Федеральное агентство связи

Санкт-Петербургский университет телекоммуникаций
им. проф. М. А. Бонч-Бруевича

Кафедра ОПДС

Владимиров С. С.

Методические указания к лабораторным работам по курсу

«ИНТЕРНЕТ-ТЕХНОЛОГИИ И МУЛЬТИМЕДИА»

Для всех специальностей

Санкт-Петербург
2014

Содержание

Часть I

Лабораторные работы

1 Анализ протоколов. Изучение структуры кадра.

1.1 Цель работы

Изучение протоколов стека TCP/IP сетевого и транспортного уровня. Анализ пакетов различных протоколов.

1.2 Материал для изучения и ознакомления

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

1. Разделы 7, 8, 9 данного пособия.
2. IP — Википедия // `http://ru.wikipedia.org/wiki/IP`
3. IPv4 — Википедия // `http://ru.wikipedia.org/wiki/IPv4`
4. TCP — Википедия // `http://ru.wikipedia.org/wiki/TCP`
5. UDP — Википедия // `http://ru.wikipedia.org/wiki/UDP`
6. DNS — Википедия // `http://ru.wikipedia.org/wiki/DNS`
7. RTP — Википедия // `http://ru.wikipedia.org/wiki/RTP`

1.3 Задание на лабораторную работу

Используя анализатор протоколов `wireshark` отловить по несколько разных пакетов для каждого из протоколов транспортного уровня (TCP, UDP, RTP). Изучить структуру пойманных пакетов.

1.4 Порядок выполнения лабораторной работы

1.4.1 Изучение протокола TCP

1. Захватить пакеты, посылаемые при установлении соединения в начале сеанса TCP (процедура “тройного рукопожатия”).
2. Построить для данных пакетов диаграмму Flow Graph, иллюстрирующую процедуру “тройного рукопожатия”.
3. Построенную диаграмму и захваченные пакеты сохранить для отчета. (Пакеты в отчете должны быть представлены в полностью развернутом виде, включая шестнадцатиричное представление.)
4. Захватить пакеты, посылаемые при завершении соединения в конце сеанса TCP.
5. Построить для данных пакетов диаграмму Flow Graph.
6. Построенную диаграмму и захваченные пакеты сохранить для отчета. (Пакеты в отчете должны быть представлены в полностью развернутом виде, включая шестнадцатиричное представление.)

1.4.2 Изучение протоколов UDP и DNS

1. Захватить пакеты, иллюстрирующие получение IP-адреса по доменному имени.
2. Построить для захваченных пакетов диаграмму Flow Graph.
3. Построенную диаграмму и захваченные пакеты сохранить для отчета. (Пакеты в отчете должны быть представлены в полностью развернутом виде, включая шестнадцатиричное представление.)

1.5 Содержание отчета

1. Титульный лист согласно приложению.
2. Цель работы.
3. Диаграмма и захваченные пакеты согласно пункту 3 подраздела 1.4.1.
4. Диаграмма и захваченные пакеты согласно пункту 6 подраздела 1.4.1.
5. Диаграмма и захваченные пакеты согласно пункту 3 подраздела 1.4.2.

1.6 Контрольные вопросы

1. Протокол IP. [2, 3]
2. Протокол TCP. [4]
3. Протокол UDP. [5]
4. Протокол DNS. [6]
5. Протокол RTP. [7]

2 Вычисление контрольной суммы в IP-пакетах

2.1 Цель работы

Изучить структуру IP-пакета, TCP-сегмента и UDP-датаграммы. Получить практические навыки в вычислении контрольной суммы заголовка IP-пакета.

2.2 Материал для изучения и ознакомления

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

- RFC 791 — Протокол IP // `http://rfc2.ru/791.rfc`
- RFC 793 — Протокол управления передачей (TCP) // `http://rfc2.ru/793.rfc`
- RFC 768 — Протокол датаграмм клиента (UDP) // `http://rfc2.ru/768.rfc`
- RFC 1071 — Расчет контрольных сумм в Internet // `http://rfc2.ru/1071.rfc`
- IP — Википедия // `http://ru.wikipedia.org/wiki/IP`
- TCP — Википедия // `http://ru.wikipedia.org/wiki/TCP`
- UDP — Википедия // `http://ru.wikipedia.org/wiki/UDP`

2.3 Порядок выполнения лабораторной работы

1. Получить у преподавателя бланк с индивидуальным заданием.
2. Согласно материалам из рекомендованных источников "разбить" заданный кадр на поля, как показано ниже в примере.
3. Рассчитать контрольную сумму заголовка IP-пакета. Вписать результат в соответствующее поле на бланке задания. Процесс расчета привести на бланке с заданием.
4. Рассчитать контрольную сумму TCP-сегмента/UDP-датаграммы (в зависимости от выданного задания). Вписать результат в соответствующее поле на бланке задания.

