ab80e8462d
3
4 mininet@mininet-vm:~$ sudo -i
5 mininet@mininet-vm:~$ xauth list
6 xauth: file /root/.Xauthority does not exist
```

¹Значение для *MIT-MAGIC-COOKIE* приведено условно.

```
7 root@mininet-vm:~# xauth add mininet-vm/unix:10
   ↪ MIT-MAGIC-COOKIE-1 295acad8e35d17636924c5ab80e8462d
8 root@mininet-vm:~# xauth list $DISPLAY
9 mininet-vm/unix:10 MIT-MAGIC-COOKIE-1
   ↪ 295acad8e35d17636924c5ab80e8462d
10 root@mininet-vm:~# logout
```

После выполнения этих действий графические приложения должны запускаться под пользователем mininet.

1.3.1.8. Работа с Mininet из-под Windows

Варианты Xserver. Использовать можно любой вариант Xserver.

- VcXsrv Windows X Server:
 - Сайт: <https://sourceforge.net/projects/vcxsrv/>
 - Лицензия: GPLv3.
 - Компиляция: Visual C++ 2012 Express Edition
- Xming X Server:
 - Сайт: <http://www.straightrunning.com/XmingNotes/>
 - Лицензия:
 - *Public Domain Release*: MIT
 - *Website Release*: за загрузку новых выпусков необходимо вносить плату; приобретение лицензии дает пользователю доступ к новым загрузкам в течение одного года.
 - Компиляция: MinGW, Pthreads-Win32.

Установка программного обеспечения.

- Установите putty:

```
1 choco install putty
```

- Установите VcXsrv Windows X Server:

```
1 choco install vcxsrv
```

Запуск Xserver.

- Запустите XLaunch.
- Выберите опции:
 - *Multiple windows*;
 - *Display number*: -1;
 - *Start no client*.
- Можно сохранить параметры, тогда при следующем запуске не нужно будет отмечать эти опции.

Запуск putty. При подключении добавьте опцию перенаправления X11:

- `Connection >> SSH >> X11` : *Enable X11 forwarding.*

1.3.2. Основы работы в Mininet

1.3.2.1. Работа с Mininet с помощью командной строки

1. Вызов Mininet с использованием топологии по умолчанию.

- Для запуска минимальной топологии введите в командной строке:

```
1 sudo mn
```

Эта команда запускает Mininet с минимальной топологией, состоящей из коммутатора, подключённого к двум хостам.

- Для отображения списка команд интерфейса командной строки Mininet и примеров их использования введите команду в интерфейсе командной строки Mininet:

```
1 help
```

- Для отображения доступных узлов введите:

```
1 nodes
```

Вывод этой команды показывает, что есть два хоста (хост h1 и хост h2) и коммутатор (s1).

- Иногда бывает полезно отобразить связи между устройствами в Mininet, чтобы понять топологию. Введите команду `net` в интерфейсе командной строки Mininet, чтобы просмотреть доступные линки:

```
1 net
```

Вывод этой команды показывает:

- Хост h1 подключён через свой сетевой интерфейс h1-eth0 к коммутатору на интерфейсе s1-eth1.
- Хост h2 подключён через свой сетевой интерфейс h2-eth0 к коммутатору на интерфейсе s1-eth2.
- Коммутатор s1:
 - имеет петлевой интерфейс lo.
 - подключается к h1-eth0 через интерфейс s1-eth1.
 - подключается к h2-eth0 через интерфейс s1-eth2.
- Mininet позволяет выполнять команды на конкретном устройстве. Чтобы выполнить команду для определенного узла, необходимо сначала указать устройство, а затем команду, например:

```
1 h1 ifconfig
```

Эта запись выполняет команду `ifconfig` на хосте `h1` и показывает интерфейсы хоста `h1` — хост `h1` имеет интерфейс `h1-eth0`, настроенный с IP-адресом `10.0.0.1`, и другой интерфейс `lo`, настроенный с IP-адресом `127.0.0.1`.

- Посмотрите конфигурацию всех узлов. В отчёте укажите, какие адреса присвоены интерфейсам устройств.
2. Проверка связности.
- По умолчанию узлам `h1` и `h2` назначаются IP-адреса `10.0.0.1/8` и `10.0.0.2/8` соответственно. Чтобы проверить связь между ними, вы можете использовать команду `ping`. Команда `ping` работает, отправляя сообщения эхо-запроса протокола управляющих сообщений Интернета (ICMP) на удалённый компьютер и ожидая ответа.
- Например, команда

```
1 h1 ping 10.0.0.2
```

проверяет соединение между хостами `h1` и `h2`. Для остановки теста нажмите `Ctrl`+`C`.

3. Остановка эмуляции.

Для завершения работы режима эмуляции mininet используйте команду

