

Лабораторная работа №5

Конфигурирование сетевых интерфейсов

1. Теоретические сведения

NET-Simulator позволяет строить виртуальные вычислительные сети из виртуальных сетевых устройств: маршрутизаторов, настольных компьютеров, концентраторов и т.п. Устройствами можно управлять при помощи интерфейса командной строки из виртуальных терминалов.

Сетевое ядро

В NET-Simulator реализованы только два уровня ISO OSI: канальный и сетевой. Таким образом NET-Simulator позволяет решать следующие образовательные задачи:

1. Изучение принципов работы коммутаторов второго и третьего уровня, пассивных концентраторов.
2. Отработка практических навыков статической маршрутизации в IP-сетях.
3. Изучение принципов работы протоколов канального уровня, ARP, IP4, ICMP.
4. Отработка практических навыков поисков неисправностей в IP-сетях.

Физическая природа сети не учитывается. Предполагается, что пакеты канального уровня распространяются в среде аналогичной локальной сети на основе Ethernet.

На канальном уровне используется простейший Ethernet-образный протокол, который предусматривает адресацию по 6-ти байтовым MAC-адресам. Уникальность MAC-адресов обеспечивает ядро NET-Simulator. Пакет канального протокола представляет собой объект Java и не имеет аналогов в реальных сетях.

На сетевом уровне используется ограниченная реализация IP в соответствии с RFC791. Для преобразования IP-адресов в MAC реализована служба ARP на основе широковещательных запросов.

Для работы служебных утилит, таких как ping, используется ограниченная реализация ICMP в соответствии с RFC792.

Графический интерфейс

В главном окне NET-Simulator отображается поле в которое можно добавлять различные сетевые устройства из меню Устройство.

