Лабораторная работа № 3-4

**Сетевое программирование с использованием RAW-сокетов.**

*Цель работы:* **1.** Изучить принципы работы с RAW-сокетами. **2.** Научиться формировать IP, TCP, UDP, ICMP пакеты на RAW сокетах. **3.** Изучить принципы приема пакетов с использованием RAW сокетов.

*Продолжительность работы* – 8 ч.

**Задания для самостоятельной работы**

**Вариант №1**

**Генератор IP-пакетов (см. приложение 1)**

***1. Описание задачи***

Реализация генератора IP-пакетов в соответствии со стандартом RFC 791.

***2. Требования к реализации***

* Приложение должно выполнять функции генерации IP пакетов.
* Приложение должно позволять задавать адрес отправителя и получателя пакета
* Графический интерфейс.
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

**Сниффер (см. приложение 3)**

***1. Описание задачи***

Необходимо разработать программу для получения всех пакетов начиная с уровня IP-протокола, получаемых сетевым адаптером, используя RAW Socket. Программа должна выводить значения всех полей заголовков пакетов IP, ICMP, TCP, UDP на экран/в файл.

***2. Требования к реализации***

Приложение должно выполнять:

* Cбор приходящих пакетов.
* Анализ заголовка пришедшего пакета.
* Вывод результатов на экран или в файл (включая поле данных).
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

Примечание: Следует сохранять полученный пакет до анализа в виде HEX + ASCII, после анализа в виде «Название поля» = значение, с указанием вида представления (DEC, HEX, BIN). Выбор представления значения зависит от характера информации, содержащейся в нем. Например, контрольную сумму следует выводить в виде HEX, IP-адрес в виде DEC.DEC.DEC.DEC, флаги в битовом виде.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

***4. Дополнительные требования***

* Предусмотреть реализацию фильтров по протоколам
* Предусмотреть возможность остановки/приостановки процесса перехвата пакетов из сети.

.

**Вариант №2**

**Генератор TCP-пакетов (см. приложение 1)**

***1. Описание задачи***

Реализация генератора TCP-пакетов в соответствии со стандартом RFC 793.

***2. Требования к реализации***

* Приложение должно выполнять функции генерации TCP пакетов.
* Приложение должно позволять задавать: адрес отправителя и получателя пакета, порт отправителя и получателя, передаваемые данные.
* Графический интерфейс.
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

**Сниффер (см. приложение 3)**

***1. Описание задачи***

Необходимо разработать программу для получения всех пакетов начиная с уровня IP-протокола, получаемых сетевым адаптером, используя RAW Socket. Программа должна выводить значения всех полей заголовков пакетов IP, ICMP, TCP, UDP на экран/в файл.

***2. Требования к реализации***

Приложение должно выполнять:

* Cбор приходящих пакетов.
* Анализ заголовка пришедшего пакета.
* Вывод результатов на экран или в файл (включая поле данных).
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

Примечание: Следует сохранять полученный пакет до анализа в виде HEX + ASCII, после анализа в виде «Название поля» = значение, с указанием вида представления (DEC, HEX, BIN). Выбор представления значения зависит от характера информации, содержащейся в нем. Например, контрольную сумму следует выводить в виде HEX, IP-адрес в виде DEC.DEC.DEC.DEC, флаги в битовом виде.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

***4. Дополнительные требования***

* Предусмотреть реализацию фильтров по протоколам
* Предусмотреть возможность остановки/приостановки процесса перехвата пакетов из сети.

**Вариант №3**

**Генератор ICMP-пакетов (см. приложение 1)**

***1. Описание задачи***

Реализация генератора ICMP-пакетов в соответствии со стандартом RFC 792.

***2. Требования к реализации***

* Приложение должно выполнять функции генерации ICMP пакетов различного типа.
* Приложение должно позволять задавать: адрес отправителя и получателя пакета, тип ICMP пакета, код ICMP пакета.
* Графический интерфейс.
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

**Сниффер (см. приложение 3)**

***1. Описание задачи***

Необходимо разработать программу для получения всех пакетов начиная с уровня IP-протокола, получаемых сетевым адаптером, используя RAW Socket. Программа должна выводить значения всех полей заголовков пакетов IP, ICMP, TCP, UDP на экран/в файл.

***2. Требования к реализации***

Приложение должно выполнять:

* Cбор приходящих пакетов.
* Анализ заголовка пришедшего пакета.
* Вывод результатов на экран или в файл (включая поле данных).
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

Примечание: Следует сохранять полученный пакет до анализа в виде HEX + ASCII, после анализа в виде «Название поля» = значение, с указанием вида представления (DEC, HEX, BIN). Выбор представления значения зависит от характера информации, содержащейся в нем. Например, контрольную сумму следует выводить в виде HEX, IP-адрес в виде DEC.DEC.DEC.DEC, флаги в битовом виде.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

***4. Дополнительные требования***

* Предусмотреть реализацию фильтров по протоколам
* Предусмотреть возможность остановки/приостановки процесса перехвата пакетов из сети.

**Вариант №4**

**Реализация эхо-сообщения ICMP (см. приложение 2)**

***1. Описание задачи***

Необходимо разработать программу для определения времени двойного оборота пакета до тестируемого узла и обратно (аналог утилиты Ping), используя RAW socket (ICMP\_ECHO и ICMP\_ECHOREPLY).

***2. Требования к реализации***

* Приложение должно выполнять функции отправки ICMP\_ECHO запросов и приема ответов - ICMP\_ECHOREPLY.
* Приложение должно позволять задавать адрес назначения.
* Графический интерфейс.
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

**Сниффер (см. приложение 3)**

***1. Описание задачи***

Необходимо разработать программу для получения всех пакетов начиная с уровня IP-протокола, получаемых сетевым адаптером, используя RAW Socket. Программа должна выводить значения всех полей заголовков пакетов IP, ICMP, TCP, UDP на экран/в файл.

***2. Требования к реализации***

Приложение должно выполнять:

* Cбор приходящих пакетов.
* Анализ заголовка пришедшего пакета.
* Вывод результатов на экран или в файл (включая поле данных).
* Отображать сообщения о возникающих ошибках и корректно их обрабатывать.

Примечание: Следует сохранять полученный пакет до анализа в виде HEX + ASCII, после анализа в виде «Название поля» = значение, с указанием вида представления (DEC, HEX, BIN). Выбор представления значения зависит от характера информации, содержащейся в нем. Например, контрольную сумму следует выводить в виде HEX, IP-адрес в виде DEC.DEC.DEC.DEC, флаги в битовом виде.

***3. Требования к надежности***

К приложению предъявляются следующие требования по надежности:

* Не допускается зависание приложения при любых действиях пользователя.
* Не допускается аварийное завершение приложения при любых действиях пользователя.
* Любая ошибочная ситуация должна корректно обрабатываться с выводом соответствующего сообщения.
* Не допускается утечка памяти/дескрипторов в процессе эксплуатации приложения.
* Не допускается полная загрузка процессора приложением в пассивном состоянии.

***4. Дополнительные требования***

* Предусмотреть реализацию фильтров по протоколам
* Предусмотреть возможность остановки/приостановки процесса перехвата пакетов из сети.

**Контрольные вопросы**

1. Что такое Winsock? (Если была использована другая библиотека – уметь рассказать про нее).

2. Что такое сокет?

3. Какие сокеты называются “сырыми сокетами”?

4. Расскажите коротко о стандарте ***RFC* 791.**

5. Расскажите коротко о стандарте ***RFC* 768**

6. Расскажите коротко о стандарте ***RFC* 793.**