

## Практика 9. Сетевой уровень (сдать до 23.04.2022)

### 1. Wireshark: ICMP

В лабораторной работе предлагается исследовать ряд аспектов протокола ICMP:

- ICMP-сообщения, генерируемые программой Ping
- ICMP-сообщения, генерируемые программой Traceroute
- Формат и содержимое ICMP-сообщения.

#### 1. Ping (4 балла)

Программа Ping на исходном хосте посылает пакет на целевой IP-адрес; если хост с этим адресом активен, то программа Ping на нем откликается, отсылая ответный пакет хосту, инициировавшему связь. Оба этих пакета Ping передаются по протоколу ICMP.

Выберите какой-либо хост, расположенный на другом континенте (например, в Америке или Азии). Захватите с помощью Wireshark ICMP пакеты от утилиты ping.

Для этого из командной строки запустите команду (аргумент *-n 10* означает, что должно быть отослано 10 ping-сообщений): *ping -n 10 имя\_хоста*

Для анализа пакетов в Wireshark введите строку *icmp* в области фильтрации вывода.

Ответьте на вопросы и подготовьте подтверждающие скрины:

1. Каков IP-адрес вашего хоста? Каков IP-адрес хоста назначения?
2. Почему ICMP-пакет не обладает номерами исходного и конечного портов?
3. Рассмотрите один из ping-запросов, отправленных вашим хостом. Каковы ICMP-тип и кодовый номер этого пакета? Какие еще поля есть в этом ICMP-пакете? Сколько байт приходится на поля контрольной суммы, порядкового номера и идентификатора?
4. Рассмотрите соответствующий ping-пакет, полученный в ответ на предыдущий. Каковы ICMP-тип и кодовый номер этого пакета? Какие еще поля есть в этом ICMP-пакете? Сколько байт приходится на поля контрольной суммы, порядкового номера и идентификатора?

#### 2. Traceroute (4 балла)

Программа Traceroute может применяться для определения пути, по которому пакет попал с исходного на конечный хост.

Traceroute отсылает первый пакет со значением TTL=1, второй – с TTL=2 и т.д. Каждый маршрутизатор понижает TTL-значение пакета, когда пакет проходит через этот маршрутизатор. Когда на маршрутизатор приходит пакет со значением TTL=1, этот маршрутизатор отправляет обратно к источнику ICMP-пакет, свидетельствующий об ошибке.

Задача – захватить ICMP пакеты, инициированные программой traceroute, в сниффере Wireshark.

В ОС Windows вы можете запустить: *tracert имя\_хоста*

Выберите хост, который **расположен на другом континенте**.

Ответьте на вопросы и подготовьте подтверждающие скрины:

1. Рассмотрите ICMP-пакет с эхо-запросом на вашем скриншоте. Отличается ли он от ICMP-пакетов с ping-запросами из Задания 1 (Ping)? Если да – то как?
2. Рассмотрите на вашем скриншоте ICMP-пакет с сообщением об ошибке. В нем больше полей, чем в ICMP-пакете с эхо-запросом. Какая информация содержится в этих дополнительных полях?

3. Рассмотрите три последних ICMP-пакета, полученных исходным хостом. Чем эти пакеты отличаются от ICMP-пакетов, сообщающих об ошибках? Чем объясняются такие отличия?
4. Есть ли такой канал, задержка в котором существенно превышает среднее значение? Можете ли вы, опираясь на имена маршрутизаторов, определить местоположение двух маршрутизаторов, расположенных на обоих концах этого канала?

## 2. Программирование.

### 1. IP-адрес и маска сети (1 балл)

Выведите IP-адрес вашего компьютера и маску сети на консоль.

### 2. Доступные порты (2 балла)

Выведите все доступные (свободные) порты в указанном диапазоне для заданного IP-адреса.

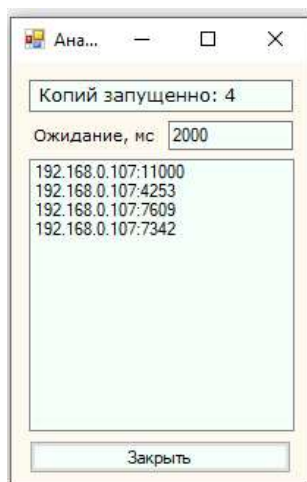
### 3. Широковещательная рассылка для подсчета копий приложения (6 баллов)

Разработать приложение, подсчитывающее количество копий себя, запущенных в локальной сети. Приложение должно использовать набор сообщений, чтобы информировать другие приложения о своем состоянии. После запуска приложение должно рассылать широковещательное сообщение о том, что оно было запущено. Получив сообщение о запуске другого приложения, оно должно сообщать этому приложению о том, что оно работает. Перед завершением работы приложение должно информировать все известные приложения о том, что оно завершает работу. На экран должен выводиться список IP адресов компьютеров (с указанием портов), на которых приложение запущено.

Приложение считает другое приложение запущенным, если в течение промежутка времени, равного нескольким интервалам между рассылками широковещательных сообщений, от него пришло сообщение.

(\*Такое приложение может быть использовано, например, при наличии ограничения на количество лицензионных копий приложений)

Пример GUI:



### 3. Задачи. Работа протокола TCP

#### 1. Докажите формулы (3 балла)

Пусть за период времени, в который изменяется скорость соединения с  $W/(2 \times RTT)$  до  $W/RTT$ , только один пакет был потерян (очень близко к концу периода).

1. Докажите, что частота потери (доля потерянных пакетов) равна

$$L = \text{частота потери} = \frac{1}{\frac{3}{8}W^2 + \frac{3}{4}W}$$

2. Используйте выше полученный результат, чтобы доказать, что, если частота потерь равна  $L$ , то средняя скорость приблизительно равна

$$\approx \frac{1,22 \times MSS}{RTT\sqrt{L}}$$

#### 2. Найдите функциональную зависимость (3 балла)

Рассмотрим модификацию алгоритма управления перегрузкой протокола TCP. Вместо аддитивного увеличения, мы можем использовать мультипликативное увеличение. TCP-отправитель увеличивает размер своего окна в небольшую положительную константу  $a$  ( $a > 1$ ), как только получает верный ACK-пакет.

1. Найдите функциональную зависимость между частотой потерь  $L$  и максимальным размером окна перегрузки  $W$ .
2. Докажите, что для этого измененного протокола TCP, независимо от средней пропускной способности, TCP-соединение всегда требуется одинаковое количество времени для увеличения размера окна перегрузки с  $W/2$  до  $W$ .