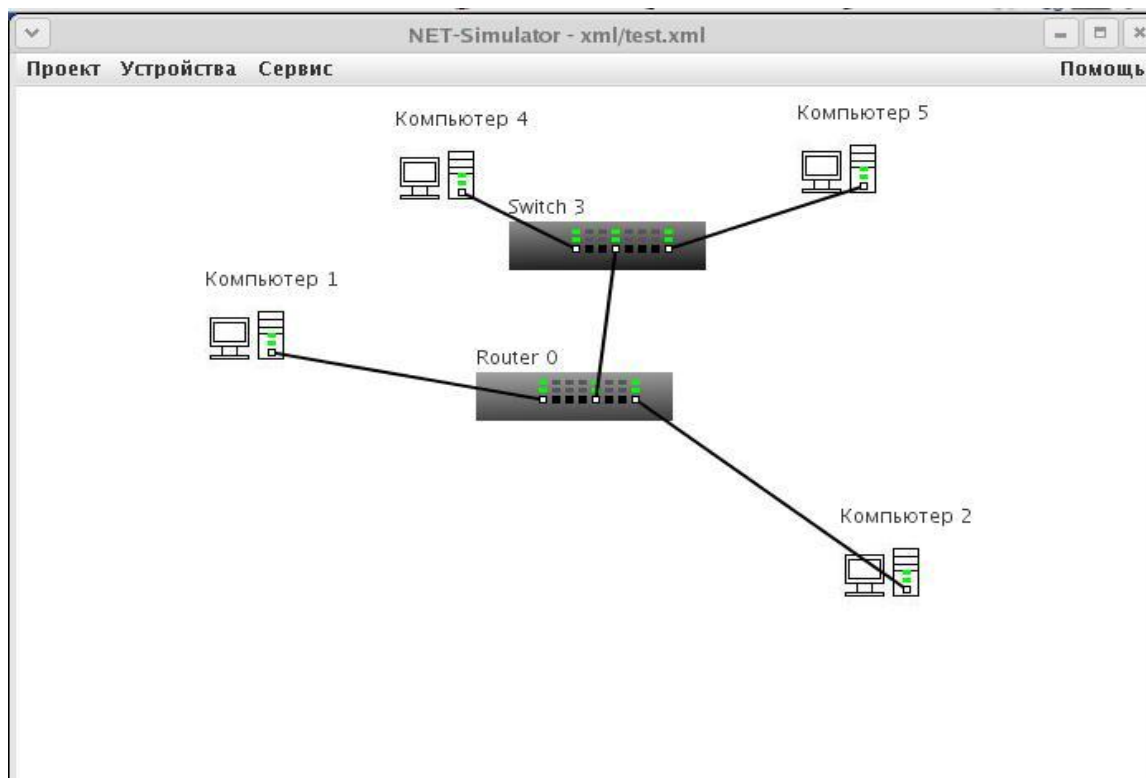


Рис. 1 Внешний вид главного окна симулятора

```

Welcome to terminal of the NET-Simulator virtual router!

Type help for list of commands. To get help for the command
type <command> -h.
Press Ctrl+L to refresh the screen.

64 bytes from 12.0.0.11: icmp_seq=7 ttl=62 time=13 ms
=>
=>
=>route
IP routing table
  Destination      Gateway         Netmask         Flags Metric Iface
  10.0.0.0          *              255.0.0.0       U      1      eth0
  11.0.0.0          10.0.0.11     255.0.0.0       UG     0      eth0
  12.0.0.0          10.0.0.11     255.255.255.0  UG     0      eth0
=>ping 12.0.0.11
PING 12.0.0.11
64 bytes from 12.0.0.11: icmp_seq=0 ttl=62 time=460 ms
64 bytes from 12.0.0.11: icmp_seq=1 ttl=62 time=9 ms
64 bytes from 12.0.0.11: icmp_seq=2 ttl=62 time=17 ms
64 bytes from 12.0.0.11: icmp_seq=3 ttl=62 time=30 ms
64 bytes from 12.0.0.11: icmp_seq=4 ttl=62 time=8 ms
=>
  
```

Рис. 2 Внешний вид окна настройки компьютера

Поддерживаются следующие типы устройств:

1. Маршрутизатор. Коммутатор 3-го уровня с 8-мью интерфейсами и поддержкой IP4.

2. Настольный компьютер. Фактически маршрутизатор с одним интерфейсом.

3. Концентратор (Hub). Простейшее устройства ретранслирующее пакеты канального уровня на свои интерфейсы. Не имеет терминала и соответственно никак не управляется.

4. Коммутатор (Switch). Коммутатор 2-го уровня с 8-мью интерфейсами. Коммутирует пакеты канального уровня на основе таблиц MAC-адресов, по аналогии с известными алгоритмами используемыми в Ethernet-свитчах.

Устройства соединяются с помощью универсальной среды передачи данных, виртуального патчкорда. При прохождении пакета через патчкорд, он подсвечивается для визуального отслеживания активности в сети.

Вновь добавленные устройств появляются в верхнем левом углу, после чего их можно перетаскивать мышкой в удобное место. Вилки патчкордов «приклеиваются» к розеткам интерфейсов устройств. Нажатие правой кнопки мыши на устройстве открывает контекстное меню, которое позволяет просмотреть свойства, открыть терминал или удалить устройство. Двойной щелчок левой кнопкой мыши открывает терминал.

Сохранение/загрузка проектов

Проекты сохраняются в формате xml. DTD для проектов NET-Simulator находится в каталоге dtd — net_simulator.dtd

Виртуальные терминалы и интерфейс командной строки.

Виртуальные устройства в NET-Simulator управляются при помощи интерфейса командной строки из виртуальных терминалов. Терминал устройства можно открыть двойным кликом на значке устройства или через контекстное меню. Поддерживается история команд, клавиши вверх/вниз позволяют просматривать историю команд.

Список команд доступных на данном устройстве можно посмотреть командой help. Сочетание клавиш Ctrl+L очищает терминал. Краткая справка по любой команде выводится при вызове команды с опцией -h.

