

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (ДГТУ)**

Кафедра «Вычислительные системы и информационная безопасность»

**Методические указания по выполнению контрольной работы по дисциплине**

«Технологии обеспечения информационной безопасности информационно-телекоммуникационных сетей»

Ростов-на-Дону 2024

**Оглавление**

[Инструкция по технике безопасности при выполнении лабораторных работ 5](#_Toc147706572)

[Материально-техническое обеспечение работ 6](#_Toc147706573)

[Лабораторная работа № 1 – Программные средства анализа информационно-телекоммуникационных сетей 7](#_Toc147706574)

[Лабораторная работа № 2 – Атака на коммутатор путём переполнения CAM-таблицы и методы защиты от ней 12](#_Toc147706579)

[Лабораторная работа № 3 – Атака DHCP-Spoofing 16](#_Toc147706584)

[Лабораторная работа № 4 – Защита от атак типа DHCP-spoofing 24](#_Toc147706589)

[Лабораторная работа № 5 – Атака ARP-Poisoning 28](#_Toc147706594)

[Лабораторная работа № 6 – Защита от атаки ARP-spoofing 36](#_Toc147706599)

[Лабораторная работа № 7 – Подмена маршрутов в протоколе RIP 39](#_Toc147706604)

[Лабораторная работа № 8 – Защита от подмены маршрутов в протоколе RIP и OSPF 45](#_Toc147706609)

[Лабораторная работа № 9 – Настройка правил фильтрации трафика в сетевом экране Linux с помощью утилиты iptables 48](#_Toc147706614)

[Лабораторная работа № 10 – Настройка прокси-сервера squid в Linux 54](#_Toc147706619)

[Перечень использованных информационных ресурсов 62](#_Toc147706624)

# Инструкция по технике безопасности при выполнении лабораторных работ

1. Запрещено входить в кабинет в верхней одежде, головных уборах, с громоздкими предметами и едой.
2. Напряжение в сети кабинета включается и выключается только преподавателем.
3. Убедитесь в отсутствии видимых повреждений рабочего места; сядьте так, чтобы линия взора приходилась в центр экрана, чтобы, не наклоняясь пользоваться клавиатурой и воспринимать передаваемую на экран монитора информацию.
4. Внимательно слушайте объяснения и старайтесь понять цель и последовательность действий; в случае необходимости обращайтесь к преподавателю.
5. Запрещено бегать и прыгать, самовольно передвигаться по кабинету.
6. Запрещается шуметь, громко разговаривать и отвлекать других учащихся.
7. Работа на ЭВМ требует большого внимания, чётких действий и самоконтроля. С техникой обращаться бережно: не стучать по мониторам, не стучать мышкой о стол, не стучать по клавишам клавиатуры.
8. При возникновении неполадок: появлении изменений в функционировании аппаратуры, самопроизвольного её отключения необходимо немедленно прекратить работу и сообщить об этом преподавателю.
9. Не пытаться исправить неполадки в оборудовании самостоятельно.
10. Выполнять за компьютером только те действия, которые говорит преподаватель.
11. По окончанию работы выключите компьютер.

# Материально-техническое обеспечение работ

1. Столы аудиторные.
2. Стулья аудиторные.
3. Доска аудиторная.
4. ПК с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду вуза.
5. Пакет офисных приложений для удаленного использования для студентов: Microsoft 0365ProPlusOpenStudents ShrdSvr ALNG SubsVL OLV NL 1Mth Acdmc Stdnt w/Faculty (Государственный контракт № 58100011820000013-01).
6. Программное обеспечение для рабочих станций пользователей: Microsoft DsktpEdu ALNG LicSAPk OLV E (Государственный контракт № 0358100011820000007).
7. Комплекты лицензионного ПО: Microsoft DsktpEdu ALNG LicSAPk OLV E Стандартный Russian Edition. 2500-4999 Node 1 year Educational Renewal License. (Антивирус Касперского для рабочих станций) (лиценз. Российское ПО).
8. Маршрутизатор CISCO 2800.
9. Маршрутизатор CISCO 1720.
10. Межсетевой экран ZyXEL ZyWALL USG50.
11. WiFi-точка доступа D-Link.
12. Беспроводной маршрутизатор ZyXEL Keenetic Ultra 7571.
13. Коммутатор CISCO 2950.
14. Коммутатор CISCO 2960.
15. Тестер кабеля 5bites LY-CT007.

