Содержание

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

DHCP – Dynamic Host Configuration Protocol

IP - Internet Protocol

GNS3 - Graphical Network Simulator-3

ПО - Программное обеспечение

OSI - Open Systems Interconnection

RFS - Request for Comments

UDP - User Datagram Protocol

MITM - Man in the middle

NAT - Network Address Translation

ВВЕДЕНИЕ

С приходом компьютерных технологий, у людей появилась возможность быстро общаться через сеть и передавать разные данные, не ожидая личной встречи или, пока курьер доставит письмо. Для общения устройств в сети между собой, требуются их сетевые идентификаторы – IP-адреса. Есть 2 способа устройству его получить. Первый – специальный человек (системный администратор) вручную задаёт каждому устройству свой собственный IP-адрес. Такой IP-адрес называется статическим. У этого подхода есть ряд минусов. С ним можно работать если у вас в сети немного устройств, но что, если у вас целый офис. При ручном распределении IP-адресов в подобных сетях системный администратор может ошибиться, а искать подобные ошибки бывает трудно и долго, что может выйти очень дорого для компании. Тут и приходит на помощь второй способ - Dynamic Host Configuration Protocol, который автоматически распределяет IP-адреса в сети. Поэтому данный протокол очень важен и нужно уметь защищаться от атак на него.

Целью моей курсовой работы является изучение DHCP, а также изучение атаки с ложным DHCP сервером «Rogue DHCP Server» и защиты от неё с применением графического симулятора сети GNS3. Также выделены следующие задачи для достижения заданной цели:

- Изучить DHCP
- Узнать принципы и произвести «Rogue DHCP Server» атаку
- Узнать и применить принципы защиты от «Rogue DHCP Server» атаки

Первый раздел курсовой работы повествует об общих сведениях курсовой работы. Второй раздел описывает DHCP, его строение и работу в сети. В третьем разделе описываются принципы, а также реализуется «Rogue DHCP Server» атака и защита от неё.

1. Общие сведения

1.1. Цель работы

Исследовать DHCP, его взаимодействие с устройствами. Изучить и реализовать «Rogue DHCP Server» атаку и защиту от неё.

1.2. Задачи работы

Задачи работы представлены следующими пунктами:

1. Рассмотреть общую информацию о DHCP
2. Рассмотреть строение DHCP-сообщения
3. Рассмотреть опции DHCP
4. Рассмотреть принципы работы DHCP в сети
5. Рассмотреть общую информацию о «Rogue DHCP Server» атаке
6. Создать тестовую локальную сеть в GNS3 для проведения атаки
7. Произвести «Rogue DHCP Server» атаку
8. Рассмотреть последствия «Rogue DHCP Server» атаки
9. Рассмотреть принципы защиты от «Rogue DHCP Server» атаки
10. Реализовать защиту от «Rogue DHCP Server» атаки

1.3. Технологии и ПО, с помощью которых исследован DHCP, исследована и реализована «Rogue DHCP Server» атака

Для исследования DHCP и исследования, реализации и защиты от «Rogue DHCP Server» атаки использовалось система графической симуляции сети GNS3, ПО для виртуализации операционных систем VirtualBox с установленными системами «Windows 10» и «Kali Linux», ПО «Yersinia» и «Ettercap» для реализации атаки и ПО «Wireshark» для анализа сетевого траффика.

2. Исследование DHCP

2.1. Общая информация о DHCP

Dynamic Host Configuration Protocol (DHCP) - сетевой протокол, позволяющий сетевым устройствам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети. Существуют разногласия по тому, к какому уровню в моделях OSI и TCP/IP причислить данный протокол. Первый подход - причислять протокол к 3 уровню модели OSI или к 2 модели TCP/IP (Сетевой), так как протокол исполняет вспомогательные сетевые функции. Такое представление, например, можно найти в книге Таненбаума [4, с 498]. Второй подход – причисление к 7 уровню модели OSI или к 4 в модели TCP/IP (Прикладной), так как при инкапсуляции DHCP-сообщение входит в состав UDP дейтаграммы (Рисунок 2.1). Я буду придерживаться второго подхода.

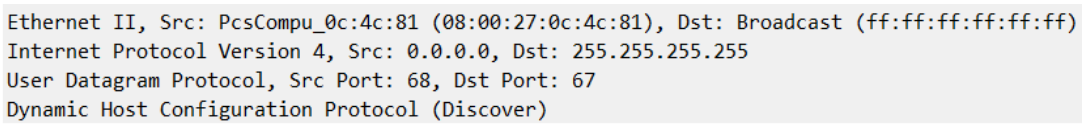

Рисунок 2.1 – Инкапсуляция протоколов.

Протокол был предложен в 1990 году. Настоящая версия протокола от 1997 года описана в документе RFC 2131 [1]. Так же существует новая версия DHCP, принятая в июле 2003 года, предназначенная для использования в среде IPv6, носит название DHCPv6 и определена в RFC 3315. Я же в работе буду рассматривать DHCP в реализации для IPv4.

DHCP является расширением BOOTP (bootstrap protocol), использовавшегося ранее для обеспечения бездисковых рабочих станций IP-адресами при их загрузке, и сохраняет с ним обратную совместимость. Главные отличия BOOTP и DHCP представлены в таблице 1.

Таблица 1 – Сравнение DHCP и BOOTP

Основа для сравнения	**BOOTP**	**DHCP**
Назначение	Предназначен для настройки бездисковых рабочих станций с ограниченными возможностями загрузки	Предназначен для настройки параметров протоколов стека TCP/IP часто перемещаемых сетевых компьютеров (допустим,

		портативных), которые имеют жесткие диски и полные возможности загрузки
Этапность настройки	Описывает процесс настройки загрузки, состоящий из двух фаз, следующим образом. Клиенты связываются с BOOTP-серверами для определения адреса и выбора файла загрузки. Клиенты связываются с серверами TFTP (Trivial File Transfer Protocol) для передачи файла образа загрузки	Описывает однофазный процесс настройки загрузки, посредством которого DHCP-клиент "договаривается" с DHCP-сервером об определении IP-адреса и получает другие подробности начальной настройки, необходимые для работы в сети
Обновление IP-адреса	BOOTP-клиенты не выполняют операцию получения нового адреса и не обновляют параметры настройки с помощью BOOTP-сервера, за исключением перезапуска системы	DHCP-клиенты не требуют перезапуска системы для получения адреса или обновления параметров настройки с помощью DHCP-сервера. Клиенты в установленные интервалы времени автоматически переходят в состояние обновления арендуемых адресов с помощью DHCP-сервера. Этот процесс происходит в фоновом режиме.

Работает DHCP по модели клиент-сервер: клиент - любое IP-устройство, подключаемое к сети, запрашивающее IP-адрес и параметры конфигурации сети от DHCP-сервера; сервер – компьютер, который предоставляет IP-адрес, данные о сети и ведёт таблицу выделенных IP адресов, чтобы избежать дублирования. Клиент и сервер обмениваются сообщениями DHCP в режиме запрос-ответ. DHCP работает при помощи транспортного протокола UDP (так как UDP не устанавливает жёсткого соединения, а значит может использовать широковещательный запросы, в отличие, например, от «Transmission Control Protocol», который требует установки соединения между устройствами) и использует 67 порт сервера и 68 порт клиента по умолчанию.

DHCP, помимо выделения динамического IP-адреса, поддерживает так же выделение статического IP-адреса для определённого MAC-адреса, если требуется, что бы у определённого клиента всегда был определённый IP-адрес (например, другой сервер), в независимости от состояния самого клиента.

Сервер распределяет адреса из специального, задаваемого системным администратором, диапазона (списка) IP-адресов, называемого пулом адресов. Так же протокол следит за их уникальностью: один IP-адрес может быть только

у одного MAC-адреса. Поскольку клиент может находится в сети временно, для него не корректно выделять постоянный IP-адрес, поэтому IP-адреса выделяются на определённое время, называемое временем аренды. Чем чаще в сети меняются клиенты, тем на более меньшее время стоит выделять IP-адрес. Например, в общественной сети время аренды может равняться часу, а в офисной сети – несколько дней. После окончания аренды, если клиент не продлевает её, IP-адрес освобождается, и его может повторно использовать другой клиент.

2.2. DHCP-сообщение

Представим DHCP-сообщение в виде таблицы (Таблица 2). В полях таблицы прописаны, сверху деление полей на байты и слева – с какого байта поле начинается.

Таблица 2 – Вид DHCP-сообщения

Байты	**1**	**2**	**3**	**4**
0	Op	Htype	Hlen	Hops
4	Xid			
8	Secs		Flags	
12	Ciaddr			
16	Yiaddr			
20	Siaddr			
24	Giaddr			
28	Chaddr			
44	Sname			
108	File			
236+	Options			

Разберём все поля:

Op (opcode, 1 байт) – Код операции сообщения / тип сообщения, 1 = BOOTREQUEST (запрос от клиента к серверу), 2 = BOOTREPLY (ответ от

сервера к клиенту). Допустимые значения этого поля определены в RFC 1700 [3, с 162].

Htype (hardware type, 1 байт) – Тип адреса на канальном уровне. Например, для Ethernet (10Mb) это 1. Допустимые значения этого поля определены в RFC 1700 [3, с 162].

Hlen (hardware length, 1 байт) – Длина аппаратного адреса в байтах.

Hops (1 байт) – Количество промежуточных маршрутизаторов (relay-агентов), которые находятся на пути между клиентом и сервером.

Xid (transaction id, 4 байта) – Идентификатор транзакции, случайное число, генерируемое клиентом.

Secs (seconds elapsed, 2 байта) – Время в секундах с начала процесса получения или обновления адреса, указываемое клиентом.

Flags (2 байта) – Поле для флагов или специальных параметров протокола DHCP (первый бит маркирует широковещательные сообщения).

Ciaddr (client ip-address, 4 байта) – Адрес, который запрашивает клиент, если тот у него уже есть. Используется, например, при продлении аренды адреса.

Yiaddr (your ip-address, 4 байта) – Адрес, который хочет предложить DHCP-сервер клиенту.

Siaddr (server ip-address, 4 байта) – IP-адрес следующего сервера DHCP. Например, для бездисковых рабочих станций для взятия образа операционной системы. Если этого не требуется, то DHCP сервер может вписать туда свой адрес.

Giaddr (gateway ip-address, 4 байта) – IP-адрес Relay-агента.

Chaddr (client hardware address, 16 байт) – Адрес клиента канального уровня. Например, MAC-адрес для Ethernet.

Sname (server host name, 64 байта) – Имя хоста DHCP сервера, если оно есть.

File (boot file, 128 байт) – Указатель для бездисковых рабочих станций о том, как называется файл на сервере, который следует использовать для загрузки, если он есть.

Options (переменная длинна) – Дополнительные параметры.

Разберём более подробно поле «options». Все опции делятся на стандартные и не стандартные (поддерживаются не всеми реализациями DHCP-сервера). Список стандартных опций прописан в RFS 2132 [2]. Каждая опция имеет уникальный код от 0 до 255.

Поле опций имеет переменную длину, однако DHCP-клиент должен быть готов принять DHCP-сообщение с длиной до 576 байт (поле options имеет минимальную длину в 312 байт) [1, с 9]. Но клиенты могут согласовать большую длину с помощью опции 57 «maximum DHCP message size».

Поле начинается с фиксированной последовательности, называемой «Magic Cookie». Это четыре байта со значениями 99, 130, 83, 99. По этой последовательности устройство понимает, что фиксированная часть пакета закончилась и начались DHCP опции.

Заканчивается поле всегда опцией с кодом 255.

Сами опции, за исключением 0 и 255 имеют следующий вид: первый байт опции – код. Затем идёт байт длинны данной опции (начиная с байта после байта длинны) в байтах. Далее идёт сама информация опции.

В таблице 3 представлены некоторые опции, которые чаще всего используются.

Таблица 3 – Некоторые возможные опции.

Код	**Название**	**Длина опции в байтах**	**Описание**
1	Маска подсети	4	Маска подсети.
3	Маршрутизаторы	N	IP-адреса доступных шлюзов по умолчанию.
6	DNS сервер	N	IP-адреса доступных DNS серверов.
12	Имя хоста	N	Имя хоста.
50	Требуемый IP	4	Запрос IP-адреса
53	Типы DHCP-сообщения	1	Эта опция используется для передачи типа сообщения DHCP. Существует 8 типов сообщений: 1. DHCPDISCOVER 2. DHCPOFFER 3. DHCPREQUEST 4. DHCPDECLINE

			5. DHCPACK 6. DHCPNAK 7. DHCPRELEASE 8. DHCPINFORM
55	Список запросов параметров	N	Используется DHCP-клиентом для запроса значений для указанных параметры конфигурации. Список запрашиваемых параметров указывается как n байтов, где каждый байт является допустимым кодом опции DHCP.
255	Конечная опция	0	Помечает о окончании передачи поля опций.

2.3. Принципы работы DHCP в сети

Разберёмся, как работает DHCP в сети. Для этого посмотрим, как происходит коммуникация между сервером и клиентом. Смотреть мы будем работу сети в программе GNS3 при помощи ПО «Wireshark».