Справочник команд:

help

выводит список доступных команд.

help [-h]

Опции	Описание
-h	Краткая справка.

route

позволяет управлять таблицей маршрутизации устройств поддерживающих протокол IP4.

route [-h] [{-add|-del} <target> [-netmask <address>] [-gw <address>] [-metric <M>] [-dev <If>]]

Опции	Описание
-h	Краткая справка.
target	Адрес назначения. Назначением может быть подсеть или отдельный узел в зависимости от значения маски подсети. Если маска равна 255.255.255.255 или отсутствует совсем назначением будет узел, иначе назначением будет сеть.
-add	Добавляет новый маршрут в таблицу маршрутизации.
-del	Удаляет маршрут из таблицы маршрутизации.
-dev <If>	Принудительно присоединяет маршрут к определенному интерфейсу. If — имя интерфейса.
-gw <address>	Направляет пакеты по этому маршруту через заданный шлюз. address — адрес шлюза.
-netmask <address>	Маска подсети используемая совместно с адресом назначения при добавлении маршрута. address — маска. Если маска не задана явно подразумевается 255.255.255.255.
-metric <M>	Метрика используемая в данном маршруте. M — целое число большее или равное нулю.

Если route вызывается без параметров, то команда выводит на экран таблицу маршрутизации:

```
=>route
IP routing table
Destination      Gateway          Netmask          Flags    Metric  Iface
10.0.0.0          *                255.0.0.0        U         1       eth0
11.0.0.0          10.0.0.10       255.0.0.0        UG         1       eth0
192.168.120.1     10.0.0.10       255.255.255.255 UGH         1       eth0
```

Если маршрут не использует шлюз, вместо адреса шлюза выводиться *. Flags может содержать значение: U — маршрут активен, G — маршрут использует шлюз, H — назначением является узел.

Примеры:

```
=>route -add 192.168.120.0 -netmask 255.255.255.0 -dev eth0
=>route
IP routing table
Destination      Gateway          Netmask          Flags    Metric  Iface
192.168.120.0     *                255.255.255.0    U         1       eth0
=>
=>route -add 192.168.121.10 -gw 192.168.120.10
=>route
IP routing table
Destination      Gateway          Netmask          Flags    Metric  Iface
192.168.120.0     *                255.255.255.0    U         1       eth0
192.168.121.10    192.168.120.1    255.255.255.255  UGH         1       eth0
=>
```

ifconfig

конфигурирует сетевые интерфейсы.

ifconfig [-h] [-a] [<interface>] [<address>] [-broadcast <address>] [-netmask <address>] [-up|-down]

Опции	Описание
-h	Краткая справка.
-a	Показывать информацию о всех интерфейсах. Если данная опция отсутствует выводится информация только об активных интерфейсах.
interface	Конфигурировать или показать информацию только о заданном интерфейсе.
address	IP-адрес присваиваемый интерфейсу.
- broadcast <address>	Широковещательный адрес присваиваемый интерфейсу. address — широковещательный адрес.
-netmask <address>	Маска подсети используемая совместно с адресом. address — маска. Если маска не задана явно, маска принимается равной стандартным значения для стандартных классов подсетей А, В и С.
-up	Активирует интерфейс. При активизации интерфейса для него автоматически добавляется соответствующий маршрут в таблице маршрутизации.
-down	Деактивирует интерфейс. При деактивации интерфейса соответствующий маршрут автоматически удаляется из таблицы маршрутизации.

Если ifconfig вызывается без параметров, то команда выводит на экран данные о состоянии всех активных интерфейсов:

```
=>ifconfig
eth0      Link encap:Ethernet  HWaddr 0:0:0:0:CF:0
          inet addr:192.168.120.1  Bcast:192.168.120.255
Mask:255.255.255.0
          UP
          RX packets:23 errors:0 dropped:0
          TX packets:23 errors:0 dropped:0
          RX bytes:0 TX bytes:0
```

HWaddr — уникальный 6-ти байтовый адрес интерфейса, аналогичный MAC-адресу в Ethernet сетях. Назначается автоматически.

Примеры:

```
=>ifconfig eth0 192.168.120.1 -up
=>ifconfig
eth0      Link encap:Ethernet  HWaddr 0:0:0:0:CF:0
          inet addr:192.168.120.1  Bcast:192.168.120.255
Mask:255.255.255.0
```

```

UP
RX packets:0 errors:0 dropped:0
TX packets:0 errors:0 dropped:0
RX bytes:0 TX bytes:0

```

ping

использует ICMP протокол что бы проверить достижимость интерфейса удаленного узла. ping посылает удаленному узлу ICMP ECHO_REQUEST и ожидает в течении определенного промежутка времени ICMP ECHO_RESPONSE. В случае получения ответа выводит данные о прохождении ICMP-пакета по сети.