# Лабораторная работа № 1 – Программные средства анализа информационно-телекоммуникационных сетей

## Цель работы: изучить утилиты для захвата сетевого трафика, научиться анализировать захваченные пакеты, извлекать из них полезную информацию.

## Теоретическая часть

Утилита TCPDump представляет собой мощный и популярный инструмент для перехвата и анализа сетевых пакетов. Она позволяет просматривать все входящие и исходящие из определенного интерфейса пакеты и работает в командной строке. Её использование необходимо в случаях, когда невозможно запустить Wireshark, например, в операционных системах без графической оболочки. Данная утилита часто используется для устранения неполадок в сети, а также для обеспечения безопасности.

Если tcpdump запустить без параметров, он будет выводить информацию обо всех сетевых пакетах. С помощью параметра -i можно указать сетевой интерфейс, с которого следует принимать данные, например:

sudo tcpdump -i enp0s8

или

sudo tcpdump -i any

Чтобы посмотреть названия всех доступных для захвата интерфейсов, нужно запустить TCPDump с ключом -D.

Для более точной настройки работы TCPDump можно использовать один из перечисленных далее параметров, или их комбинацию.

* выводить сокращённую информацию: -q;
* выводить подробную информацию: -v -vv -vvv (чем больше v, тем подробнее информация);
* не преобразовывать порты: -n (вместо названий, например, http или ftp будут показываться номера, для указанных протоколов: 80 и 21);
* не преобразовывать имена: -nn (вместо имён компьютеров, например, UbuntuServer1804 будут показываться их IP-адреса);
* показывать MAC-адреса: -e;
* захватить только определённое количество пакетов: -с количество;

Для протокола, чьи пакеты должны захватываться, достаточно указать его название, например:

tcpdump icmp

или

tcpdump udp port 53

Параметр port позволяет ограничить для захватываемого трафика порт или их диапазон. Данный параметр применим только к протоколам TCP и UDP.

По умолчанию TCPDump выводит информацию о захваченных пакетах на экран, однако её можно записывать в файл. Для это используется параметр -w:

tcpdump -w /home/student/1.cap

Полученный в результате файл имеет формат, совместимый с Wireshark и может быть в дальнейшем проанализирован в данной программе с использованием более наглядного графического интерфейса, а также широких возможностей по созданию фильтров и подсчёту статистики.

Ещё одна полезная сетевая утилита — NMap («Network Mapper») предназначена для исследования сети и проверки безопасности. Результатом её работы является список просканированных целей с дополнительной информацией по каждой из них в зависимости от заданных опций. Ключевой информацией является «таблица важных портов».

Эта таблица содержит номер порта, протокол, имя службы и состояние. Состояние может иметь значение open (открыт), filtered (фильтруется), closed (закрыт) или unfiltered (не фильтруется). Открыт означает, что приложение на целевой машине готово для установки соединения/принятия пакетов на этот порт. Фильтруется означает, что брандмауэр, сетевой фильтр, или какая-то другая помеха в сети блокирует порт, и Nmap не может установить открыт этот порт или закрыт. Закрытые порты не связаны ни с каким приложением, но могут быть открыты в любой момент. Порты расцениваются как не фильтрованные, когда они отвечают на запросы Nmap, но Nmap не может определить открыты они или закрыты. Nmap выдает комбинации открыт|фильтруется и закрыт|фильтруется, когда не может определить, какое из этих двух состояний описывает порт. Эта таблица также может предоставлять детали о версии программного обеспечения, если это было указано при запуске сканирования.