Для расчета контрольной суммы можно использовать калькулятор `wcalc`

Пример расшифрованного пакета

Для примера показана расшифровка ARP-запроса:

Заголовок Ethernet кадра														*ARP пакет*	
MAC получателя						*MAC отправителя*						*Прот. ARP*		*прот. Eth-t*	
ff	ff	ff	ff	ff	ff	00	90	f5	3c	37	ef	08	06	00	01
ARP пакет															
прот. IP		*HLEN*	*PLEN*	*ARP запрос*		*MAC отправителя*						*IP-адр. отправителя*			
08	00	06	04	00	01	00	90	f5	3c	37	ef	c0	a8	01	0a
ARP пакет															
MAC получателя						*IP-адр. получателя*									
00	00	00	00	00	00	c0	a8	01	0c						

2.4 Содержание отчета

Заполненный бланк задания является отчетом для данной лабораторной работы.

2.5 Контрольные вопросы

1. Протокол IP. Формат пакета.
2. Принцип расчета контрольной суммы заголовка IP-пакета.
3. Протокол TCP. Формат сегмента.
4. Принцип расчета контрольной суммы заголовка TCP-сегмента.
5. Протокол UDP. Формат датаграммы.
6. Принцип расчета контрольной суммы заголовка UDP-датаграммы.

3 Изучение протоколов электронной почты

3.1 Цель работы

Изучение протоколов электронной почты SMTP и POP3.

3.2 Материал для изучения и ознакомления

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

1. Разделы 7, 8, 9 данного пособия.
2. Электронный курс «Структура и протоколы электронной почты в Internet». (`http://opds.sut.ru/`)
3. Электронная почта — Википедия (`http://ru.wikipedia.org/wiki/Электронная_почта`)
4. POP3 — Википедия (`http://ru.wikipedia.org/wiki/POP3`)
5. RFC 1939 «Протокол POP3» (`http://rfc2.ru/1939.rfc`)
6. SMTP — Википедия (`http://ru.wikipedia.org/wiki/SMTP`)
7. RFC 5321 «Протокол SMTP» (`http://rfc2.ru/5321.rfc`)
8. LMTP — Википедия (`http://ru.wikipedia.org/wiki/LMTP`)
9. RFC 2033 «LMTP Protocol» (`http://tools.ietf.org/html/rfc2033`)

3.3 Задание на лабораторную работу

IP-адрес сервера: `172.16.100.20`

Адрес e-mail:	`student01@opds.net`
(по номеру ПК)	`student02@opds.net`
	...
	`student16@opds.net`

Пароль: `12345`

Для установления соединения и передачи/приема письма использовать утилиту `telnet`.

3.4 Порядок выполнения лабораторной работы

3.4.1 Отправка письма по протоколу SMTP

1. Начать захват пакетов при помощи анализатора протоколов (`wireshark`, `tcpdump`). Захват проводить по фильтру (IP-адреса источника и получателя, протокол TCP, номер порта со стороны сервера; для `tcpdump` дополнительно указать размер пакета 1500 байт и запись в двоичный файл).
2. Соединиться с SMTP-сервером кафедры.
3. Сформировать и передать письмо.

4. Разорвать соединение.
5. Сохранить результаты работы в текстовый файл.
6. По захваченным пакетам построить диаграмму Flow Graph с помощью `wireshark`. Диаграмму сохранить либо в виде текстового файла либо в виде изображения.

3.4.2 Получение письма по протоколу POP3

1. Начать захват пакетов при помощи анализатора протоколов (`wireshark`, `tcpdump`). Захват проводить по фильтру (IP-адреса источника и получателя, протокол TCP, номер порта со стороны сервера; для `tcpdump` дополнительно указать размер пакета 1500 байт и запись в двоичный файл).
2. Соединиться с POP3-сервером кафедры.
3. Вывести на экран список писем.
4. Получить полный текст переданного почтового сообщения.
5. Удалить переданное почтовое сообщение.
6. Разорвать соединение.
7. Сохранить результаты работы в текстовый файл.
8. По захваченным пакетам построить диаграмму Flow Graph с помощью `wireshark`. Диаграмму сохранить либо в виде текстового файла либо в виде изображения.

3.5 Содержание отчета

1. Титульный лист согласно приложению.
2. Цель работы.
3. Результаты работы с протоколом SMTP.
4. Диаграмма Flow Graph для SMTP.
5. Результаты работы с протоколом POP3.
6. Диаграмма Flow Graph для POP3.

3.6 Контрольные вопросы

1. Структура электронного сообщения. Назначение полей заголовка электронного письма, полученного по протоколу POP3. [1, 2]
2. Адресация и маршрутизация электронной почты в Internet. [2, 3]
3. Структура электронной почты в Internet. [2, 3]
4. Протокол POP3. [4, 5]
5. Протокол SMTP. [6, 7]
6. Протокол LMTP. [8, 9]

4 Установление аудио/видеоконференцсвязи.

4.1 Цель работы

Изучение работы систем аудио/видеоконференцсвязи.