2.3.1. Подключение к сети

Когда клиент подключается к новой сети, он ничего не знает о ней, в том числе своего IP-адреса в ней. У него есть только MAC-адрес, который вшит в сетевую плату устройства. Поэтому, при подключении к сети, компьютеру требуется получить свой собственный IP-адрес в этой сети. Для этого, клиент отправляет широковещательное (broadcast) сообщение в сеть. Данная процедура в опции 53 помечается, как DHCPDISCOVER. Так же в этом сообщении клиент помечает информацию о себе и запрашиваемую информацию о сети. Если клиент уже был недавно в сети, то он может запросить использовавшийся ранее IP-адрес (Рисунок 2.2).

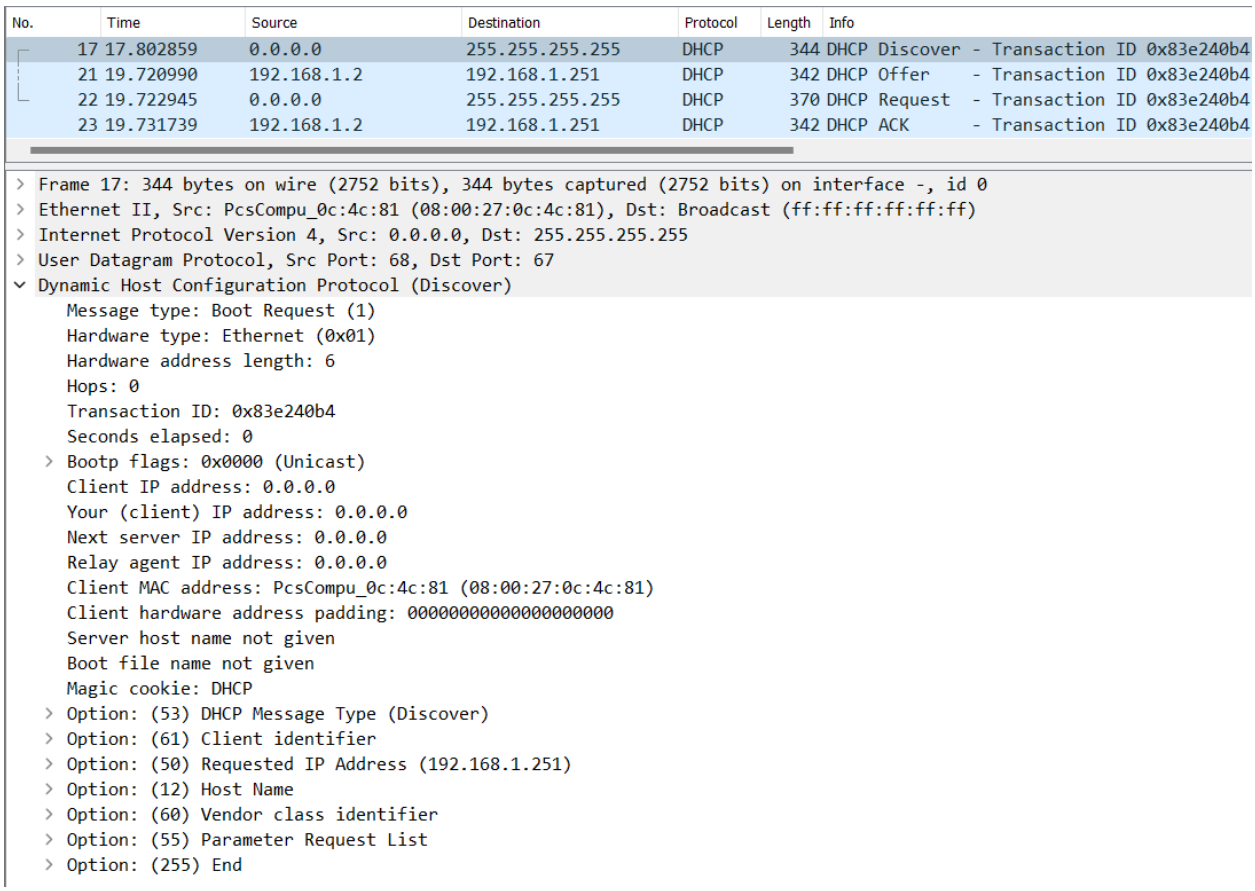

Рисунок 2.2 – DHCPDISCOVER.

Данный запрос получают все компьютеры в сети, но обрабатывает его только DHCP сервер. После обработки, если был найден свободный IP-адрес в пуле адресов сервера, он высылает широковещательный ответ с предлагаемым адресом. При этом DHCP сервер не обязан резервировать на данном этапе IP-адрес для клиента, но чаще всего DHCP сервера так делают – резервируют адрес на небольшое время, так как это делает протокол более эффективным [1, с 12]. В противном случает, при подключении сразу 2-х и более клиентов им мог быть предложен один и тот же IP-адрес. Его бы смог принят только один, а остальные повторили бы процедуру подключения к сети. Данное сообщение называется DHCPOFFER (Рисунок 2.3). Так как у клиента ещё нет IP-адреса, то сервер отправляет сообщение по MAC адресу. Если запрашиваемый клиентом адрес свободен, то будет отправлен именно данный адрес. Так же сервер отправляет всю доступную запрашиваемую и необходимую для работы в сети информацию.

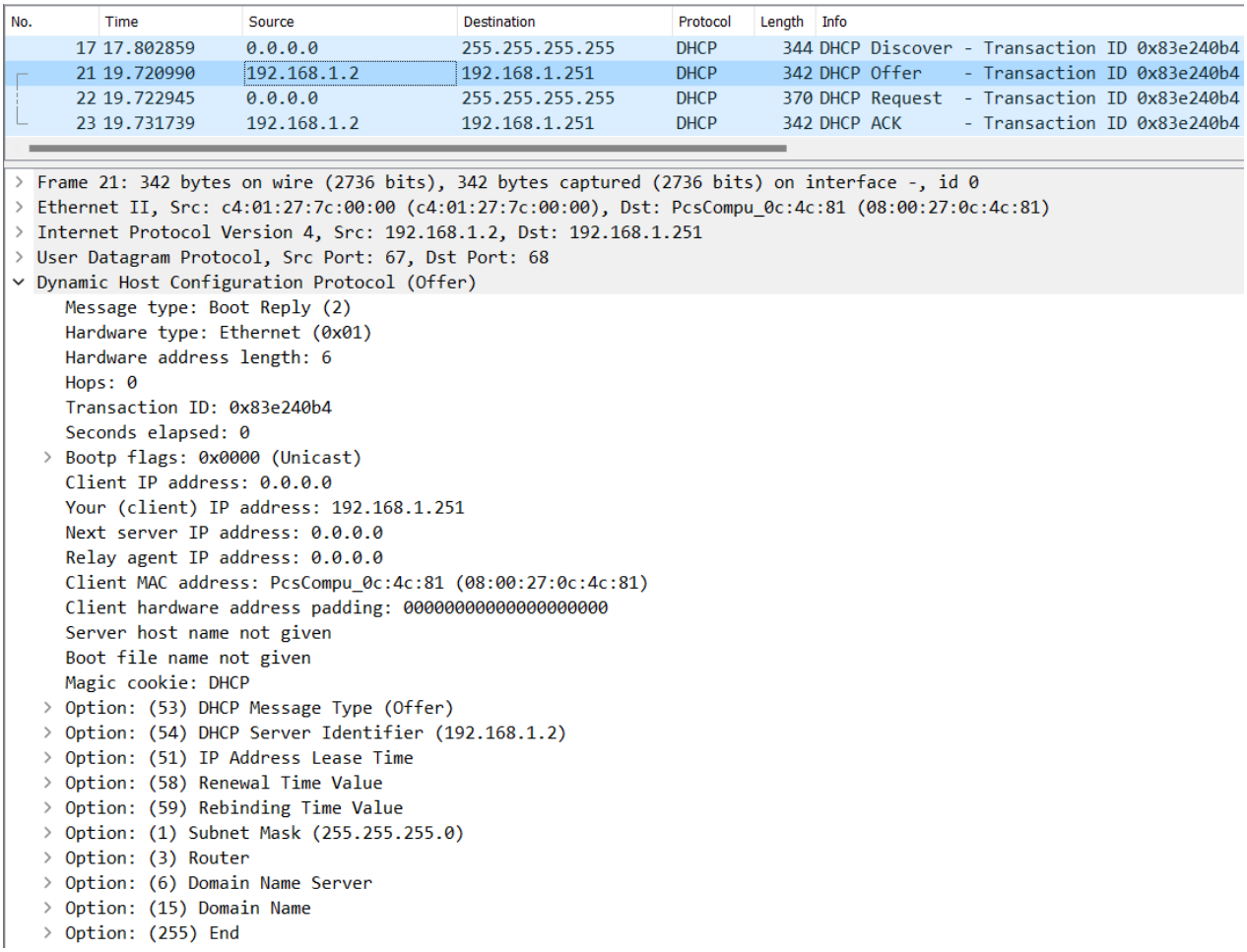

Рисунок 2.3 – DHCPOFFER.

Но, полученный IP-адрес клиент ещё не может использовать. Допустим, в сети не один, а два DHCP сервера. Каждый из них получит запрос и каждый даст ответ клиенту. Таким образом, клиент зарезервирует сразу 2 IP-адреса. Так же IP-адрес может исказиться при передаче, например, на физическом уровне. Что бы избежать проблем, существует процедура подтверждения резервирования IP-адреса. До её завершения клиент не может использовать выданный адрес.

Клиент выбирает один из предложенных IP-адресов и высылает широковещательное сообщение DHCPREQUEST (Рисунок 2.4). В нём клиент помечает выбранный адрес и сервер, который его выдал. После получения, DHCP сервера проверяют, они ли предложили данный адрес. Если нет, то для них это означает отказ от предложенного ими адреса и процесс резервирования для них заканчивается.

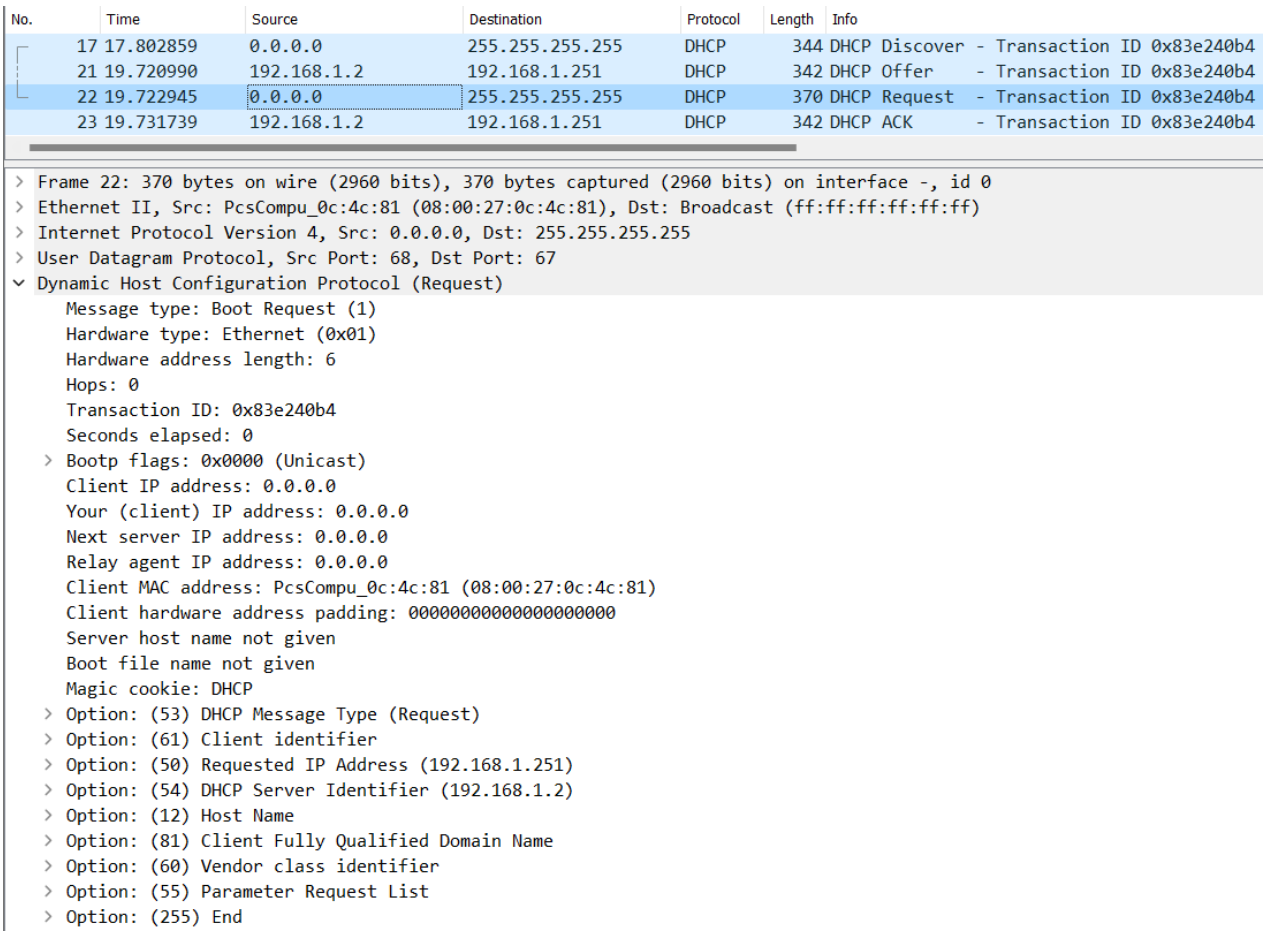

Рисунок 2.4 – DHCPREQUEST.