Использование NMap заключается в выполнении команды:

nmap [<Тип сканирования>] [<Опции>] {<цель сканирования>}

Тип сканирования определяет используемые пакеты и подход к интерпретации результата. Для протокола TCP можно задать один или несколько из следующих типов сканирования:

-sS — TCP SYN;

-sT — TCP connect;

-sN; -sF; -sX — NULL, FIN и Xmas;

-sA — TCP ACK;

-sW — TCP Window;

-sM — TCP сканирование Мэймона.

Подробнее про их различие можно прочитать в официальной русскоязычной документации (<https://nmap.org/man/ru/man-port-scanning-techniques.html>). Чем больше типов сканирования выбрано при запуске команды, тем дольше оно будет осуществляться.

Для сканирования UDP-портов необходимо использовать параметр -sU.

По умолчанию NMap проверяет только первую тысячу портов (от 1 до 1000). При необходимости изменить данный диапазон можно использовать параметр -p, например:

nmap -sU -p 520 172.16.0.1-254

В представленной команде указывается сканировать только UDP-порт 520 на всех узлах с адресами от 172.16.0.1 до 172.16.0.254.

Возможен и противоположный вариант, например, сканировать все TCP-порты от 1 до 65535 на одном узле с адресом 10.0.0.1:

nmap -sS -p 1-65535 10.0.0.1

Последний параметр называется целью сканирования и определяет множество адресов, для которых должно выполняться сканирование. В нём допустимо указывать:

* единичный IP адрес: nmap 192.168.1.1
* доменное имя узла: nmap donstu.local
* множество IP-адресов: nmap 192.168.1.1 192.168.1.2 192.168.1.3
* диапазон IP-адресов: 192.168.1.1-254
* подсеть: 192.168.1.0/24

NMap имеет множество дополнительных опций, обеспечивающих широкие возможности сбора информации о сети. Среди них:

-sV — детальное исследование портов для определения версий служб;

-O — определять операционную систему.

Для упрощения использования NMap для неё существует официальный графический интерфейс — ZenMap.

## Порядок выполнения работы

1. TCPDump.

1.1. С помощью утилиты TCPDump захватите и выведите на экран информацию из входящих пакетов протоколов HTTP и ICMP.

1.2. С помощью утилиты TCPDump захватите и запишите в файл трафик, передаваемый виртуальной машиной с Ubuntu Server в процессе обращения к ней по протоколу HTPP и выполнения команды ping.

1.3. Передайте этот трафик на компьютер с операционной системой Windows и установленной программой Wireshark.

2. Wireshark

2.1. С использованием Wireshark выделите из общего трафика отдельно HTTP и отдельно ICMP-пакеты.

2.2. Определите все протоколы, чьи пакеты присутствуют в захваченном трафике.

2.3. Определите все MAC-адреса, которые присутствуют в захваченном трафике.

2.4. Определите все IP-адреса, которые присутствуют в захваченном трафике.

2.5. Определите все запрашиваемые доменные имена.

3. NMap

3.1 Просканируйте какую-либо сеть (реальную или виртуальную) с целью определить операционные системы оборудования, входящего в неё, наличие других сетевых сервисов, используемые служебные сетевые протоколы (например DHCP или протоколы маршрутизации).

## Контрольные вопросы

1. Что такое TCPDump?
2. Приведите примеры нескольких фильтров для TCPDump.
3. Как проанализировать трафик, захваченный с помощью TCPDump в Wireshark?
4. Какая статистика может быть получена в Wireshark для сетевого трафика?
5. Что такое NMap и ZenMap?
6. Что является результатом работы NMap/ZenMap?
7. Что означает «открытый», «закрытый» и «фильтруемый» порт?

# Лабораторная работа № 2 – Атака на коммутатор путём переполнения CAM-таблицы и методы защиты от ней

## Цель работы: изучить способы выполнения атаки на коммутатор путём переполнения CAM-таблицы.

## Теоретическая часть

Во всех моделях коммутаторов Cisco используется таблица CAM (Content Address Memory) для коммутации на втором уровне. Когда кадры поступают на порты коммутатора, MAC-адрес источника запоминается и записывается в таблицу CAM. Порт, на который были получены кадры, и сеть VLAN записываются в таблице вместе с меткой времени. Если MAC-адрес, изученный одним портом коммутатора, был перемещен на другой порт, записывается MAC-адрес и метка времени того порта, который получил кадры последним.