4.2 Материал для изучения и ознакомления

Перед выполнением работы рекомендуется ознакомиться со следующими материалами:

1. Конференц-связь — Википедия // `http://ru.wikipedia.org/wiki/Конференцсвязь`

2. Skype — Википедия // `http://ru.wikipedia.org/wiki/Skype`

4.3 Задание на лабораторную работу

Используя VoIP-сервис Skype и одноименный софтфон организовать конференцсвязь между тремя участниками.

4.4 Порядок выполнения лабораторной работы

4.4.1 Организация конференцсвязи с использованием Skype

1. Создать аккаунт Skype.
2. Добавить в список абонентов преподавателя и одного-двух студентов.
3. Организовать конференцию между абонентами.
4. Удалить созданный аккаунт.

4.5 Содержание отчета

1. Титульный лист согласно приложению.
2. Цель работы.
3. Порядок организации конференции в Skype.

4.6 Контрольные вопросы

1. Конференцсвязь. [1]

5 Создание простейшего веб-сайта на языке HTML

5.1 Цель работы

Изучение основ языка HTML. Получение практических навыков в написании простейших веб-страниц.

5.2 Задание на лабораторную работу

Создать веб-сайт (личную страницу студента) на языке HTML и разместить её в сети Интернет (можно использовать один из бесплатных хостингов или собственный веб-сервер).

На сайте должны быть использованы как минимум следующие элементы:

- заголовки;
- таблица;
- активные гиперссылки;
- рисунки;
- списки.

5.3 Порядок выполнения лабораторной работы

5.4 Содержание отчета

Отчётом по лабораторной является работающий сайт с доступом через сеть Интернет.

5.5 Контрольные вопросы

Защита лабораторной не требуется.

6 Организация потокового аудио/видео вещания

6.1 Цель работы

Изучение принципов организации потокового аудио и видео вещания. Получение практических навыков в организации потокового аудио/видео вещания.

6.2 Задание на лабораторную работу

Используя программный комплекс VideoLAN организовать потоковое вещание видео.

6.3 Порядок выполнения лабораторной работы

6.3.1 Организация потокового Unicast вещания с использованием протокола UDP

1. Скачать с локального сервера кафедры один из предоставленных видеофайлов.
2. Запустить VLC Media player на передающем ПК (в терминале команда `vlc`).
3. Настроить трансляцию:
   (a) открыть диалог выбора источника (пункт меню «Медиа» → «Передавать»);
   (b) на вкладке «Файл» добавить в список один или несколько видео файлов;
   (c) перейти к настройке типа трансляции (вывода потока) (кнопка «Поток»);
   (d) на вкладке «Настройка вывода» выбрать путь назначения «UDP (legacy)» и добавить его;
   (e) на появившейся вкладке указать IP-адрес ПК, на который будет осуществляться трансляция (номер порта можно либо оставить по умолчанию либо задать свой (более 1024));
   (f) в настройках перекодирования выбрать профиль «H.264 + AAC (TS)»;
   (g) запустить трансляцию.
4. Запустить VLC Media player на принимающем ПК и настроить приём видеопотока:
   (a) открыть диалог выбора источника сигнала (пункт меню «Медиа» → «Открыть URL...»);
   (b) ввести сетевой адрес `udp://@:2134`;
   (c) запустить просмотр (кнопка «Воспроизвести»).

6.3.2 Организация потокового вещания с использованием протокола HTTP

1. Запустить VLC Media player на передающем ПК (в терминале команда `vlc`).
2. Настроить трансляцию:
   (a) открыть диалог выбора источника (пункт меню «Медиа» → «Передавать»);
   (b) на вкладке «Файл» добавить в список один или несколько видео файлов;
   (c) перейти к настройке типа трансляции (вывода потока) (кнопка «Поток»);

(d) на вкладке «Настройка вывода» выбрать путь назначения «HTTP» и добавить его;

(e) на появившейся вкладке по умолчанию указан номер порта 8080 и путь «/»;

(f) в настройках перекодирования выбрать профиль «H.264 + AAC (TS)»;

(g) запустить трансляцию.

3. Запустить VLC Media player на нескольких ПК и настроить приём видеопотока; убедиться, что сигнал одновременно принимают несколько ПК:

(a) открыть диалог выбора источника сигнала (пункт меню «Медиа» → «Открыть URL...»);

(b) ввести сетевой адрес `http://`*`ip-адрес_передающего_ПК`*`:8080`;

(c) запустить просмотр (кнопка «Воспроизвести»).

6.4 Содержание отчета

Отчёт по лабораторной не требуется.

6.5 Контрольные вопросы

Защита лабораторной не требуется.

Часть II

Руководства к программам

7 Некоторые утилиты ОС GNU/Linux

7.1 man

man (от англ. manual — руководство) — команда Unix, предназначенная для форматирования и вывода справочных страниц, которая предоставляется почти со всеми *nix-дистрибутивами. Каждая страница справки является самостоятельным документом и пишется разработчиками программного обеспечения.

Чтобы вывести справочное руководство по какой-либо команде, можно ввести:

```
man <command_name>
```

в консоли. Например, чтобы посмотреть справку по команде `ls`, нужно ввести `man ls`.