Сервер, который выдал адрес, фиксирует привязку к клиенту в постоянном хранилище и отправляет сообщение DHCPACK, если все данные верны (Рисунок 2.5). После этого клиент может пользоваться выданным IP-адресом. Или сообщение DHCPNAK, если обнаружена ошибка (например, данный IP-адрес уже назначен). После этого сообщения, через некоторое время, клиент повторит всю процедуру подключения к сети. После получения DHCPACK от сервера, клиенту следует проверить то, что он является единоличным владельцем данного адреса [1, с 15].

Всю процедуру подключения к сети, описанную выше, часто называют DORA, по первым буквам сообщений DISCOVER, OFFER, REQUEST, ACKNOLIGMENT.

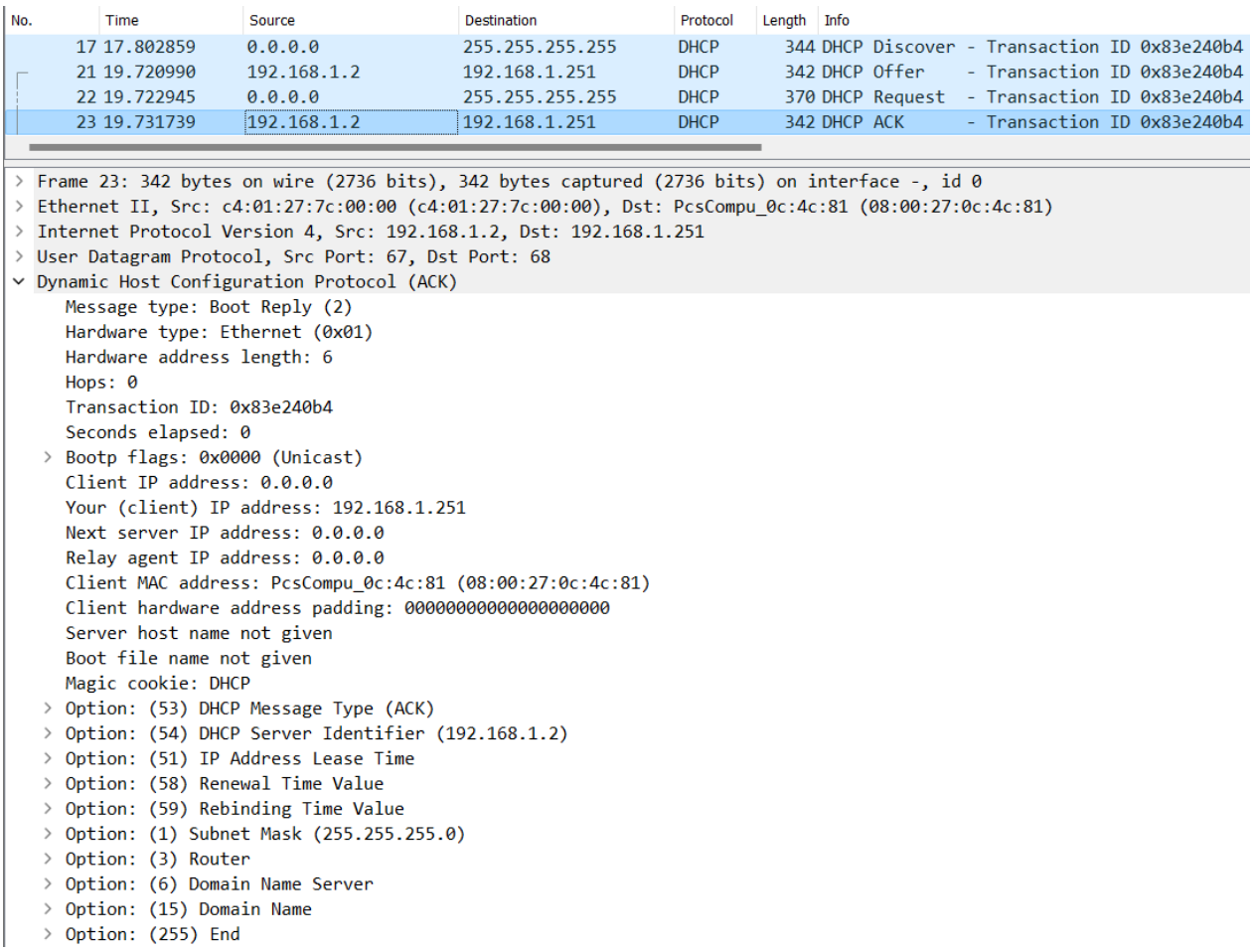

Рисунок 2.5 – DHCPACK.

Клиенты повторяют передачу DHCPDISCOVER и DHCPREQUEST по тайм-ауту, если ответа от сервера не поступает. Делают они это достаточное количество раз, чтобы обеспечить достаточную вероятность конфликта с сервером, при этом не ожидая очень долго [1, с 16].

Если в сети работают 2 DHCP сервера и оба задействованы в сети, то не стоит выделать для них один и тот же пул IP-адресов. Иначе может возникнуть конфликт адресов, как показано на рисунке 2.6.

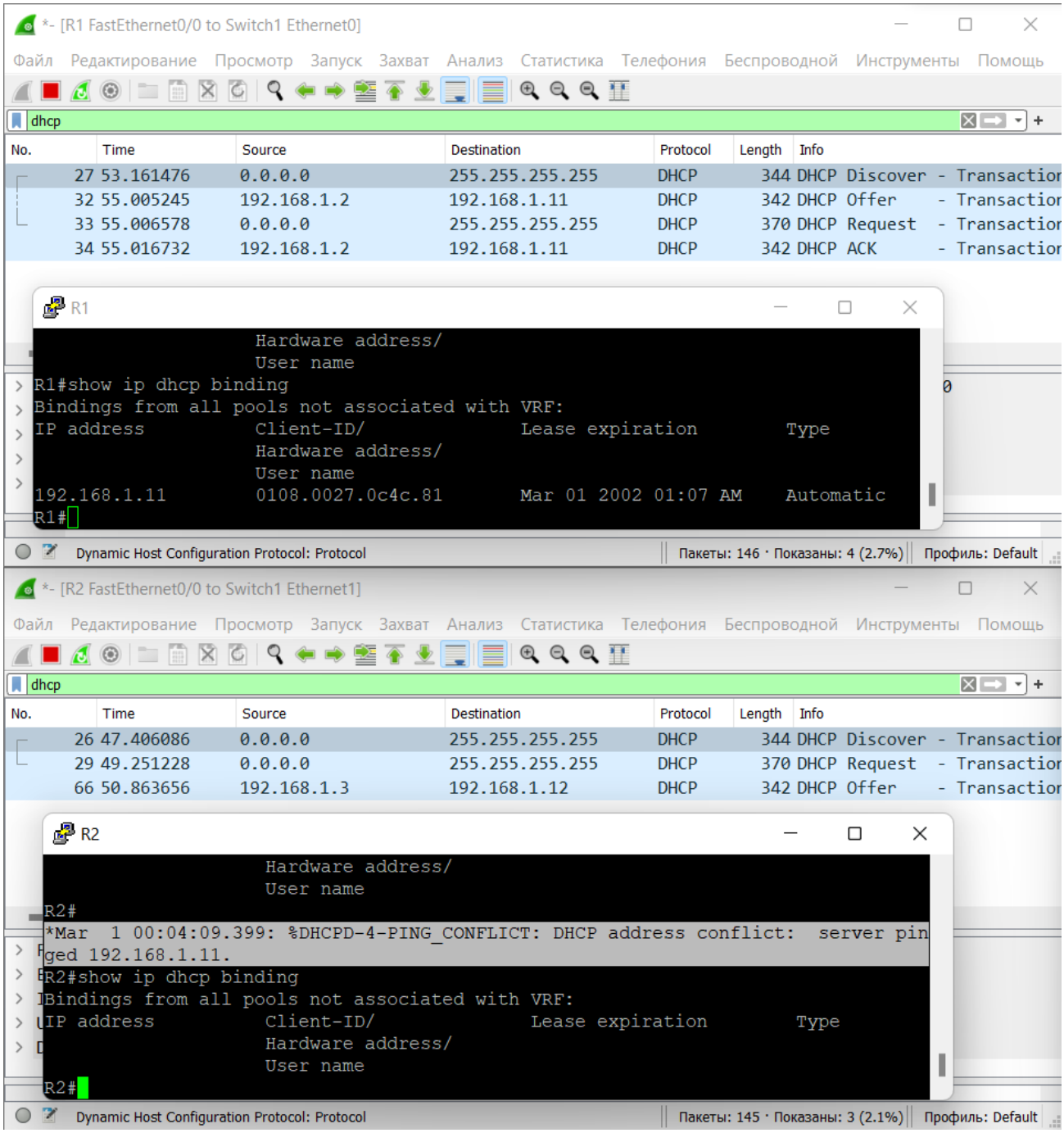

Рисунок 2.6 – Конфликт выделения IP-адреса.

2.3.2. Аренда IP-адреса

IP-адреса, чаще всего, выдаются на определённый срок аренды (lease time). Это время в секундах, оно относительно - часы могут быть не синхронизированы у пользователя с сервером. Время указывается 32-х битовым словом, а поэтому время аренды может быть от 0 и, приблизительно, до 100 лет. Значение с 32-мя единицами зарезервировано для представления бесконечности. Аренду можно продлить, при этом процедура продления проходит в более простой форме.

Клиент делает первую попытку продления по истечении половины срока аренды. Делает он это запросом DHCPREQUEST при помощи unicast сообщения, то есть обратившись напрямую к DHCP серверу, который выдал IP-адрес. Сервер же высылает сообщение DHCPACK. Если ответа не поступит, то через половину от оставшегося времени клиент отправит в сеть широковещательное сообщение DHCPREQUEST. Например, на рисунке 2.7 можно увидеть, как в сообщениях под номерами 246 – 247 идёт продление аренды адреса. После сообщения 247 я отключил DHCP сервер. Видно, как в сообщениях 385 – 394 клиент пытается продлит IP-адрес, обратившись напрямую к DHCP серверу, который его выдал. После этого, через половину от оставшегося времени он высылает запрос продления на широковещательный адрес и повторяет это. Если ответа не поступит, то клиент обязан прекратить использовать полученный в данной сети IP-адрес. Для подключения к сети ему надо снова получить IP-адрес.

No.	Time	Source	Destination	Protocol	Length	Info
246	319.934095	192.168.1.251	192.168.1.2	DHCP	358	DHCP Request - Transaction ID 0xf3c9ce3
247	319.944886	192.168.1.2	192.168.1.251	DHCP	342	DHCP ACK - Transaction ID 0xf3c9ce3
385	619.975285	192.168.1.251	192.168.1.2	DHCP	358	DHCP Request - Transaction ID 0xcb089591
387	621.976643	192.168.1.251	192.168.1.2	DHCP	358	DHCP Request - Transaction ID 0xcb089591
390	623.977534	192.168.1.251	192.168.1.2	DHCP	358	DHCP Request - Transaction ID 0xcb089591
394	626.977498	192.168.1.251	192.168.1.2	DHCP	358	DHCP Request - Transaction ID 0xcb089591
485	846.250597	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x21aefe09
486	849.271386	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x21aefe09
487	851.271899	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x21aefe09
506	888.321870	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x941e9c55
507	890.348388	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x941e9c55
508	893.350250	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x941e9c55
509	897.350618	192.168.1.251	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0x941e9c55

Рисунок 2.7 – Продление аренды IP-адреса

Если в сети есть резервный сервер, то он может продлить аренду клиенту, когда тот начнёт посылать широковещательные запросы. В этом случае клиент получит DHCPACK, но сервер может и отказать клиенту в продлении, например, если у резервного сервера нет пула IP-адресов, с адресом клиента.

Если клиент отключается от сети и затем подключается к ней снова, при этом время аренды IP-адреса с прошлого подключения не истекло, то клиент начинает процедуру с сообщения DHCPREQUEST, запрашивая возможность продолжать использовать выданный ранее адрес.

Так же возможно принудительно завершить аренду IP-адреса со стороны клиента, отправив сообщение DHCPRELEASE. Со стороны сервера так сделать невозможно.

2.3.3. Другие значения опии 53

Если клиент обнаружил, что выданный IP-адрес уже используется в сети, то он посылает серверу сообщение DHCPDECLINE. Сервер сразу же помечает адрес как недоступный, а клиент, не менее чем через 10 секунд начинает заново процесс подключения к сети.

Так же клиент запросом DHCPINFORM может запросить необходимые конфигурационные параметры, кроме IP-адреса (так как подразумевается, что он на момент данного запроса уже есть у клиента).

Если IP-адрес клиент получает каким-то иным образом (например, ручная настройка), то при подключении к сети клиент может воспользоваться запросом DHCPINFORM и получить все актуальные параметры сети. Ответом высылается сообщение DHCPACK по указанному адресу в поле ciaddr. Сервер так же проверяет согласованность адреса, но не проверяет время аренды [1, с 20].

