**Оглавление**

[**1.Постановска задачи** 2](#_Toc433229715)

[**2. Выполнение лабораторной работы** 2](#_Toc433229716)

# **1.Постановска задачи**

1. Запустить утилиту wireshark
2. Выбрать интерфейс p5p1и посмотреть передачу/приём пакетов и запросов
3. В параллельной консоли пропинговать сервер кафедры (10.0.75.11)
4. Описать запрос ping reply.

# **2. Выполнение лабораторной работы**

Утилита **WireShark** - [программа-анализатор трафика](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%82%D0%BE%D1%80_%D1%82%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0) для компьютерных сетей [Ethernet](https://ru.wikipedia.org/wiki/Ethernet) и некоторых других. Имеет [графический пользовательский интерфейс](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81).

Основные возможности:

* Перехват трафика в режиме онлайн
* Протокольные декодировщики декодировщиков (SMB, MySQL, TELNET, POP, RLOGIN, NAPSTER, IRC, RIP, FTP, ICQ, HTTP, NNTP, X11, BGP, IMAP 4, SOCKS 5, NFS, SNMP, MSN,VNC, LDAP,  YMSG).
* Импорт и экспорт данных из других пакетных анализаторов (Sun snoop, libpcap, Shomiti/Finisar Surveyor, tcpdump, atmsnoop,  Microsoft Network Monitor, Novell LANalyzer, AIX’s iptrace).
* Поиск пакетов по множеству показателей
* Подсветка захваченных пакетов разных протоколов
* Поддерживает создание статистики

1. **Запуск Wireshark**

Для запуска утииты WireShark необходимо в консоли набрать **# wireshark**, после чего произойдет запуск графического интерфейса. Далее, выбрать интерфейс p5p1.

1. **Пинг другого компьютера**

Параллельно в терминале пропинговать другой компьютер. При пинговани хоста в утилите отображается ping request (запрос) и ping reply (ответ). Результат ping request представлен на рисунке 1.