Для навигации в справочной системе man можно использовать клавиши  и для построчного перехода,  и  для постраничного перехода вверх и вниз соответственно.

Для выхода из справочной системы используется клавиша  , соответствующая английскому слову Quit — выход.

Для получения детальной инструкции по использованию команды используется конструкция

```
man man
```

7.2 Консольные текстовые редакторы

1. `vim`
2. `mcedit`
3. `nano`

7.3 Просмотр содержимого файла в терминале

1. `cat` *`имя_файла`* — вывод всего файла на экран.
2. `less` *`имя_файла`* — постраничный вывод файла.

7.4 Консольные калькуляторы

1. `bc`
2. `calc`
3. `wcalc` — работает с разными системами счисления.

7.5 Перенаправление вывода консольной программы в файл

1. *`программа`* `>` *`имя_файла`* — создать новый файл или перезаписать (без уведомления) существующий (на экран результат не выводится).
2. *`программа`* `>>` *`имя_файла`* — дописать в конец файла (на экран результат не выводится).

3. *программа* | tee *имя_файла* — создать новый файл или перезаписать (без уведомления) существующий (результат выводится на экран).

7.6 Создание скриншотов экрана

1. При помощи программы GIMP:

   Для этого в GIMP есть встроенная функция, которая позволяет делать снимки экрана. Вызывается в меню
   `Файл` → `Создать` → `Снимок экрана...`

   Откроется окно с настройками для создания скриншота. В окне есть выбор задержки перед захватом экрана в секундах и два варианта для создания снимка: «Захватить одно окно» и «Захватить весь экран».

   При захвате одного окна откроется небольшое окошко с крестиком посередине. Этот крестик нужно перетащить на то окно, которое вы хотите «сфотографировать». После этого снимок создастся как новое изображение.

2. При помощи пакета `imagemagick`: Используется команда:

   ```
   import -pause 5 -window root screen.png
   ```

   Приведённая команда создаёт скриншот всего экрана с задержкой в 5 секунд и сохраняет его в файл `screen.png`

8 Руководство по работе с программой `tcpdump`

8.1 Краткое описание

`tcpdump` (от TCP и англ. dump — свалка, сбрасывать) — утилита UNIX, позволяющая перехватывать и анализировать сетевой трафик, проходящий через компьютер, на котором запущена данная программа.

Программа `tcpdump` была написана Van Jacobson, Craig Leres и Steven McCanne, во время их работы в лаборатории Lawrence Berkeley, Калифорнийского университета, Беркли.

Программа `tcpdump` разработана таким образом, чтобы переводить сетевую плату в смешанный режим (promiscuous mode), при этом, каждый пакет, проходящий по кабелю, фиксируется. Обычно сетевые платы для сред передачи, таких как Ethernet, захватывают только фреймы канального уровня, адресованные конкретному интерфейсу или отправленные на широковещательный адрес.

Операционная система должна позволить поместить интерфейс в смешанный режим и позволить пользовательскому процессу захватывать фреймы. Поэтому, для выполнения программы требуется наличие прав суперпользователя и прямой доступ к устройству.

Основные назначения tcpdump:

- Отладка сетевых приложений.
- Отладка сети и сетевой конфигурации в целом.

8.2 Запуск программы

```
tcpdump [параметры] [фильтр]
```

В ОС Debian/Ubuntu Linux запуск `tcpdump` осуществляется через `sudo`:

```
sudo tcpdump [параметры] [фильтр]
```

Фильтр необходим для того, чтобы отображать на экране (или сохранять в файл) только определенные пакеты (т. е. те, которые соответствуют фильтру). Фильтр представляет из себя один или несколько примитивов фильтрации, объединенных логическими операторами. Предпочтительно (но не обязательно) записывать строку фильтрации в апострофах (см. примеры).

8.3 Примеры использования

1. Перехватить пакеты ICMP отправленные на хост `192.168.0.1`. Отобразить пакет в шестнадцатеричном виде с заголовком канального уровня. Размер пакета 1500 байт.

   ```
   tcpdump -s 1500 -xx 'dst host 192.168.0.1 and ip proto \icmp'
   ```

2. Перехватить пакеты TCP отправленные с ПК 192.168.0.1 на ПК 192.168.0.2 и обратно. Отобразить пакет в ASKII и шестнадцатеричном виде с заголовком канального уровня. Ограничиться 6 пакетами.

   ```
   tcpdump -c 6 -XX 'host 192.168.0.1 and host 192.168.0.2 and ip proto \tcp'
   ```

.