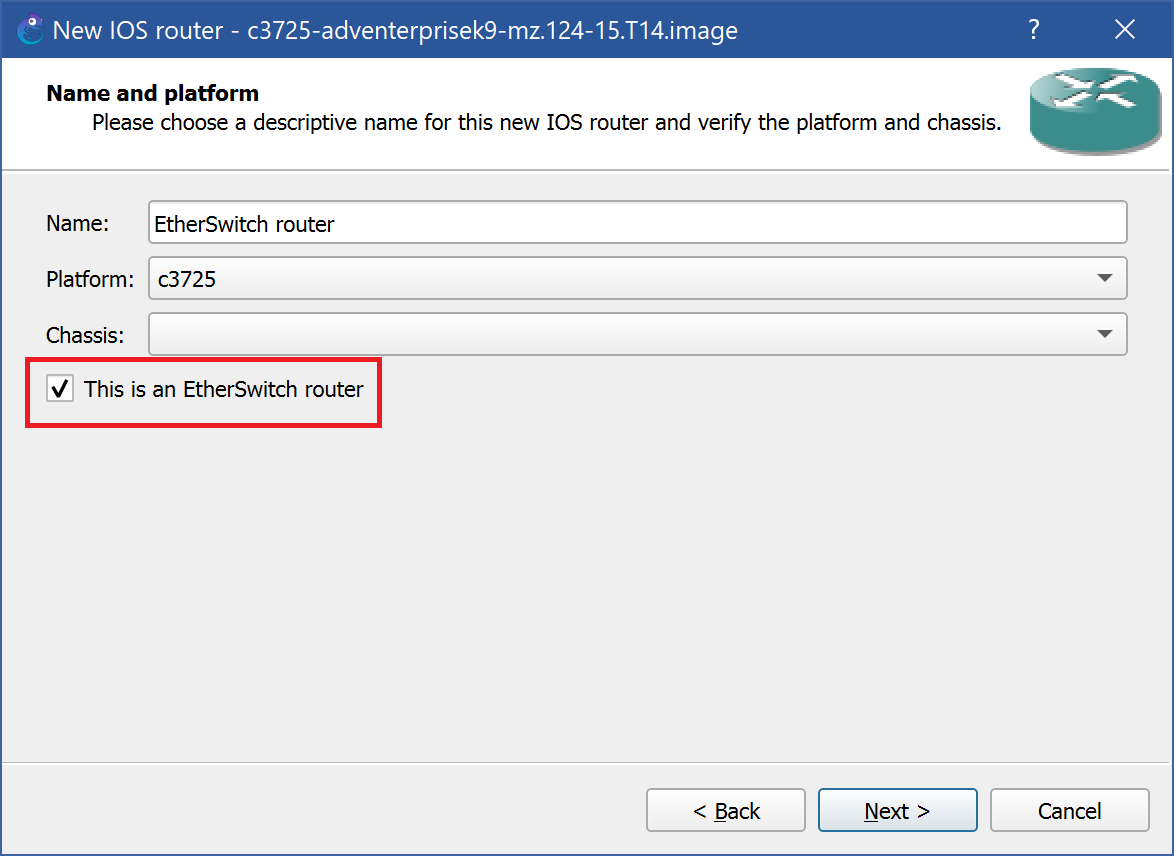
Разумеется, CAM-таблица имеет ограничения на свой размеры, связанные с ограниченным объёмом памяти коммутатора. Например, коммутатор Cisco Catalyst 2960 может содержать 8192 MAC-адресов, Cisco Catalyst 6000 серии – 128000 MAC-адресов. по умолчанию записи в CAM-таблице хранятся в течении 300 секунд, однако это значение может быть вручную изменено администратором.