2.3.4. Relay агент

Предположим, что наша сеть состоит из нескольких подсетей, в каждой из которых не очень много устройств. Не рационально для каждой подсети создавать и настраивать свой собственный DHCP сервер. Лучше использовать 1 общий.

DHCP работает в пределах одной подсети, так как маршрутизаторы не пропускают широковещательный трафик. Для работы вне её рамок, на маршрутизаторы требуется установить специальные relay-агенты для передачи DHCP-сообщений.

Агент-ретранслятор DHCP (DHCP relay agent) передаёт сообщения DHCP между сервером и клиентами, когда они не находятся в одной подсети. Таким образом, в больших сетях, состоящих из многих подсетей, один DHCP-сервер может обслуживать всю сеть при помощи агентов-ретрансляторов, которые располагаются на граничных маршрутизаторах подсетей. В сети можно сконфигурировать до 400 агентов-ретрансляторов. Работают ретрансляторы следующим образом (Рисунок 2.8).

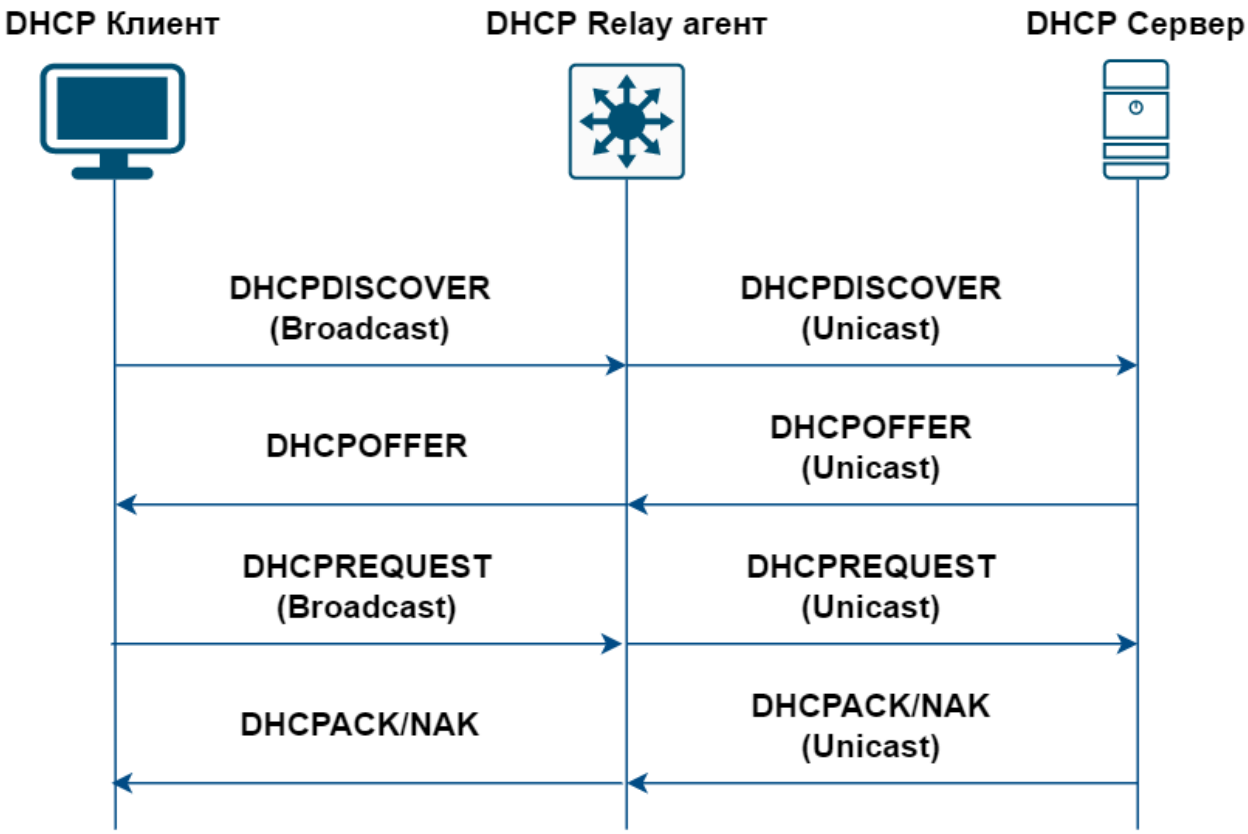

Рисунок 2.8 – Принцип работы relay агента

Relay агент принимает любой запрос к DHCP серверу из своей подсети и отправляет его напрямую DHCP серверу. Сервер же отправляет ответ напрямую relay агенту и тот, в зависимости от назначения DHCP-сообщения, отсылает его на широковещательный адрес или напрямую клиенту. Убедимся в этом, собрав простую сеть в GNS3 (Рисунок 2.9). Тут же мы можем заметить, что в поле relay агента появился IP-адрес агента, а количество пройденных relay агентов увеличилось на 1. (Рисунок 2.10).

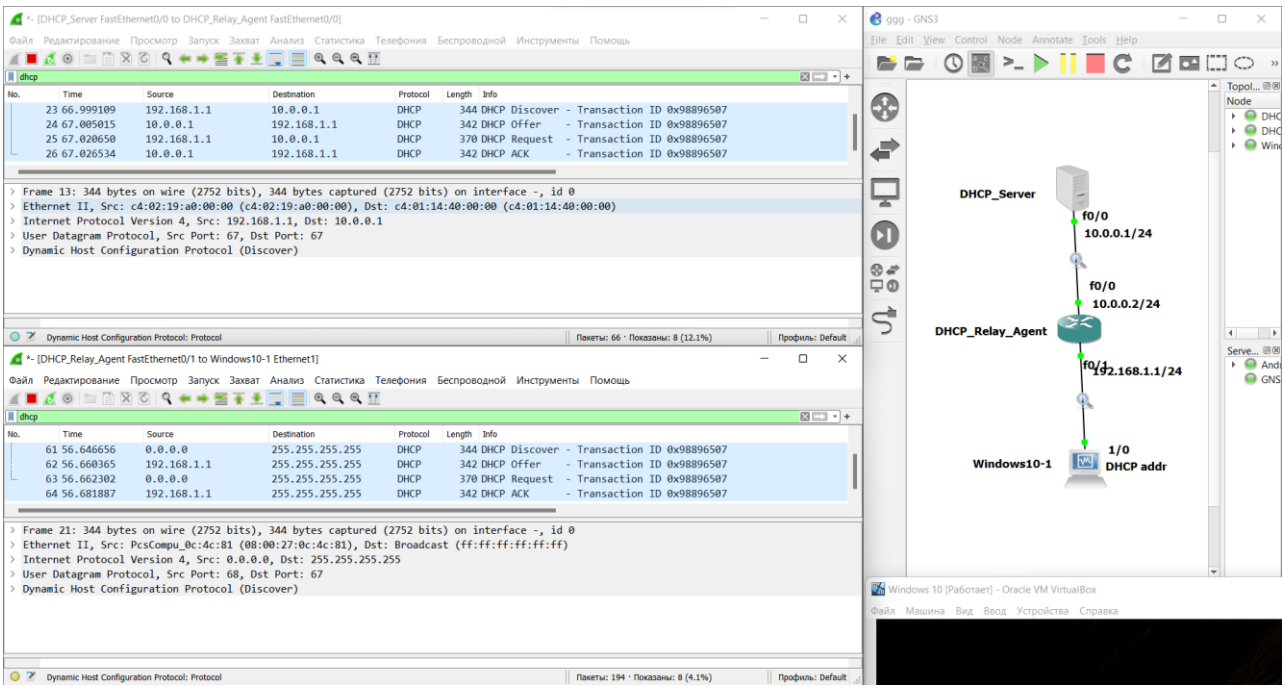

Рисунок 2.9 – Работа relay агента

No.	Time	Source	Destination	Protocol	Length	Info
23	66.999109	192.168.1.1	10.0.0.1	DHCP	344	DHCP Discover - Transaction ID 0x98896507
24	67.005015	10.0.0.1	192.168.1.1	DHCP	342	DHCP Offer - Transaction ID 0x98896507
25	67.020650	192.168.1.1	10.0.0.1	DHCP	370	DHCP Request - Transaction ID 0x98896507
26	67.026534	10.0.0.1	192.168.1.1	DHCP	342	DHCP ACK - Transaction ID 0x98896507

```
> Frame 23: 344 bytes on wire (2752 bits), 344 bytes captured (2752 bits) on interface -, id 0
> Ethernet II, Src: c4:02:19:a0:00:00 (c4:02:19:a0:00:00), Dst: c4:01:14:40:00:00 (c4:01:14:40:00:00)
> Internet Protocol Version 4, Src: 192.168.1.1, Dst: 10.0.0.1
> User Datagram Protocol, Src Port: 67, Dst Port: 67
v Dynamic Host Configuration Protocol (Discover)
    Message type: Boot Request (1)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 1
    Transaction ID: 0x98896507
  v Seconds elapsed: 12
      > [Expert Info (Note/Protocol): Seconds elapsed appears to be encoded as little-endian]
  > Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 192.168.1.1
```

Рисунок 2.10 – Работа relay агента

3. Исследование сетевой атаки «Rogue DHCP Server»

3.1. Общая информацию о «Rogue DHCP Server» атаке

Протокол DHCP очень важен и используется повсеместно. Но в нём есть большой недостаток. На многих этапах используются широковещательные сообщения, их получают все устройства в сети. Этим и могут воспользоваться злоумышленники. Они могут прикинутся DHCP сервером сами и произвести процедуру подключения к сети клиента, но в информации о сети указать другие параметры. Данная атака, с DHCP сервером злоумышленника в сети называется «Rogue DHCP Server», и она является платформой для более опасных атак. Указав свои данные, злоумышленник может видеть и анализировать весь трафик жертвы, который идёт на шлюзовой роутер (default-router). Это называется «Man-in-the-Middle» атака (MITM).

Обычно злоумышленники не производят атаку «Rogue DHCP Server» стразу. Если к сети просто подключится другой DHCP сервер, то получится ситуация, когда в сети 2 DHCP сервера: один авторизованный, другой – злоумышленника. Запросы о подключении к сети будут приходить обоим серверам и оба ответят на них, как уже разбиралось ранее, но клиент, обычно, выбирает тот ответ, что пришёл раньше или сервер, с которым он раннее уже общался. Чаще всего это и будет авторизованный DHCP сервер, так как к нему, обычно, сообщения доходят быстрее, чем к злоумышленнику, и он быстрее их обрабатывает в силу спецификации. Что бы этого избежать, злоумышленники стараются избавиться от DHCP сервера. Вручную резать провода или ломать сам сервер – плохой, очень опасный и трудный вариант. Поэтому злоумышленники пытаются вывести сервер из строя другим способом – при помощи атаки «DHCP starvation».

«DHCP starvation» - атака, которая очень похожа на DOS атаку: злоумышленник отправляет множество запросов DHCPDISCOVER, меняя в каждом из них MAC-адрес. Это приводит к тому, что весь пул IP-адресов сервера будет исчерпан, и новые пользователи не смогут запросить у него IP-адрес, а сам

сервер будет занят обработкой запросов, из-за чего уже существующие пользователи не смогут продлить аренду. Другими словами, сервер будет отказывать в обслуживании.

После этого злоумышленник подключает свой DHCP сервер к сети. Эта атака уже называется «Rogue DHCP server» Теперь, после вывода из строя авторизованного DHCP сервера, все запросы на подключение к сети или продления аренды адреса будет получать только злоумышленник.

3.2. Локальная сеть в GNS3

Для организации атаки и защиты от неё создадим небольшую локальную сеть в программе GNS3 (Рисунок 3.1).