8.4 Основные параметры (флаги)

`-i`	Указывает на то, какой сетевой интерфейс будет использоваться для захвата пакетов. По-умолчанию — `eth0`, но если отсутствует локальная сеть, то можно воспользоваться интерфейсом обратной связи `lo`.
`-e`	Отображает данные канального уровня (MAC-адрес, протокол, длина пакета). Вместо IP-адресов будут отображаться mac-адреса компьютеров.
`-w`	Сохраняет данные `tcpdump` в двоичном формате. Преимущества использования данного способа по сравнению с обычным перенаправлением в файл является высокая скорость записи и возможность чтения подобных данных другими программами, например `snort` или `wireshark`, но этот файл нельзя прочитать человеку.
`-r`	Этот параметр позволяет tcpdump прочесть трафик из файла, если он был предварительно сохранен параметром -w.
`-x`	Делает распечатку пакета в шестнадцатеричной системе, полезно для более детального анализа пакета. Количество отображаемых данных зависит от опции -s
`-xx`	Тоже, что и предыдущий параметр, но включает в себя заголовок канального уровня.
`-X`	Выводит пакет в ASCII- и hex-формате. Полезно в случае анализа инцидента связанного со взломом, так как позволяет просмотреть какая текстовая информация передавалась во время соединения.
`-XX`	Тоже, что и предыдущий параметр, но включает заголовок канального уровня.
`-s` число	Количество байтов пакета, которые будет обрабатывать `tcpdump`. При установке большого числа отображаемых байтов информация может не уместиться на экране и ее будет трудно изучать. В зависимости от того, какие цели вы преследуете, и следует выбирать значение этого параметра. По-умолчанию `tcpdump` сохраняет первые 68 байт, однако если вы хотите увидеть содержимое всего пакета, используйте значение в 1500 байт (максимально допустимый размер пакета в сети Ethernet).
`-S`	Позволяет не обрабатывать абсолютные порядковые номера (initial sequence number — ISN) в относительные.
`-n`	Отображает IP-адрес вместо имени хоста.
`-a`	Преобразовывает сетевые и широковещательные адреса в доменные имена.
`-c` число	`tcpdump` завершит работу после получения указанного числа пакетов.
`-nn`	Отображает номер порта вместо используемого им протокола.
`-N`	Не добавляет доменное расширение к именам узлов. Например `tcpdump` отобразит '`net`' вместо '`net.library.org`'
`-p`	Не переводит интерфейс в беспорядочный (promiscuous)режим.
`-q`	Выводит минимум информации. Обычно это имя протокола, откуда и куда шел пакет, порты и количество переданных данных.
`-t`	Не отображает метку времени в каждой строке.
`-tt`	Отображает неформатированную метку времени в каждой строке.
`-tttt`	Показывает время вместе с датой.
`-T` тип	Интерпретация пакетов заданного типа. Поддерживаются типы `aodv`, `cnfp`, `rpc`, `rtp`, `rtcp`, `snmp`, `tftp`, `vat`, `wb`.

`-v`	Вывод подробной информации (TTL; ID; общая длина заголовка, а также его параметры; производит проверку контрольных сумм IP и ICMP-заголовков)
`-vv`	Вывод еще более полной информации, в основном касается NFS и SMB.
`-vvv`	Вывод максимально подробной информации.
`-l`	Использовать стандартный потоковый вывод `tcpdump` (stdout), например для записи в файл: `tcpdump -l | tee out.log` отобразит работу `tcpdump` и сохранит результат в файле `out.log`
`-F файл`	Использовать фильтр, находящийся в файле. Если вы используете этот параметр, фильтр из командной строки будет игнорироваться.

8.5 Примитивы фильтрации пакетов

Примитив	Описание
`dst host <хост>`	Будет отбирать пакеты, в которых поле адреса получателя IPv4/v6 содержит адрес хоста, заданного в примитиве.
`src host <хост>`	Будет выбирать все пакеты, в которых поле отправителя содержит адрес указанного хоста.
`host <хост>`	Будет отбирать все пакеты, для которых адрес хоста указан в поле получателя или отправителя.
`ether dst <ehost>`	Будет выбирать все кадры, в которых поле MAC-адреса получателя содержит значение `ehost`.
`ether src <ehost>`	Будет выбирать все кадры, в которых поле MAC-адреса отправителя содержит значение `ehost`.
`ether host <ehost>`	Будет отбирать все пакеты с адресом, указанным значением `ehost` в поле отправителя или получателя.
`gateway <шлюз>`	Будет отбирать все пакеты, использующие указанный именем хост в качестве шлюза.
`dst net <сеть>`	Отбирает все пакеты IPv4/v6, направленные в указанную сеть.
`src net <сеть>`	Выбирает все пакеты IPv4/v6, отправленные из указанной сети.
`net <сеть>`	Выбирает все пакеты IPv4/v6, содержащие адреса из указанной сети в поле отправителя или получателя.
`dst port <порт>`	Отбирает все пакеты IP/TCP, IP/UDP, IPv6/TCP и IPv6/UDP, направленные в указанный порт. Номера портов могут задаваться номерами или именами из файла `/etc/services`. При указании имени (протокол/порт) проверяется как порт, так и протокол. Если примитив содержит номер или неоднозначное обозначение порта (только порт, без протокола) фильтру будут соответствовать пакеты обоих протоколов (tcp и udp). Например, фильтру `dst port 513` будут соответствовать пакеты TCP/login и UDP/who, а фильтру `port domain` — трафик TCP/domain и UDP/domain.
`src port <порт>`	Отбирает все пакеты, отправленные из указанного порта.