В случае, если CAM-таблица будет полностью занята, новые записи в неё не смогут добавляться и приходящие пакеты будут передаваться на все интерфейсы коммутатора, что даст возможность злоумышленнику перехватывать весь трафик, передаваемый в данном сегменте локальной сети.

Для предотвращения описанной атаки необходимо указать, что на порту коммутатора, к которому подключен компьютер, может быть, не больше определённого количества MAC-адресов, и в случае его превышения, порт необходимо перевести в отключенное состояние и отправить сообщение администратору о нарушении безопасности.

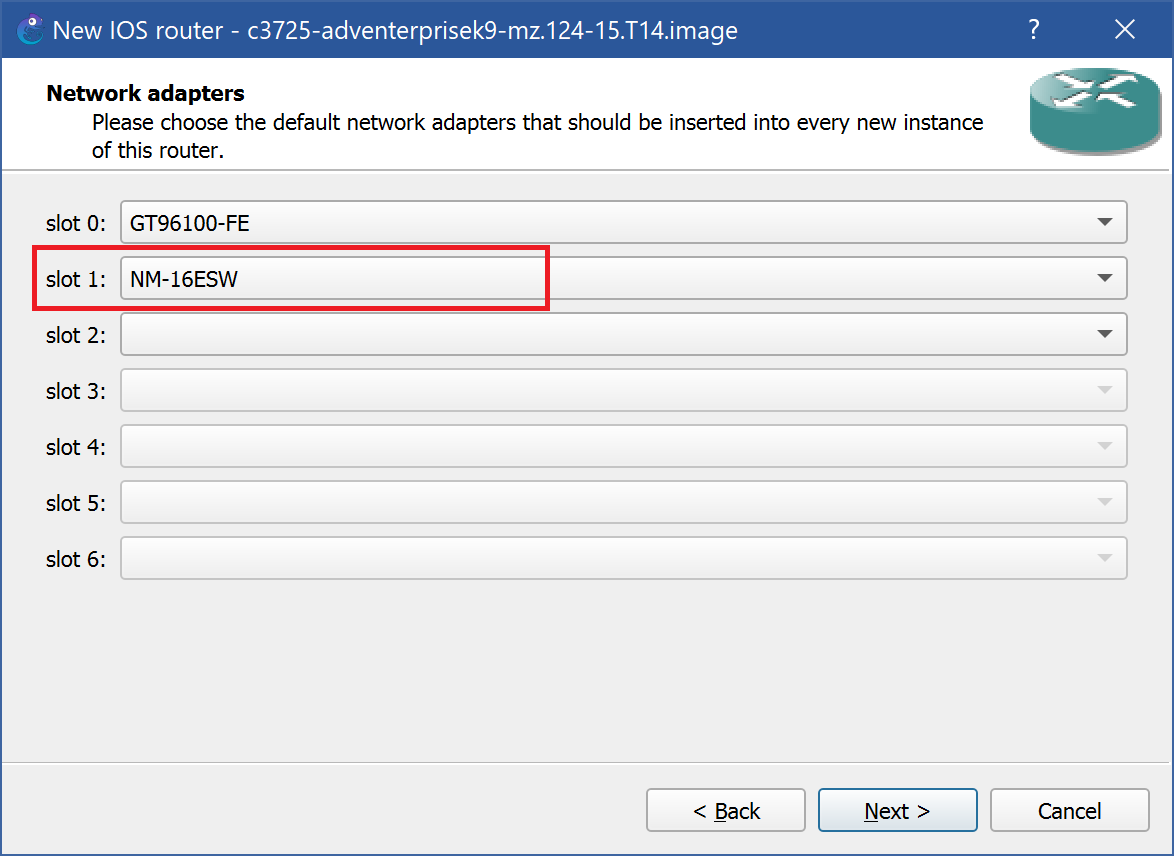
## Порядок выполнения работы

1. Добавьте в GNS3 маршрутизатор, выполняющий функции коммутатора. К ним относятся, например, маршрутизаторы моделей 3745 и, использованный ранее, 3725. Однако для использования в качестве коммутатора его необходимо добавить заново с соответствующими настройками. Для этого следуйте инструкциям из лабораторной работы на тему «GNS3», но при настройке устройства отметьте галочку «This is an EtherSwitch router»