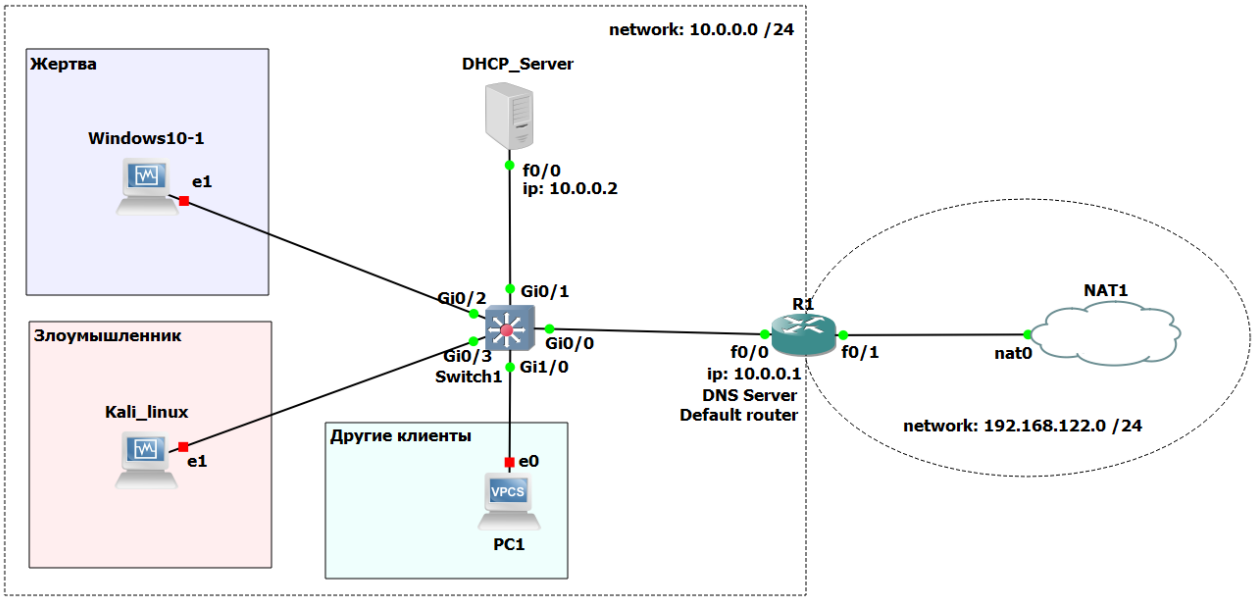

Рисунок 3.1 – Сеть для реализации атак и защиты

Здесь мы видим 2 сети:

1. С адресом 192.168.122.0 и маской 24. Это сеть для выхода в интернет через технологию NAT, позволяющую преобразовывать IP-адреса транзитных пакетов. Это позволяет устройствам с частным адресом IPv4 обращаться к ресурсам за пределами частной сети. Для этого используется NAT модуль (NAT1) и роутер R1, реализованный через роутер Cisco3745, который и занимается преобразованием адресов. Так же роутер R1 содержит в себе DNS сервер, что бы обращаться к сайтам не по IP-адресу, а по доменному имени.

2. С адресом 10.0.0.0 и маской 24. Это основная сеть тестирования, где присутствуют следующие узлы:

- DHCP_Server – Сервер DHCP, реализованный на базе роутера Cisco3745
- Switch1 – Коммутатор 2 уровня по модели OSI (канальный уровень)
- Windows10-1 – Виртуальная машина с системой Windows 10
- Kali_linux - Виртуальная машина с системой Linux, дистрибутивом Kali
- PC1 - Virtual PC Simulator — программа, написанная Полом Менгом, позволяющая имитировать легкий ПК с поддержкой DHCP и ping

Настройки DHCP_Server, Switch1 и R1 можно увидеть в приложениях 1, 2 и 3 соответственно.

3.3. Реализация «Rogue DHCP Server» атаки

Как уже писалось ранее, «Rogue DHCP Server» атака обычно не происходит в одиночку. Чаще всего, ей предшествует «DHCP starvation» атака. Поэтому я разберу их обе. Так же посмотрю на последствия данных атак, в частности - MITM атаку.

Для реализации атаки будет использовано 2 ПО на виртуальной машине с «Kali linux»:

1. «Yersinia» (версия 0.8.2) - фреймворк для выполнения атак 2-ого уровня (Канальный) модели OSI (1 по модели TCP/IP). Он предназначен для использования некоторых недостатков в различных сетевых протоколах. Я буду использовать это ПО для организации «DHCP starvation» атаки [5].
2. «Ettercap» (версия 0.8.3.1) - ПО с набором инструментов для атаки MITM. Он умеет прослушивать соединения, фильтровать на лету содержимое передаваемых данных, создавать ложный DHCP-сервер и многое другое. Я буду использовать это ПО для организации «Rogue DHCP Server» атаки и MITM атаки [6].

А также ПО для анализа сетевого траффика «Wireshark».

3.3.1. 1 этап - DHCP starvation

Для организации атаки «DHCP starvation» я подключаюсь на linux машине к сети. На рисунке 3.2 слева, в терминале видно, что DHCP сервер присвоил машине IP-адрес 10.0.0.17. Его и будет использовать эта машина в дальнейшем. После этого открываю графическую версию фреймворка «Yersinia», показанную на рисунке 3.2 в окне справа. В окне «Launch attack» выбираю вкладку «DHCP», выбираю пункт «sending DISCOVER packet» с установленным флажком «DoS», и запускаю атаку (Рисунок 3.3).

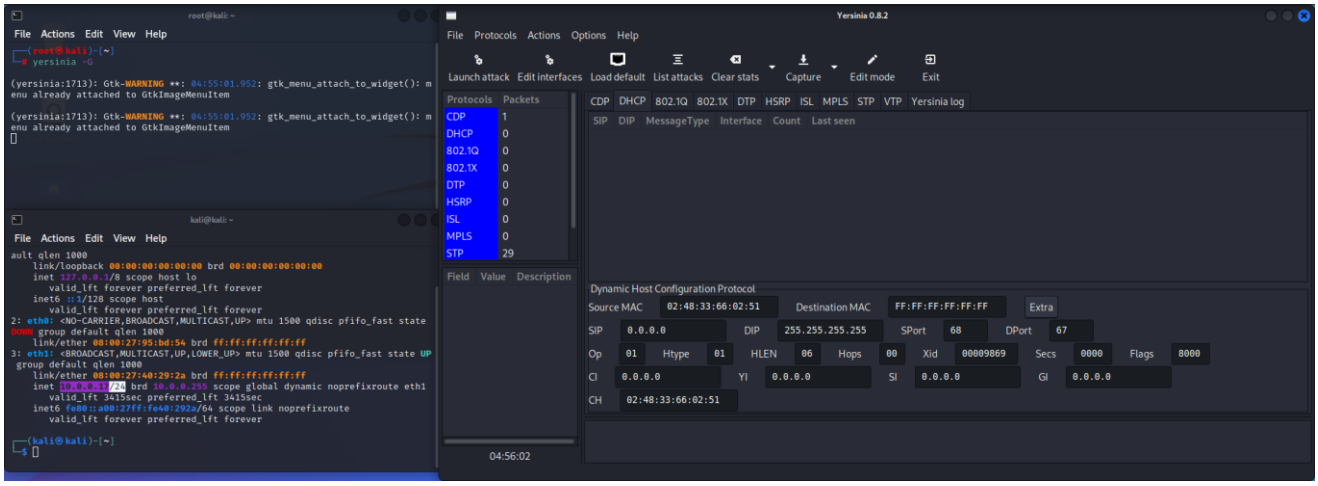

Рисунок 3.2 – Экран linux машины перед началом атаки

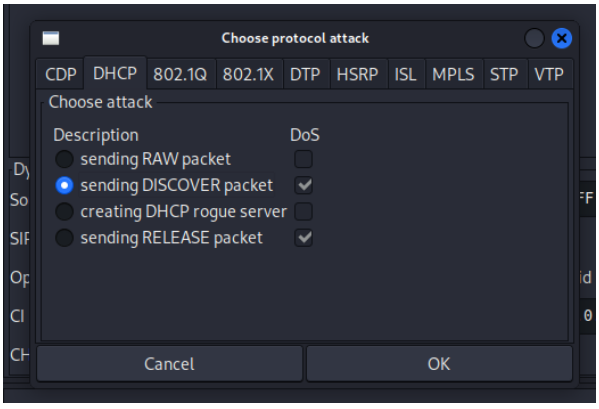

Рисунок 3.3 – Окно конфигурации атаки

В ходе атаки, в окне атаки фреймворка и по мониторингу сетевого подключения между коммутатором и сервером DHCP можно видеть отправку большого количества DHCPDISCOVER сообщений (Рисунок 3.4). Согласно фреймворку, было отослано более миллиона таких сообщений за несколько секунд работы. Из-за большого наплыва сообщений, у DHCP сервера временно

резервируются все IP-адреса в пуле адресов. Сервер просто не успевает обрабатывать всё. Пробую с PC1 получить IP-адрес, как это он сделал ранее. Он отсылает 3 DHCPDISCOVER сообщения, но, из-за атаки сервер не может ответить на них. PC1 не получает ответа от сервера, из-за чего выводит сообщение, что он не может найти DHCP сервер. Это показано на рисунке 3.4 в нижнем правом окне.

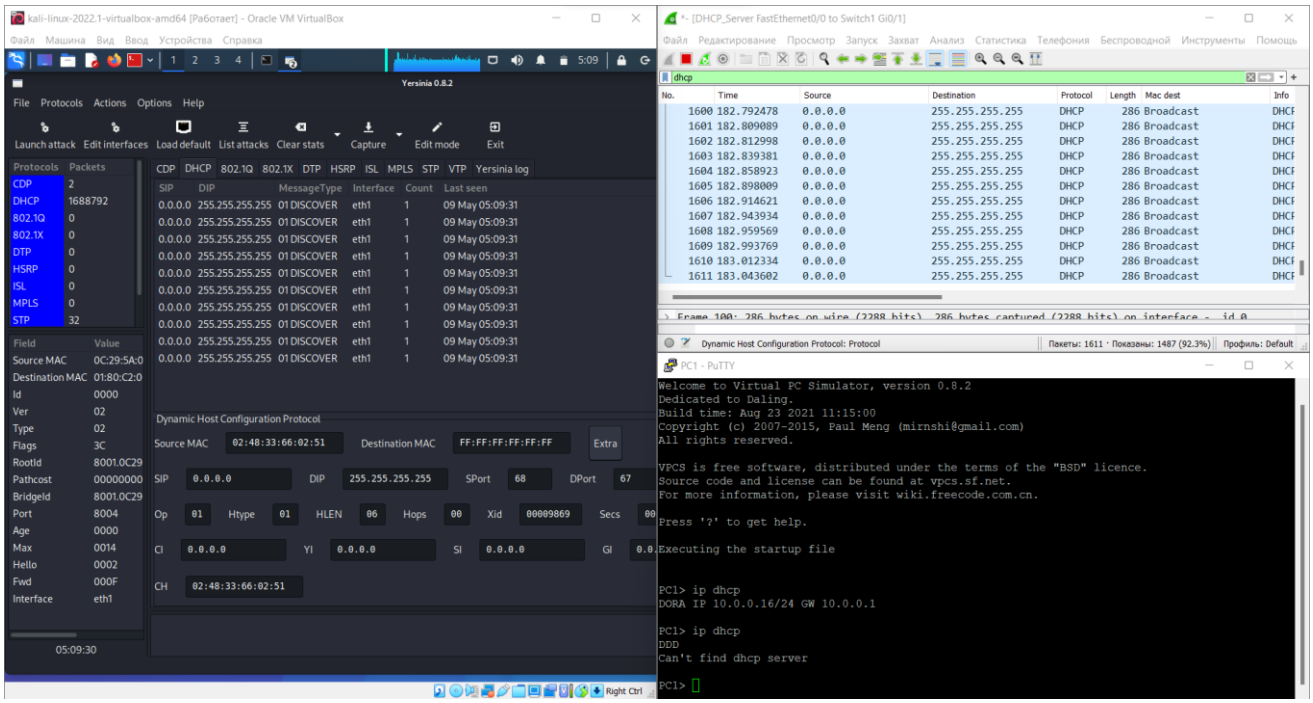

Рисунок 3.4 – «DHCP starvation» атака

Посмотрев на мониторинг сетевого подключения между коммутатором и сервером DHCP, можно увидеть, что каждый DHCPDISCOVER запрос, который участвовал в атаке имеет свой MAC-адрес, который был сгенерирован случайным образом фреймворком (Рисунок 3.5).

No.	Time	Source	Destination	Protocol	Length	Mac sourse	Info
3137	202.203523	0.0.0.0	255.255.255.255	DHCP	286	d1:ba:f4:52:6b:33	DHCP Discover
3138	202.209384	0.0.0.0	255.255.255.255	DHCP	286	41:0a:2a:6d:de:4b	DHCP Discover
3139	202.216238	0.0.0.0	255.255.255.255	DHCP	286	56:d8:91:1a:79:ec	DHCP Discover
3140	202.222087	0.0.0.0	255.255.255.255	DHCP	286	00:a3:05:2f:93:85	DHCP Discover
3141	202.227957	0.0.0.0	255.255.255.255	DHCP	286	3a:f8:52:1c:2e:34	DHCP Discover
3142	202.232853	0.0.0.0	255.255.255.255	DHCP	286	2c:8f:5a:51:05:0d	DHCP Discover
3143	202.236757	0.0.0.0	255.255.255.255	DHCP	286	bf:f9:ba:03:aa:74	DHCP Discover
3144	202.246515	0.0.0.0	255.255.255.255	DHCP	286	35:6f:e8:4e:11:ca	DHCP Discover
3145	202.252378	0.0.0.0	255.255.255.255	DHCP	286	dd:7f:9f:7d:c1:2c	DHCP Discover
3146	202.258249	0.0.0.0	255.255.255.255	DHCP	286	34:23:6b:55:30:89	DHCP Discover
3147	202.266066	0.0.0.0	255.255.255.255	DHCP	286	65:2e:45:47:a7:70	DHCP Discover
3148	202.271935	0.0.0.0	255.255.255.255	DHCP	286	3e:e6:fc:0f:81:ef	DHCP Discover
3149	202.279745	0.0.0.0	255.255.255.255	DHCP	286	21:66:8f:3d:86:69	DHCP Discover
3150	202.284631	0.0.0.0	255.255.255.255	DHCP	286	1f:e2:6b:14:d0:50	DHCP Discover
3151	202.290500	0.0.0.0	255.255.255.255	DHCP	286	60:8d:1a:38:ab:60	DHCP Discover

Рисунок 3.5 – Мониторинг сети во время атаки

3.3.2. 2 этап - Rogue DHCP Server

Теперь, когда авторизованный DHCP сервер недоступен, можно переходить ко 2 этапу атаки - «Rogue DHCP Server» атаке. На linux машине, после остановки «DHCP starvation» атаки, захожу в приложение «Ettercap» (Рисунок 3.6). Выбираю сетевой интерфейс, с помощью которого машина подключена к атакуемой сети и приложение начинает мониторить входящие пакеты, отбирая из них DHCP-сообщения. Во вкладке «MITM» выбираю пункт «DHCP spoofing» для создания ложного DHCP сервера (Рисунок 3.7). В открывшемся окне ввожу данные для этого сервера: в первом поле указываю пул адресов, во втором указываю маску сети, в третьем DNS сервер (Рисунок 3.8) и запускаю атаку (Рисунок 3.9).