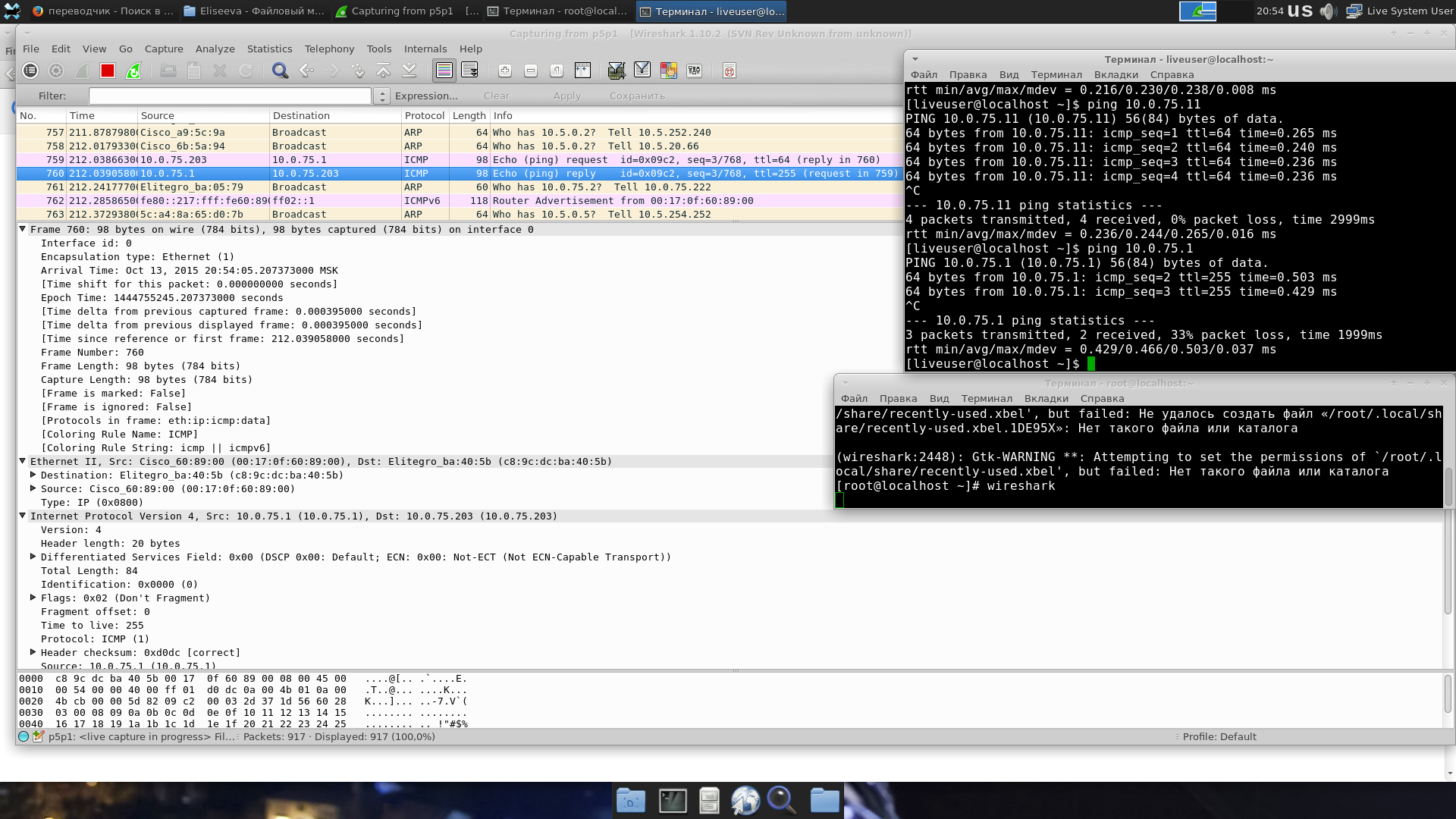


Рисунок 1. Пинг 10.0.75.11 и отслеживание трафика в wireshark

При данном запросе мы можем наблюдать как протокол ICMP инкапсулируется в IP (сетевой уровень), а IP в свою очередь инкапсулируется в Ethernet.

Рассмотрим подробнее ping reply (рисунок 2).