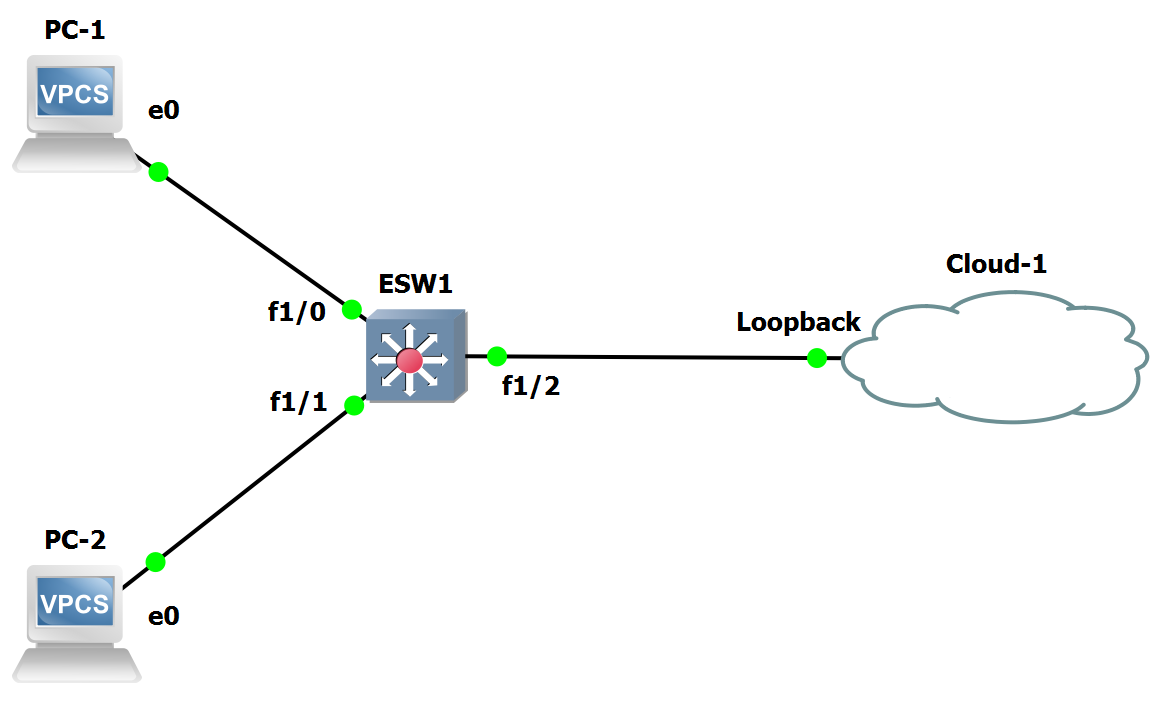


В следующем окне добавьте памяти хотя бы до 256 Мб.

Затем убедитесь, что в качестве одного из модулей выбран модуль, выполняющий функции свитча: «NM-16ESW».



2. Постройте в GNS3 сеть в соответствии со схемой, изображённой на рисунке, задайте IP-адреса и маски всем компьютерам, проверьте наличие связи между всеми компьютерами (в том числе реальным и виртуальными) с помощью команды ping.



3. Проверьте содержимое CAM-таблицы коммутатора.

В оборудовании Cisco для этого используется команда, выполняемая в привилегированном режиме:

show mac-address-table

4. С использованием библиотек SharpPcap и Packet.NET напишите программу, которая может генерировать не менее 8192 сетевых пакетов протокола Ethernet. В качестве MAC-адреса назначения указывайте в них любой адрес, а в качестве адреса отправителя — случайный MAC-адрес.

5. Определите количество записей в CAM-таблице коммутатора. Для этого используйте команду:

show mac-address-table count

6. На компьютере 1 запустите написанную программу и отправьте через интерфейс, подключённый к виртуальному свитчу, генерируемые ей пакеты.