Рисунок 3.6 – Приложение «Ettercap»

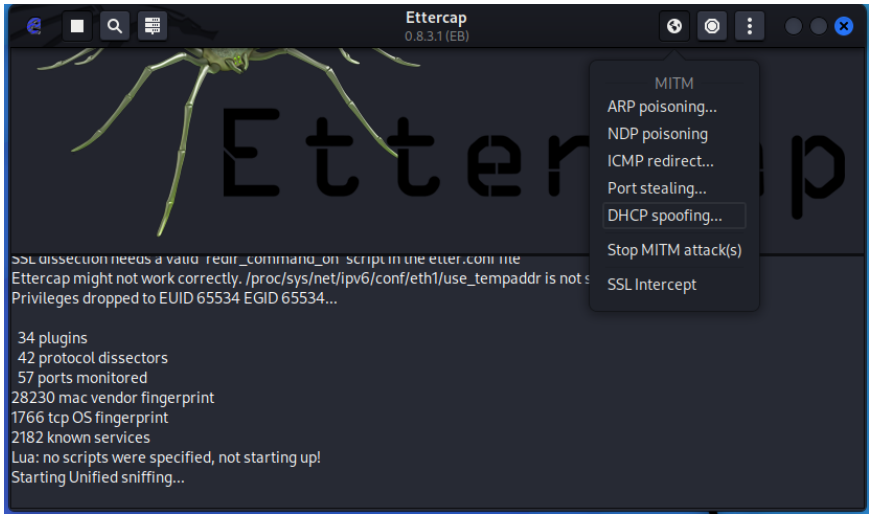

Рисунок 3.7 – Выбор атаки в приложении «Ettercap»

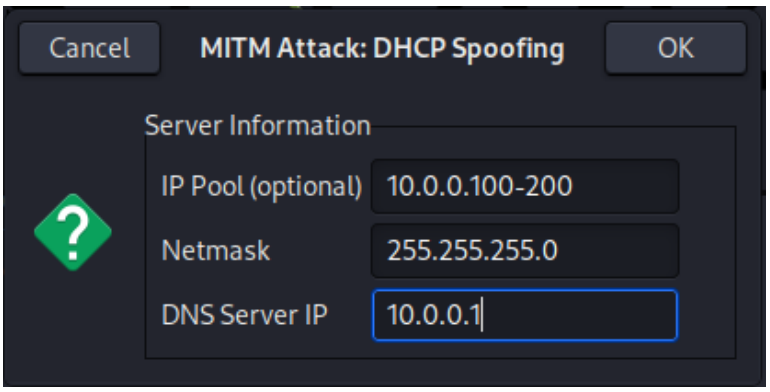

Рисунок 3.8 – Настройка атаки.

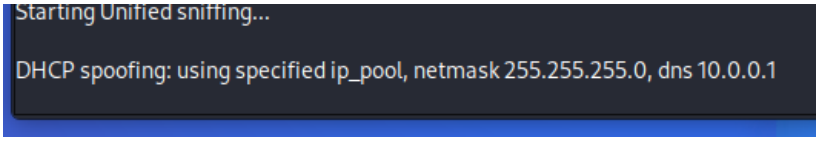

Рисунок 3.9 – Запуск ложного DHCP сервера

Если пользователь запросит адрес не из пула адресов, указанного на рисунке 3.8, то сервер его одобрит. Злоумышленнику не важно, как у пользователя IP, главное – изменить значение опции 3 (маршрутизаторы по умолчанию).

Далее я подключаю к сети windows машину. Эта машина уже была ранее в данной сети, поэтому она запрашивает IP-адрес, который ранее использовала: 10.0.0.50. Приложение «Ettercap» получает этот запрос, и, хоть этот адрес и не входит в пул заданных в приложении IP-адресов, приложение все равно его согласовывает, как это видно на рисунке 3.10 в левом окне. Но при этом, приложение ставит в опцию 3 свой IP-адрес, делая его шлюзом по умолчанию для данной windows машины, как это видно на рисунке 3.10 в правом окне. Таким образом была произведена атака «Rogue DHCP Server».

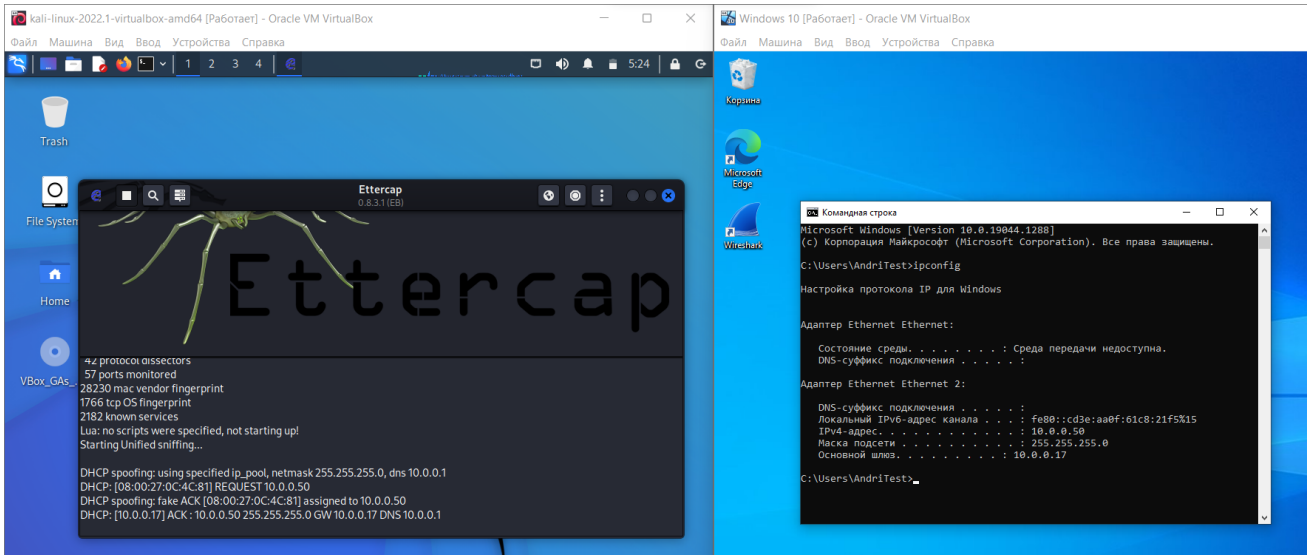

Рисунок 3.10 – «Rogue DHCP Server» атака

3.3.3. Последствия атаки

Теперь у жертвы в поле «Основной шлюз» стоит значение IP-адреса машины злоумышленника и тот может видеть весь траффик, который жертва отправляет в другую сеть. Это можно увидеть при мониторинге сетевого подключения между linux машиной и коммутатором (Рисунок 3.11) и на мониторинге сети с самой linux машины (Рисунок 2.12)

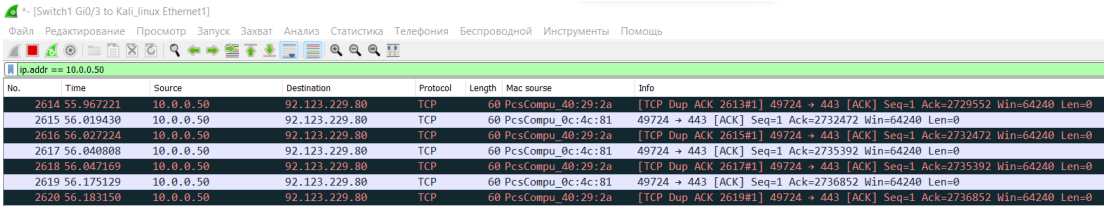

Рисунок 3.11 – Мониторинг сетевого подключения между linux машиной и коммутатором

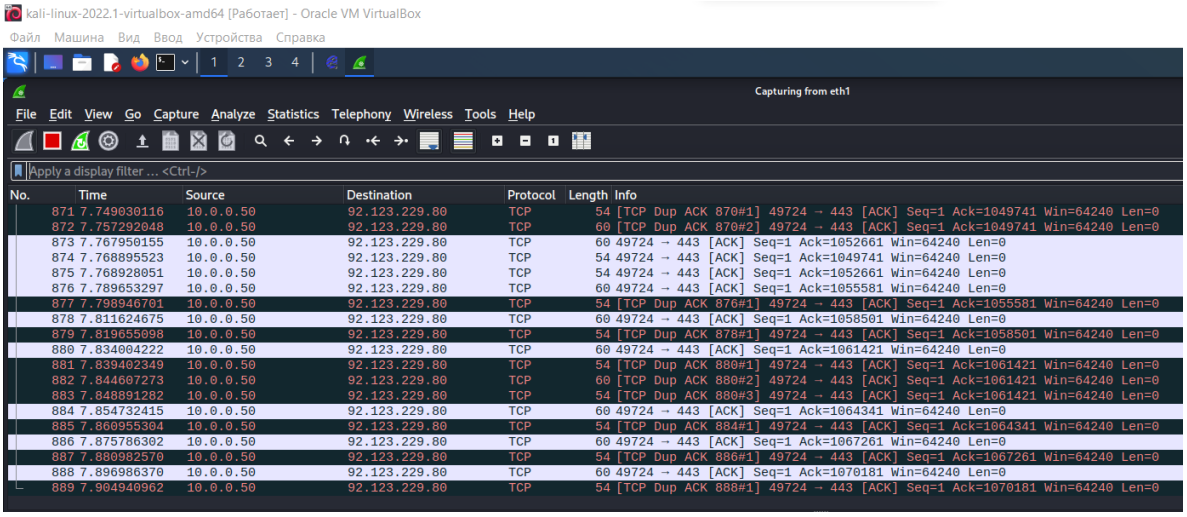

Рисунок 3.12 – Мониторинг сети на linux машине

Попробую использовать уязвимость, что я сделал. Допустим, жертва вводит свои данные на сайте, который не шифрует данные сразу. Для примера я буду использовать свой тестовый сайт, находящийся по адресу http://mysitetest.h1n.ru представленный на рисунке 3.13 в правом окне. Он не шифрует данные до обработки на отдельной странице, а значит, при переброске этих данных, их можно перехватить.