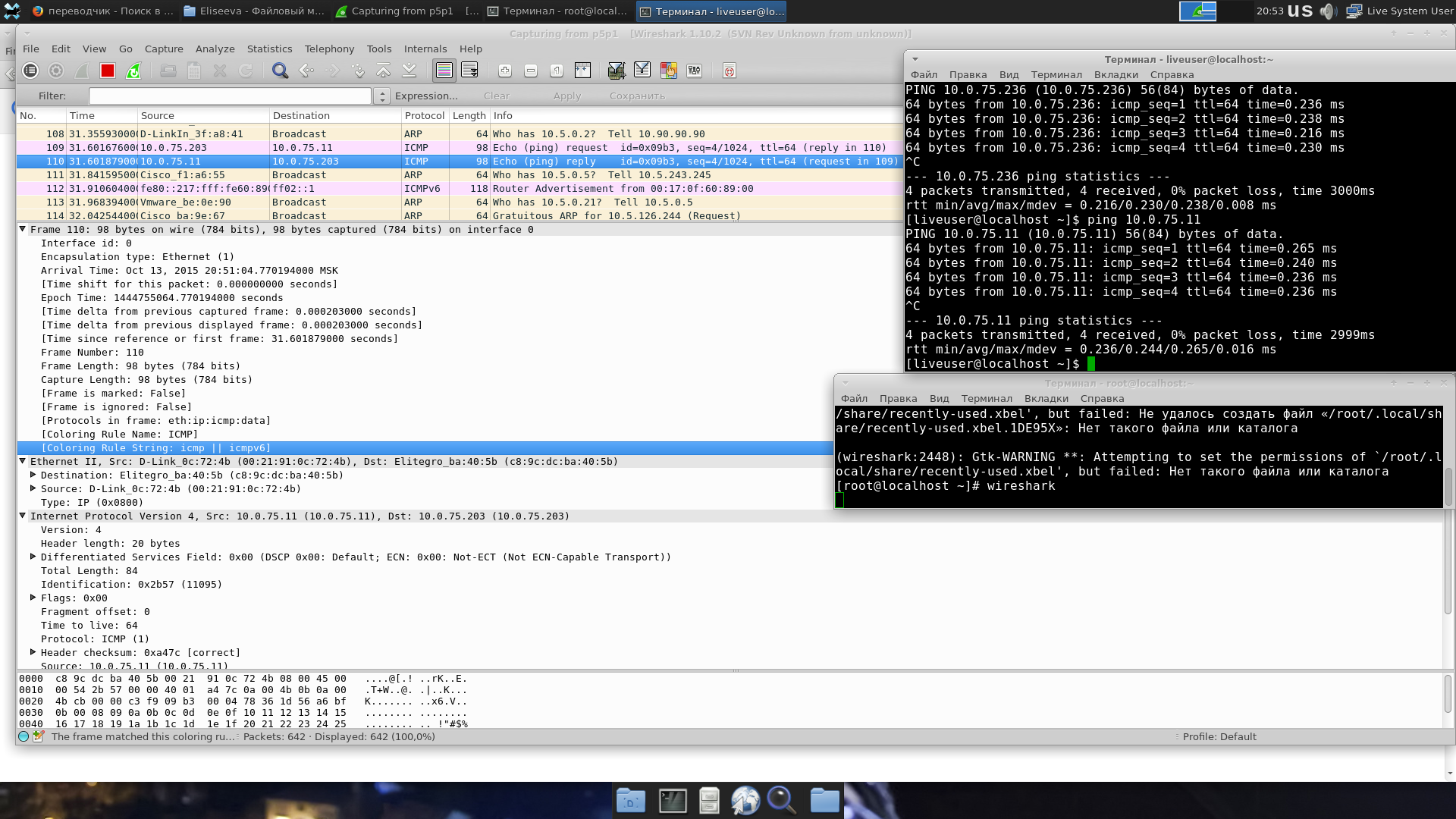


Рисунок 2. Ping reply

Рассмотрим каждый уровень формирования Ethernet-кадра

1. **ICMP**

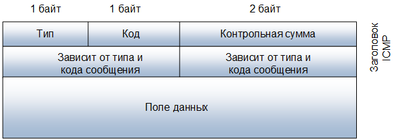


Рисунок 3. Структура ICMP пакета

**Type: 0 (Echo (ping) reply)** – тип (8 – echo (ping) request)

**Code:** **0** – числовой идентификатор, более точно определяющий тип ошибки.

**Checksum: 0xa3cc [correct]** – контрольная сумма

**Identifier (BE): 3907 (0x0f43)** – идентификатор (формат Big Endian)

**Identifier (LE): 17167 (0x430f)** – идентификатор (формат Little Endian – обратный порядок байт)

**Sequence number (BE): 2 (0x0002)** – порядковый номер (формат Big Endian)

**Sequence number (LE): 512 (0x0200)** – порядковый номер (формат Little Endian – обратный порядок байт)

**Response time: 10,613 ms** – время отклика

**Timestamp from icmp data: Oct 21, 2015 21:00:44.901728000 MSK** – отметка от ICMP данных – дата и время

**Timestamp from icmp data (relative): 0.010639000 seconds** – отметка по данным ICMP(Относительная) – смещение по времени

1. **Internet Protocol Version 4, Src: 93.158.134.3 (93.158.134.3), Dst: 10.0.2.15 (10.0.2.15)**

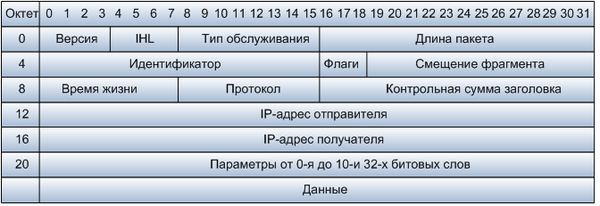


Рисунок 4. Структура IP-пакета

**Version: 4** – версия протокола

**Header length: 20 bytes** – длина заголовка. IP-пакета занимает 4 бит и указывает значение длины заголовка, измеренное в 32-битовых словах. Обычно заголовок имеет длину в 20 байт (пять 32-битовых слов).

**Total Length: 84** – общая длина. Занимает 2 байта и означает общую длину пакета с учетом заголовка и поля данных. Максимальная длина пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65 535 байт

**Identification: 0x2690 (9872)** – Идентификатор пакета. Занимает 2 байта и используется для распознавания пакетов, образовавшихся путем фрагментации исходного пакета. Все фрагменты должны иметь одинаковое значение этого поля.

**Flags: 0x00** – флаги. Поле занимает 3 бита и содержит признаки, связанные с фрагментацией. Установленный бит DF (Do not Fragment) запрещает маршрутизатору фрагментировать данный пакет, а установленный бит MF (More Fragments) говорит о том, что данный пакет является промежуточным (не последним) фрагментом. Оставшийся бит зарезервирован.

**Fragment offset: 0** - смещение фрагмента. Задает смещение в байтах поля данных этого пакета от начала общего поля данных исходного пакета, подвергнутого фрагментации.

**Time to live: 55** – время жизни. Занимает один байт и означает предельный срок, в течение которого пакет может перемещаться по сети. Время жизни данного пакета измеряется в секундах и задается источником передачи.

**Protocol: ICMP (1)** - Протокол верхнего уровня*. У*казывает, какому протоколу верхнего уровня принадлежит информация, размещенная в поле данных пакета

**Source: 93.158.134.3 (93.158.134.3)** – ip отправителя (хост, который получал ping –запрос)

**Destination: 10.0.2.15 (10.0.2.15)** – ip получателя (компьютер пользователя)

**Data (48 bytes)** – данные тестового пакета

1. **Ethernet II, Src: RealtekU\_12:35:02 (52:54:00:12:35:02), Dst: CadmusCo\_79:39:ff (08:00:27:79:39:ff)**

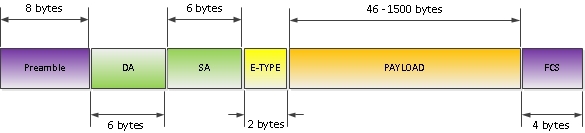


Рисунок 5. Структура кадра Ethernet ll

**Preamble**– последовательность бит, по сути, не являющаяся частью ETH заголовка определяющая начало Ethernet фрейма.

**Destination: CadmusCo\_79:39:ff (08:00:27:79:39:ff)** - MAC адрес назначения, может быть юникастом, мультикастом, бродкастом

**Source: RealtekU\_12:35:02 (52:54:00:12:35:02)** - MAC адрес отправителя. Всегда юникаст.

**Type: IP (0x0800)** - Идентифицирует протокол верхнего уровня (примеру 0x0800 – Ipv4).