Примитив	Описание
`port <порт>`	Отбирает все пакеты, содержащие указанный номер порта в поле отправителя или получателя. Любое из трех перечисленных правил для портов может включать в качестве префикса идентификатор протокола TCP или UDP (например, `tcp src port <порт>`, будет отбирать пакеты TCP, отправленные из указанного порта).
`less <размер>`	Будет собирать пакеты, размер которых не превышает указанного значения.
`greater <размер>`	Будет собирать пакеты, размер которых не меньше указанного значения.
`ip proto <протокол>`	Отбирает все пакеты IP, содержащие заданный идентификатор типа в поле типа протокола. Типы протоколов IP можно указывать по именам (icmp, icmp6, igmp, igrp, pim, ah, esp, vrrp, udp, tcp) или номерам. Поскольку tcp, udp и icmp используются также в качестве ключевых слов, перед этими идентификаторами следует помешать символ '\'. Отметим, что этот примитив не проверяет цепочки протокольных заголовков.
`ether proto <протокол>`	Отбирает кадры Ethernet с заданным типом протокола. Протокол может быть указан по номеру или имени (ip, ip6, arp, rarp, atalk, aarp, decnet, sca, lat, mopdl, moprc, iso, stp, ipx, netbeui).

8.6 Логические выражения для группировки примитивов

- скобок;
- отрицания (`!` или `not`);
- логического пересечения (`&&` или `and`);
- логического объединения (`||` или `or`).

Оператор отрицания имеет высший уровень приоритета, операции объединения и пересечения имеют одинаковый приоритет и выполняются слева направо в порядке следования. Отметим, что для операции логического пересечения недостаточно просто указать операнды рядом, а требуется явно задать операцию (`&&` или `and`).

Аргументы выражений могут передаваться программе `tcpdump` как один или множество аргументов. В общем случае выражения, содержащие мета-символы командного интерпретатора, должны передаваться как один аргумент, заключенный в кавычки или апострофы.

8.7 Источники

При написании руководства были использованы следующие источники:

1. tcpdump — Википедия (`http://ru.wikipedia.org/wiki/Tcpdump`)
2. Программа tcpdump / TCP/IP Крупным планом (`http://www.soslan.ru/tcp/tcp_a.html`)

3. Проект OpenNet: MAN tcpdump (8)
   (`http://www.opennet.ru/man.shtml?topic=tcpdump&category=8&russian=2`)

4. Краткий man по tcpdump (`http://www.inattack.ru/article/267.html`)

Для более подробного ознакомления с возможностями утилиты `tcpdump` рекомендуется ознакомиться с полным руководством к программе.