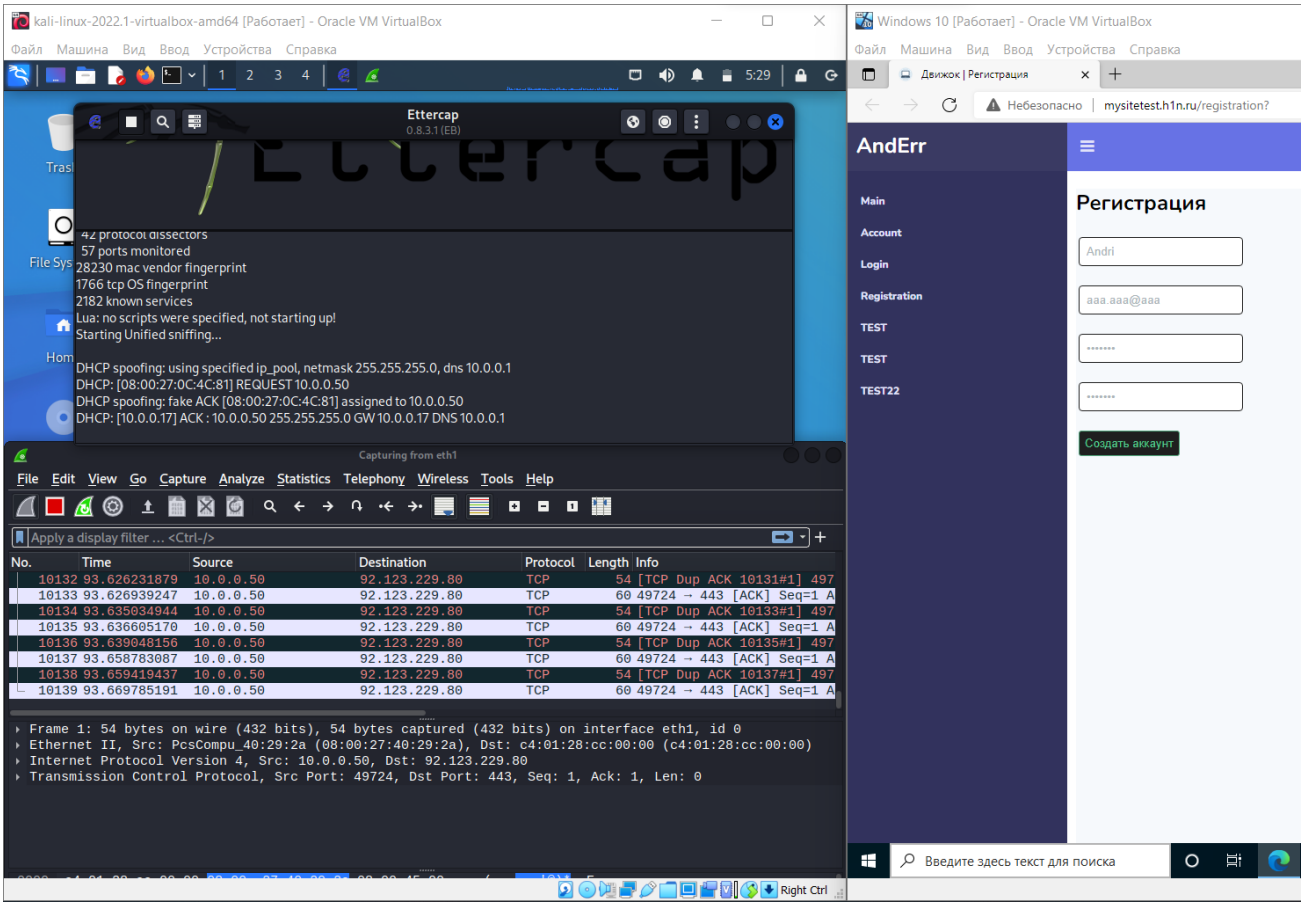

Рисунок 3.13 – Тестовый сайт для MITM атаки

После ввода данных, оправляю их. Видно, что kali машина перехватила отправленный трафик. В том числе и HTTP. В пакете под номером 13007 видны пересылаемые регистрационные данные в незашифрованном виде, как показано на рисунке 3.14 в окне слева снизу. Приложение «Ettercap» так же зафиксировала этот пакет и вывела данные из него, как видно на рисунке 3.14 в окне слева сверху.

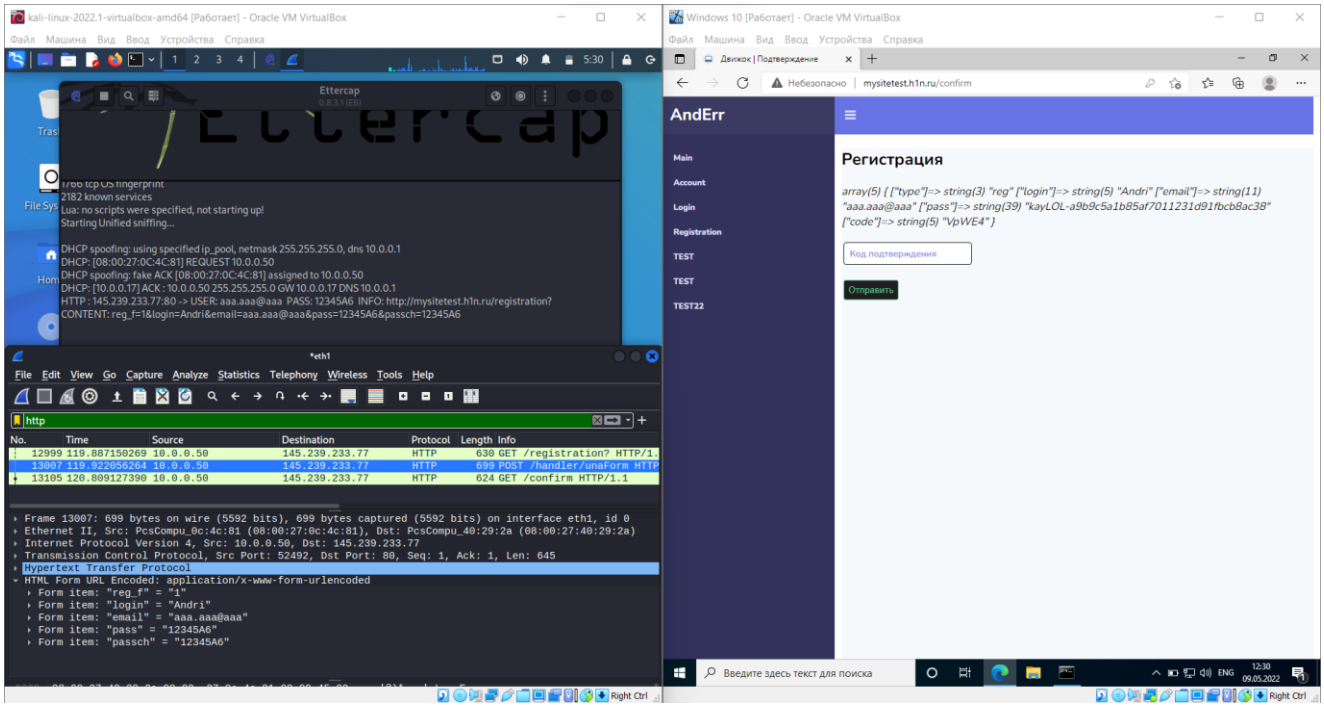

Рисунок 3.14 – MITM атака

И это лишь один пример возможного использования уязвимости, вызванной «Rogue DHCP Server» атакой.

3.4. Принципы защиты от «Rogue DHCP Server» атаки

Защититься от «Rogue DHCP Server» и «DHCP starvation» атак возможно. Но для этого понадобится коммутатор 2 уровня с поддержкой функции «DHCP Snooping».

DHCP Snooping — это технология безопасности уровня 2 по OSI (канальный уровень), встроенная в операционную систему сетевого коммутатора, которая отбрасывает трафик DHCP, определенный как неприемлемый. DHCP Snooping предотвращает несанкционированные (мошеннические) DHCP-серверы, предлагающие IP-адреса DHCP клиентам а так же может ограничивать количество проходящих DHCP-сообщений на определённом порту.

DHCP Snooping применим только к проводным пользователям. На коммутаторах вручную настраиваются доверенные порты, которые как правило подключены к маршрутизатору или DHCP серверу. В моей сети доверенным

поротом коммутатора будет «Gi0/1», подключённый к авторизованному DHCP серверу (Рисунок 3.15).

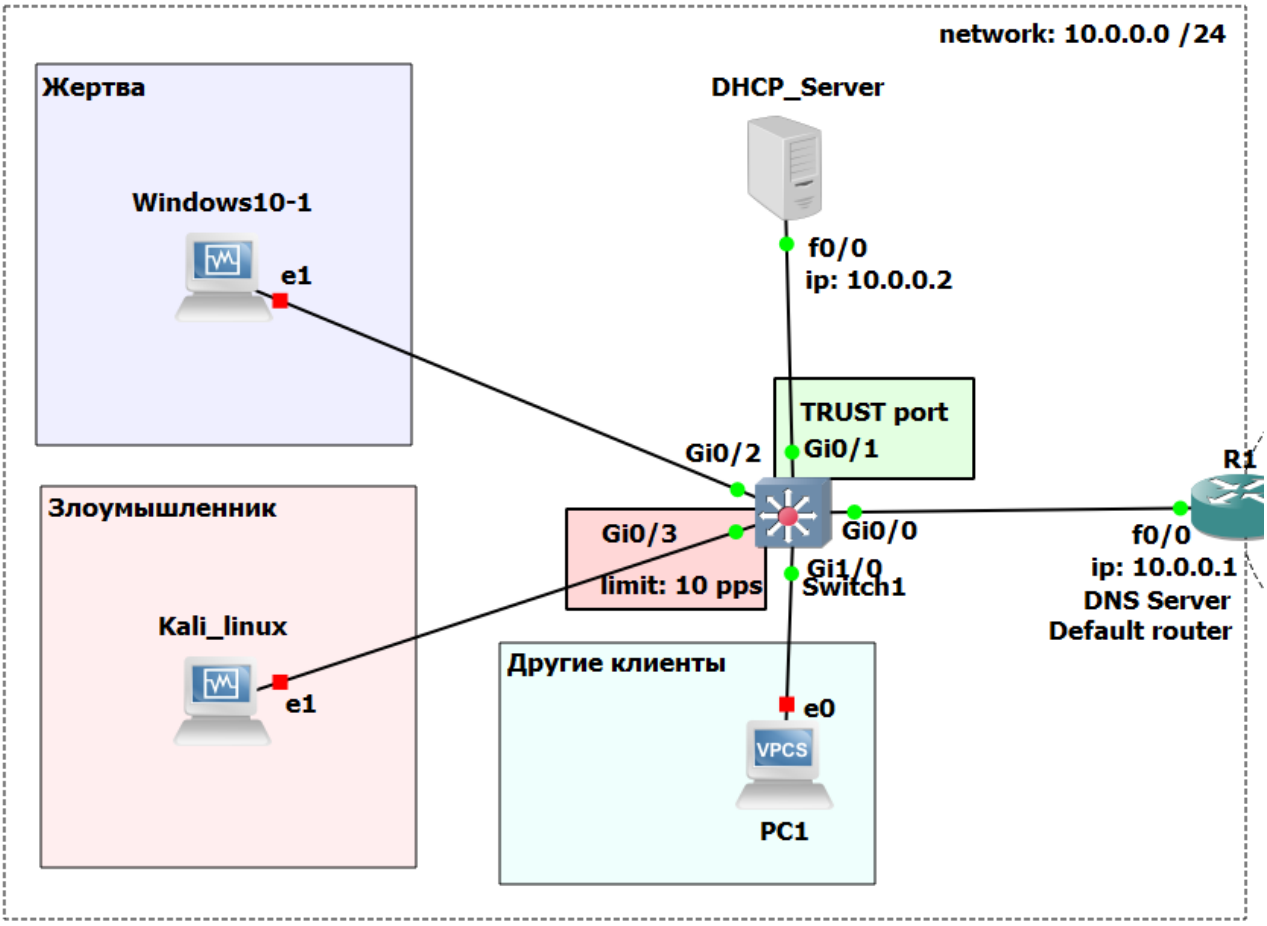

Рисунок 3.15 – Коммутатор с доверенным портом

Принцип работы защиты прост: на коммутаторе настраивается доверительный порт. Допустим, компьютер, который подключается к сети, посылает широковещательное сообщение DHCPDISCOVER. Его получают авторизованный DHCP сервер и мошеннический, и оба отвечают. Но ответ коммутатор пропустит только с доверенного порта, лишая злоумышленника возможности произвести атаку даже при отключённом DHCP сервере (Рисунок 3.16). Так же, для предотвращения атаки «DHCP starvation» под не доверенными портами можно настроить опцию ограничения количества DHCP обращений в секунду (pps - packets per second). Важно не занизить данную характеристику, чтобы не порезать валидный трафик. Cosco советует использовать число «10».

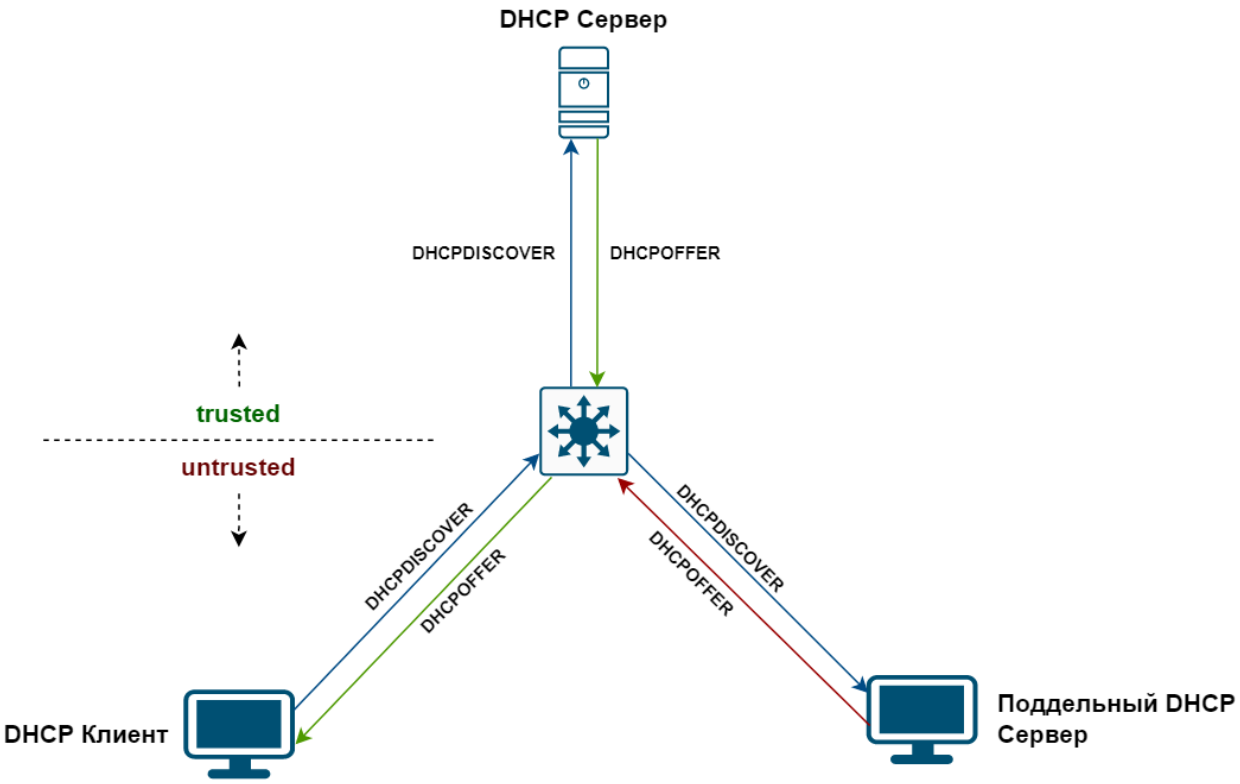

Рисунок 3.16 – Принцип работы DHCP Snooping

3.5. Реализация защиты от «Rogue DHCP Server» атаки

Поставлю на коммутатор меры защиты, описанные в прошлом пункте. Доверенным портом у меня является «Gi0/1», а портом с ограничением количества пакетов «Gi0/3». Я поставил значение «10», то есть в секунду порт будет пропускать максимум 10 клиентских запросов. Повторим атаку «DHCP starvation» (Рисунок 3.17). Можно увидеть, что «Yersinia» отправила более 2-х миллионов DHCPDISCOVER сообщений в сеть. Но коммутатор пропустил только 9 из них, что видно на рисунке 3.17 в правом окне – сообщения 63 – 72.