```
1 exit
```

Заметим, что команда `sudo mn` —с часто используется в терминале для очистки предыдущего экземпляра Mininet (например, после сбоя).

1.3.2.2. Построение и эмуляция сети в Mininet с использованием графического интерфейса

1. Построение топологии сети.

– В терминале виртуальной машины mininet запустите *MiniEdit*:



```
1 sudo ~/mininet/mininet/examples/miniedit.py
```

Основные кнопки:

- **Select**: позволяет выбирать/перемещать устройства. Нажатие `Del` на клавиатуре после выбора устройства удаляет его из топологии.
- **Host**: позволяет добавить новый хост в топологию. После нажатия этой кнопки щёлкните в любом месте пустого холста, чтобы вставить новый хост.
- **Switch**: позволяет добавить в топологию новый коммутатор. После нажатия этой кнопки щёлкните в любом месте пустого холста, чтобы вставить переключатель.



- **Link**: соединяет устройства в топологии. После нажатия этой кнопки щелкните устройство и перетащите его на второе устройство, с которым необходимо установить связь.
- **Run**: запускает эмуляцию. После проектирования и настройки топологии нажмите кнопку запуска.
- **Stop**: останавливает эмуляцию.
- Добавьте два хоста и один коммутатор, соедините хосты с коммутатором.
- Настройте IP-адреса на хостах h1 и h2. Для этого удерживая правую кнопку мыши на устройстве выберите свойства. Для хоста h1 укажите IP-адрес 10.0.0.1/8, а для хоста h2 — 10.0.0.2/8.

2. Проверка связности.

- Перед проверкой соединения между хостом h1 и хостом h2 необходимо запустить эмуляцию. Для запуска эмуляции нажмите кнопку *Run*. После начала эмуляции кнопки панели MiniEdit станут серыми, указывая на то, что в настоящее время они отключены.
- Откройте терминал на хосте h1, удерживая правую кнопку мыши на хосте h1 и выбрав *Terminal*. Это действие позволит выполнять команды на хосте h1.
- Откройте терминал на хосте h2.
- На терминале хоста h1 введите команду `ifconfig`, чтобы отобразить назначенные ему IP-адреса. Интерфейс h1-eth0 на хосте h1 должен быть настроен с IP-адресом 10.0.0.1 и маской подсети 255.0.0.0.
- Повторите эти действия на хосте h2. Его интерфейс h2-eth0 должен быть настроен с IP-адресом 10.0.0.2 и маской подсети 255.0.0.0.
- Проверьте соединение между хостами, введя в терминале хоста h1 команду `ping 10.0.0.2`. Для остановки теста нажмите  + .
- Остановите эмуляцию, нажав кнопку *Stop*.

3. Автоматическое назначение IP-адресов.

Ранее IP-адреса узлам h1 и h2 были назначены вручную. В качестве альтернативы можно полагаться на Mininet для автоматического назначения IP-адресов.



- Удалите назначенный вручную IP-адрес с хостов h1 и h2.
- В *MiniEdit* нажмите  . По умолчанию в поле базовые значения IP-адресов (IP Base) установлено 10.0.0.0/8. Измените это значение на 15.0.0.0/8.
- Запустите эмуляцию, нажав кнопку *Run*.
- Откройте терминал на хосте h1, удерживая правую кнопку мыши на хосте h1 и выбрав *Terminal*.
- Чтобы отобразить IP-адреса, назначенные хосту h1, введите команду

```
1  ifconfig
```



Интерфейс h1-eth0 на узле h1 теперь имеет IP-адрес 15.0.0.1 и маску подсети 255.0.0.0.

- Вы также можете проверить IP-адрес, назначенный хосту h2. Соответствующий интерфейс h2-eth0 на хосте h2 должен иметь IP-адрес 15.0.0.2 и маску подсети 255.0.0.0.
 - Остановите эмуляцию, нажав кнопку *Stop*.
4. Сохранение и загрузка топологии Mininet.
- В домашнем каталоге виртуальной машины mininet создайте каталог для работы с проектами mininet:

```
1 mkdir ~/work
```

- Для сохранения топологии сети в файл нажмите в MiniEdit  .
- После сохранения проекта поменяйте права доступа к файлам в каталоге проекта:

```
1 sudo chown -R mininet:mininet ~/work
```

- Для загрузки топологии в MiniEdit нажмите  .
5. Завершите соединение с виртуальной машиной mininet и выключите её.

1.4. Содержание отчёта

1. Титульный лист с указанием номера лабораторной работы и ФИО студента.
2. Формулировка задания работы.
3. Описание результатов выполнения задания:
 - скриншоты (снимки экрана), фиксирующие выполнение работы;
 - подробное описание настроек служб в соответствии с заданием;
 - полные тексты конфигурационных файлов настраиваемых в работе служб;
 - результаты проверки корректности настроек служб в соответствии с заданием (подтверждённые скриншотами).
4. Выводы, согласованные с заданием работы.