Для этого используется команда: `man tcpdump`

9 Руководство по работе с программой `wireshark`

9.1 Краткое описание

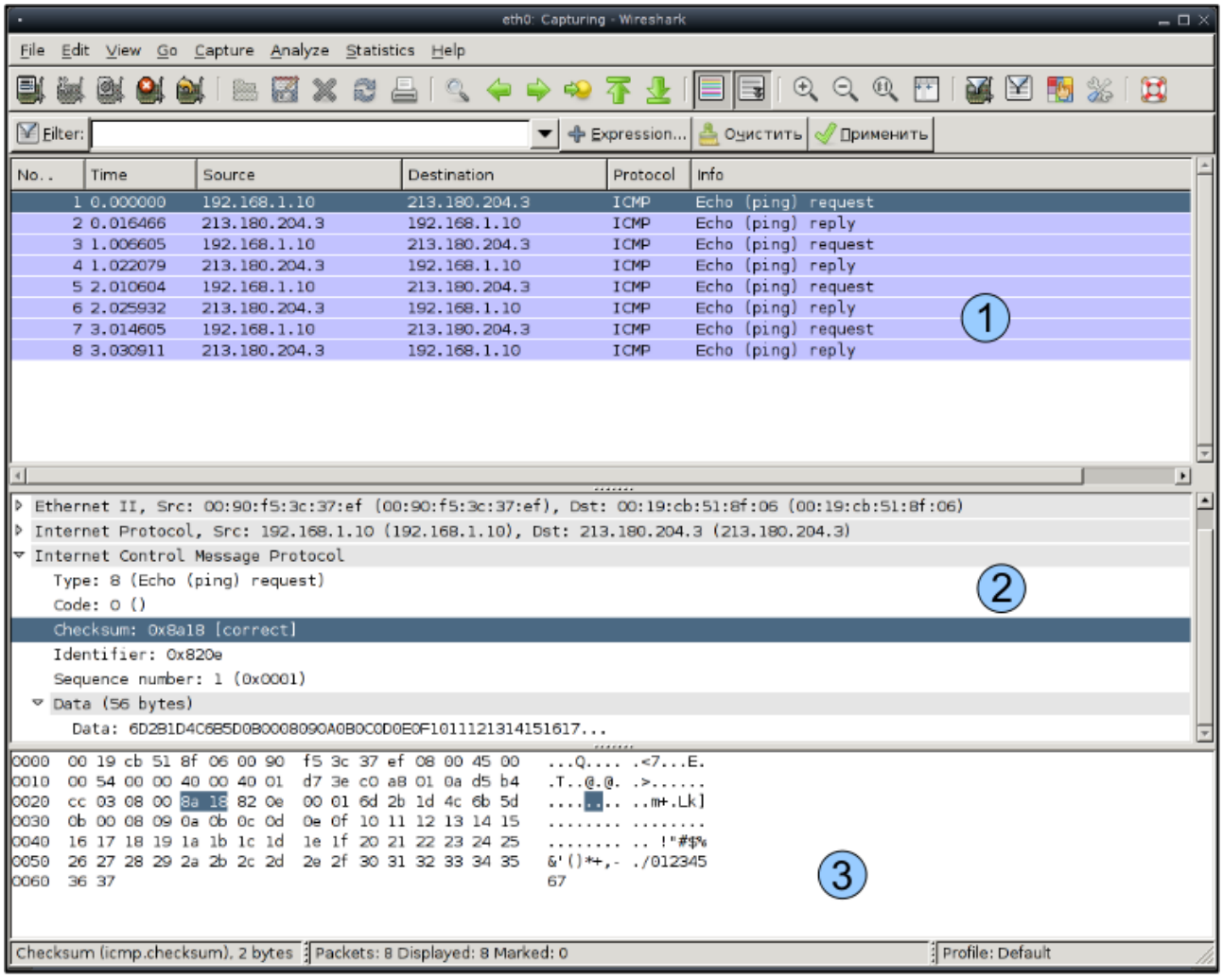
`Wireshark` — программа для анализа пакетов Ethernet и других сетей (сниффер). Позволяет просматривать весь проходящий по сети трафик в режиме реального времени, переводя сетевую карту в широковещательный режим, и при этом имеет большое количество возможностей по сортировке и фильтрации информации. Поддерживает DNS, FDDI, FTP, HTTP, ICQ, IPV6, IPX, IRC, MAPI, MOUNT, NETBIOS, NFS, NNTP, POP, PPP, TCP, TELNET, X25 и т.д. `Wireshark` способен открывать файлы данных, захваченных другими программами, что расширяет возможности захвата. Распространяется под лицензией GNU GPL и выпускается в версиях для MS Windows, Mac OS X и GNU/Linux.

9.2 Запуск программы

В ОС Linux программа `wireshark` должна запускаться от имени суперпользователя. Поэтому запуск программы производится из терминала (консоли) следующей командой:

```
sudo wireshark &
```

9.3 Внешний вид



1. Захваченные программой пакеты.

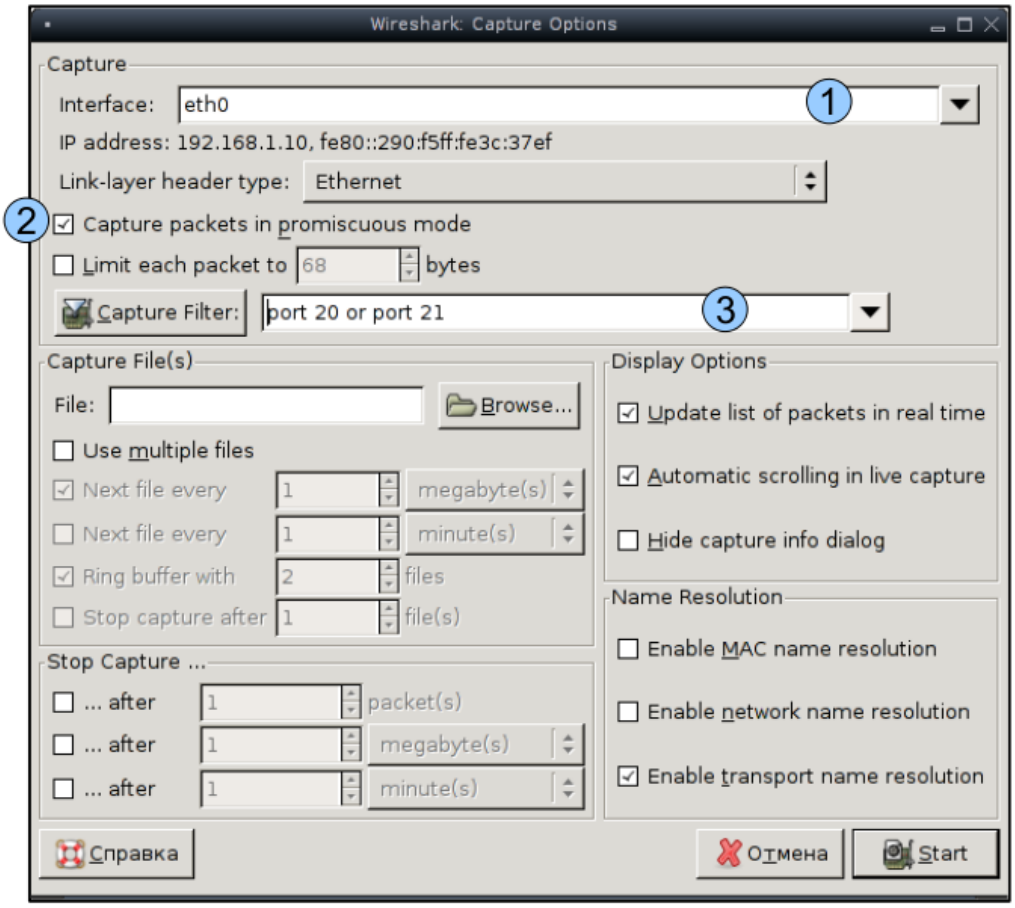
2. Структура выделенного в первом окне пакета и расшифровка его содержимого.