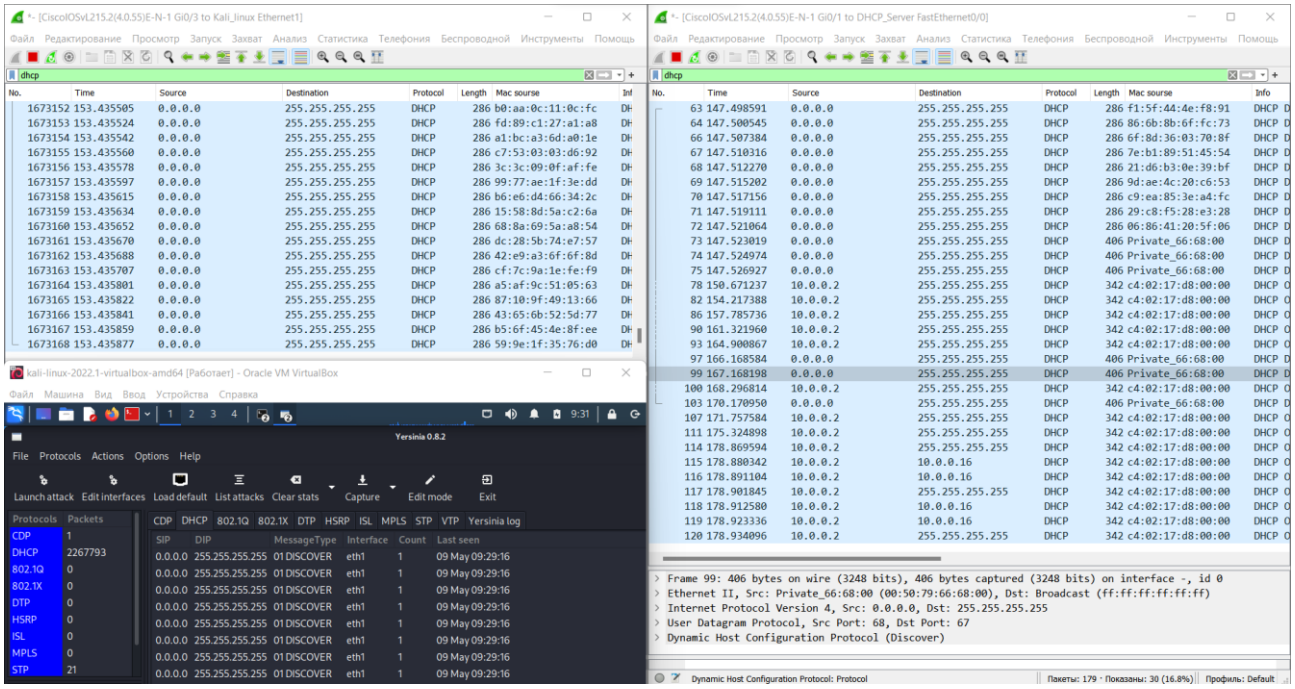

Рисунок 3.17 – Атака «DHCP starvation»

Попробуем же повторить «Rogue DHCP Server» атаку. Делаем всё так же, как и раньше со всеми теми же настройками. И начинаем атаку. Подключаем windows машину в сеть и видим, что она начала процедуру подключения к сети, но коммутатор пересылает все DHCP-сообщения на доверенный порт (Рисунки 3.18, 3.19).

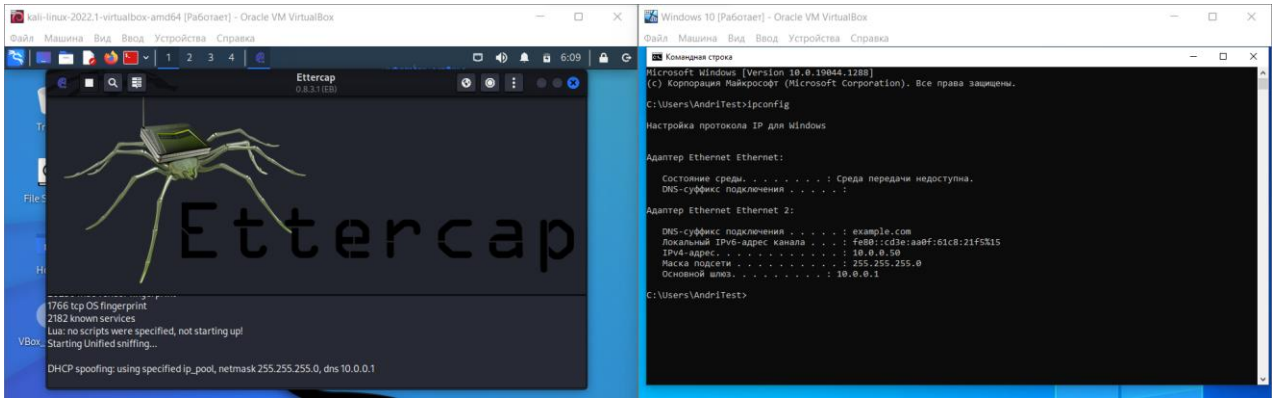

Рисунок 3.18 – «Rogue DHCP Server» неудачная атака

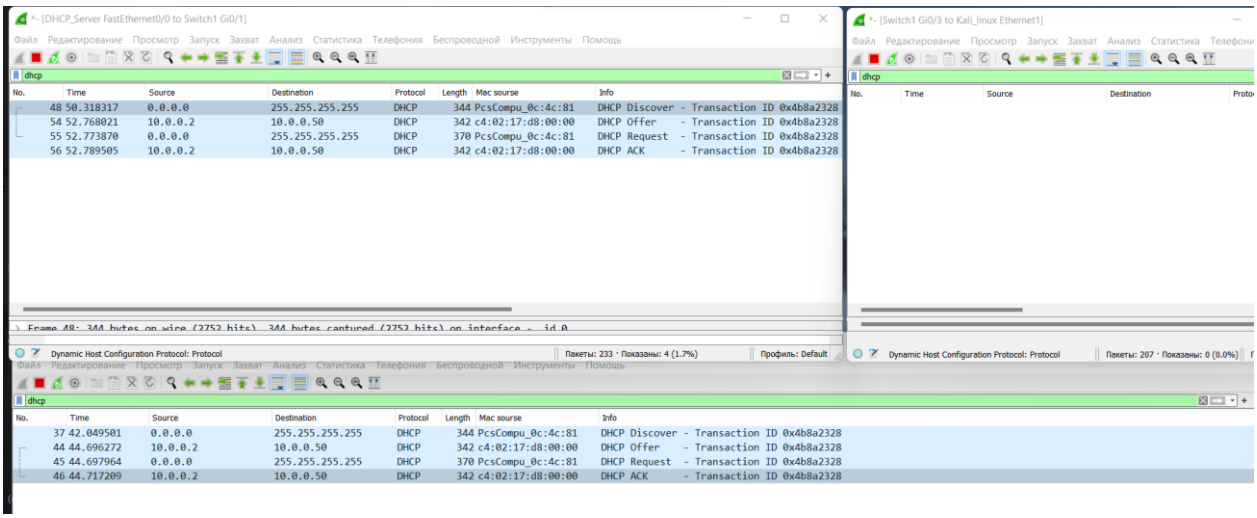

Рисунок 3.19 – «Rogue DHCP Server» неудачная атака

Таким образом я организовал защиту от «Rogue DHCP Server» атаки.

Настройки «DHCP Snooping» на коммутаторе можно увидеть в приложении 4.

ЗАКЛЮЧЕНИЕ

DHCP очень востребован в сетях. Он даёт возможность использовать своё устройство в любой сети, без предварительной ручной настройки. И его используют и будут использовать практически все более-менее крупные сети. Поэтому нужно знать уязвимости протокола и уметь противостоять атакам на них.

В ходе написания курсовой работы были достигнуты главные цели - был изучен DHCP, разобрана и реализована «Rogue DHCP Server» атака и защита от неё.

В ходе написания курсовой работы были выполнены все поставленные перед этим задачи. Был разобран DHCP – его строение, опции, работа в сети, так же разобрана атака «Rogue DHCP Server» и связанная с ней «DHCP starvation» атака, была совершена MITM атака, как результат «Rogue DHCP Server» атаки. Была реализована защита от «DHCP starvation» и «Rogue DHCP Server» атак.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. RFS2131 Dynamic Host Configuration Protocol [ЭЛЕКТРОННЫЙ РЕСУРС]: Режим доступа: https://www.ietf.org/rfc/rfc2131.txt (дата обращения 05.05.22)
2. RFC2132 DHCP Options and BOOTP Vendor Extensions [ЭЛЕКТРОННЫЙ РЕСУРС]: Режим доступа: https://www.ietf.org/rfc/rfc2132.txt (дата обращения 05.05.22)
3. RFC1700 ASSIGNED NUMBERS [ЭЛЕКТРОННЫЙ РЕСУРС]: Режим доступа: https://www.ietf.org/rfc/rfc1700.txt (дата обращения 05.05.22)
4. Таненбаум Э., Уэзеролл Д. Компьютерные сети. 5-е изд. — СПб.: Питер, 2012. — 960 с.
5. Yersinia [ЭЛЕКТРОННЫЙ РЕСУРС]: Режим доступа: https://www.kali.org/tools/yersinia/ (дата обращения 05.05.22)
6. Ettercap [ЭЛЕКТРОННЫЙ РЕСУРС]: Режим доступа: https://www.kali.org/tools/ettercap/ (дата обращения 05.05.22)

ПРИЛОЖЕНИЯ

```
DHCP_Server#configure terminal
DHCP_Server(config)#service dhcp
DHCP_Server(config)#ip dhcp excluded-address 10.0.0.1 10.0.0.15
DHCP_Server(config)#ip dhcp pool net1
DHCP_Server(dhcp-config)#network 10.0.0.0 255.255.255.0
DHCP_Server(dhcp-config)#domain-name example.com
DHCP_Server(dhcp-config)#dns-server 10.0.0.1
DHCP_Server(dhcp-config)#default-router 10.0.0.1
DHCP_Server(dhcp-config)#lease 0 1
DHCP_Server(dhcp-config)#exit
DHCP_Server(config)#interface fastEthernet 0/0
DHCP_Server(config-if)#ip address 10.0.0.2 255.255.255.0
DHCP_Server(config-if)#no shutdown
DHCP_Server(config-if)#end
```

Приложение 1 – Настройка DHCP_Server

```
Switch#configure terminal
Switch(config)#interface range GigabitEthernet 0/0 - 3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 1
Switch(config-if-range)#exit
Switch(config)#interface range GigabitEthernet 1/0 - 3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 1
Switch(config-if-range)#end
```

Приложение 2 – Настройка Switch1 без защиты от атаки

```
R1#configure terminal
R1(config)#interface fastEthernet 0/1
R1(config-if)#ip address dhcp
R1(config-if)#no shutdown
R1(config-if)#ip nat outside
```

```
R1(config-if)#exit
R1(config)#interface fastEthernet 0/0
R1(config-if)#ip address 10.0.0.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#ip nat inside source list 1 interface fastEthernet 0/1 overload
R1(config)#access-list 1 permit any
R1(config)#ip dns server
R1(config)#ip name-server 192.168.1.1
R1(config)#ip name-server 192.168.122.1
R1(config)#ip domain lookup
R1(config)#end
```

Приложение 3 – Настройка R1

```
Switch#configure terminal
Switch(config)#ip dhcp snooping
Switch(config)#ip dhcp snooping vlan 1
Switch(config)#no ip dhcp snooping information option
Switch(config)#int GIgabitEthernet 0/1
Switch(config-if)#ip dhcp snooping trust
Switch(config-if)#exit
Switch(config)#int gigabitEthernet 0/3
Switch(config-if)#ip dhcp snooping limit rate 10
Switch(config-if)#end
```

Приложение 4 – Дополнительная настройка Switch1 с защитой от атаки