ping [-h] [-i <interval>] [-t <ttd>] <destination>

Опции	Описание
-h	Краткая справка.
-i <interval>	Задаёт частоту ICMP-запросов. interval — интервал между запросами в секундах. По умолчанию отсылается один пакет в секунду.
-t <ttd>	Задаёт значение атрибута Time to Live в генерируемых IP-пакетах. ttl — целое число 0-255. По умолчанию TTL равно 64.
destination	IP-адрес исследуемого узла

Примеры:

```

=>ping 192.168.120.1
PING 192.168.120.1
64 bytes from 192.168.120.1: icmp_seq=0 ttl=62 time=477 ms
64 bytes from 192.168.120.1: icmp_seq=1 ttl=62 time=435 ms
64 bytes from 192.168.120.1: icmp_seq=2 ttl=62 time=234 ms
64 bytes from 192.168.120.1: icmp_seq=3 ttl=62 time=48 ms
64 bytes from 192.168.120.1: icmp_seq=4 ttl=62 time=87 ms
64 bytes from 192.168.120.1: icmp_seq=5 ttl=62 time=56 ms

```

ping выводит результат исследования удаленного узла в следующем формате: 64 bytes from 192.168.120.1 — размер полученного ответа и адрес источника ответа. В NET-Simulator размер пакета имеет условное значение и всегда равен 64В. icmp_seq=0 — номер пакета. Каждый запрос содержит свой номер, как правило формируется инкрементно. ping выводит номер пакета из каждого полученного ответа. ttl=62 — значение TTL из полученного ответа. time=48 ms — время прохождения пакетом полного маршрута (туда и обратно, round-trip time) в миллисекундах.

arp

показывает ARP-таблицу устройства. Кроме того опция -r позволяет сформировать запрос для определения MAC-адреса по явно заданному IP-адресу. Эта функция обычно отсутствует в реальных устройствах, в NET-Simulator она добавлена для наглядности при изучении протоколов канального и сетевого уровня.

arp [-h] [-r <IP-address> <interface>]

Опции	Описание
-h	Краткая справка.
-r <IP-address> <interface>	Прежде чем вывести ARP-таблицу предпринимает попытку найти MAC-адрес по явно заданному IP-адресу. IP-address IP-адрес для которого определяется MAC-адрес. interface имя интерфейса в сети подсоединенной к которому будет происходить поиск.

Если `arp` вызывается без параметров, то команда выводит на экран ARP-таблицу:

```
=>arp
Address          HWaddress      iface
10.0.0.10        0:0:0:0:BC:0   eth0
10.0.0.11        0:0:0:0:1F:2   eth0
```

Примеры:

```
=>arp -r 192.168.120.12 eth1
Address          HWaddress      iface
10.0.0.10        0:0:0:0:BC:0   eth0
10.0.0.11        0:0:0:0:1F:2   eth0
192.168.120.12   0:0:0:0:12:1   eth1
```

mactable

показывает таблицу MAC-адресов коммутаторов второго уровня.

`mactable [-h]`

Опции	Описание
-h	Краткая справка.

Примеры:

```
=>mactable
MACAddress      port
0:0:0:0:B3:0    0
0:0:0:0:2F:2    0
0:0:0:0:03:0    3
```

Где `port` — номер порта на коммутаторе. Нумерация портов идет по порядку начиная с нуля.

2. Задание на лабораторную работу.

Изучить команды операционной системы для управления и тестирования сетевых ресурсов. Выполнить задания лабораторной работы №4 с использованием командного режима (Net Simulator). Проверить спроектированную сеть.

Отчёт должен содержать перечень всех используемых команд с кратким описанием, последовательность выполненных команд с указанием полученного результата.

Результаты моделирования обязательно должны быть продемонстрированы на компьютере.

3. Контрольные вопросы

5. Возможности сетевого ядра программы NET-Simulator.
6. Типы устройств поддерживаемые программой NET-Simulator.
7. Команды программы NET-Simulator работы с сетью. Основные параметры.
8. Проверка наличия соединения с удалённым узлом.
9. Программные средства мониторинга и анализа использования сети в NET-Simulator.