3. Представление выделенного в первом окне пакета в шестнадцатеричном виде.

9.4 Основные пункты главного меню

File → Open	Открыть ранее сохраненный список захваченных пакетов
Capture → Interfaces	Начать захват пакетов с определенного интерфейса
Capture → Options	Настроить захват пакетов
Capture → Start	Начать захват пакетов с интерфейса по умолчанию
Capture → Stop	Остановить захват пакетов
Statistics → Flow Graph	Построить диаграмму обмена по захваченным пакетам

9.5 Окно настроек захвата пакетов



1. Интерфейс с которого захватывать пакеты.

2. Флаг включающий захват всех пакетов в сети.

3. Строка ввода фильтра захвата.

9.6 Примеры фильтров

1. Захват пакетов с определенного MAC-адреса:

`ether host [MAC-адрес]`

Пример:

`ether host 00:08:15:00:08:15`

2. Захват пакетов с определенного IP-адреса:

   `host [IP-адрес]`

   Пример:

   `host 192.168.0.1`

3. Захват пакетов с определенного порта:

   `port [номер порта]`

   Пример:

   `port 80`

4. Захват пакетов определенного протокола:

   `[Наименование протокола]`

   Примеры:

   ```
   udp
   tcp
   ip
   dns
   arp
   icmp
   ```

5. Использование нескольких фильтров вместе:

   `[фильтр] and [фильтр]`

   Пример: захватывать пакеты ICMP с IP-адреса `192.168.0.1`

   `icmp and host 192.168.0.1`

6. Использование условия "или":

   `[фильтр] or [фильтр]`

   Пример: захватывать и пакеты ICMP и пакеты DNS

   `icmp or dns`

7. Использование условия "не":

   `not [фильтр]`

   Пример: не захватывать пакеты ICMP

   `not icmp`

9.7 Источники

При написании руководства были использованы следующие источники:

1. manpages для `wireshark`

Часть III

Приложение

Правила оформления отчетов

1. Структура отчета должна соответствовать требованиям представленным в соответствующем пункте лабораторной работы.

2. Размер основного шрифта отчета: 11–12 pt

3. Результаты работы консольных программ, сами запускаемые команды и диаграммы в текстовом виде должны быть оформлены моноширинным шрифтом (Courier New, Lusida Console, FreeMono и т. п.).

   Например:

```
student@comp:~\$ ping -c 4 www.ya.ru
PING ya.ru (87.250.250.3) 56(84) bytes of data.
64 bytes from www.yandex.ru (87.250.250.3): icmp_seq=1 ttl=52 time=16.8 ms
64 bytes from www.yandex.ru (87.250.250.3): icmp_seq=2 ttl=52 time=16.8 ms
64 bytes from www.yandex.ru (87.250.250.3): icmp_seq=3 ttl=52 time=18.7 ms
64 bytes from www.yandex.ru (87.250.250.3): icmp_seq=4 ttl=52 time=13.5 ms

--- ya.ru ping statistics ---
4 packets transmitted, 4 received, 0\% packet loss, time 3014ms
rtt min/avg/max/mdev = 13.542/16.484/18.759/1.874 ms
```

4. Результаты работы консольных программ и диаграммы в текстовом виде должны вмещаться в ширину страницы (шрифт можно уменьшать до 9 pt). Если ширины вертикально расположенного листа A4 не хватает, то можно разместить диаграмму на нескольких горизонтально расположенных листах A4.

5. Текст на диаграммах и графиках должен быть свободно читаем.

6. На графиках должны быть подписаны оси и единицы измерения.

7. Листы отчета должны быть скреплены между собой (скрепкой, степлером и т. п.).

Титульный лист

Санкт-Петербургский университет телекоммуникаций
им. проф. М. А. Бонч-Бруевича

Кафедра ОПДС

Лабораторная работа №0
по курсу СДЭС

Название лабораторной работы

Группа: ГГ-00

Студент: Фамилия И. О.

Вариант: 00

Санкт-Петербург 2010