7. На компьютере 1 запустите программу Wireshark и включите захват пакетов.

8. Выполните команду ping 192.168.1.3 на 2-ом компьютере. Наблюдайте за захваченными в Wireshark пакетами на 1-ом компьютере.

9. Закрепите за интерфейсом коммутатора, к которому подключён 1-ый компьютер реальный MAC-адрес этого компьютера с помощью последовательности команд:

(config)# interface fa0/1

(config-if)# switchport mode access

(config-if)# switchport port-security

(config-if)# switchport port-security mac-address 00E0.F964.63C8

(config-if)# switchport port-security violation shutdown

*Вместо значений fa0/1 и 00E0.F964.63C8 используйте реальные номер интерфейса и MAC-адрес.*

10. Повторите пункты 6 – 8 для проверки возможности выполнения атаки.

11. Настройте ещё один способ защиты от переполнения CAM-таблицы, который заключается в ограничении максимального количества MAC-адресов для одного интерфейса:

(config)# interface fa0/2

(config-if)# switchport mode access

(config-if)# switchport port-security

(config-if)# switchport port-security maximum 5

(config-if)# switchport port-security violation shutdown

12. Повторите пункты 6 – 8 для проверки возможности выполнения атаки.

## Контрольные вопросы

1. Что такое CAM-таблица?
2. Какое максимальное количество записей может храниться в CAM-таблице? Как его определить?
3. Какие последствия может иметь переполнение CAM-таблицы?
4. Как долго записи хранятся в CAM-таблице?
5. Какие существуют способы защиты от переполнения CAM-таблицы?

# Лабораторная работа № 3 – Атака DHCP-Spoofing

## Цель работы: изучить методику выполнения атаки путём подмены ответов DHCP-сервера и возможности по перехвату сетевого трафика.

## Теоретическая часть

***Протокол DHCP***

DHCP (Dynamic Host Configuration Protocol — протокол динамической настройки узла) — сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP. Помимо IP-адреса, DHCP также может сообщать клиенту дополнительные параметры, необходимые для нормальной работы в сети. Эти параметры называются опциями DHCP.

Некоторыми из наиболее часто используемых опций являются:

* IP-адрес шлюза по умолчанию;
* маска подсети;
* адреса серверов DNS;
* имя домена DNS.

Шлюз по умолчанию (англ. Default gateway), — в маршрутизируемых протоколах — сетевой шлюз (компьютер или маршрутизатор), на который отправляется трафик, для которого невозможно определить маршрут исходя из таблиц маршрутизации. Термин «шлюз по умолчанию» применяется в основном в рабочих станциях, где его использование является штатным режимом рабочей станции. В локальных сетях шлюз по умолчанию зачастую используется для обеспечения выхода всех компьютеров сети в интернет.

***Атаки типа «человек посередине»***

Атака посредника («человек посередине», MITM-атака (Man in the middle)) — термин в криптографии, обозначающий ситуацию, когда криптоаналитик (атакующий) способен читать и видоизменять по своей воле сообщения, которыми обмениваются корреспонденты, причём ни один из последних не может догадаться о его присутствии в канале.

Атака обычно начинается с прослушивания канала связи и заканчивается тем, что криптоаналитик пытается подменить перехваченное сообщение, извлечь из него полезную информацию, перенаправить его на какой-нибудь внешний ресурс.

Предположим, объект A планирует передать объекту B некую информацию. Объект C обладает знаниями о структуре и свойствах используемого метода передачи данных, а также о факте планируемой передачи собственно информации, которую С планирует перехватить. Для совершения атаки С «представляется» объекту А как В, а объекту В — как А. Объект А, ошибочно полагая, что он направляет информацию В, посылает её объекту С. Объект С, получив информацию, и совершив с ней некоторые действия (например, скопировав или модифицировав в своих целях) пересылает данные собственно получателю — В; объект В, в свою очередь, считает, что информация была получена им напрямую от А.

***Подмена шлюза по умолчанию***

Протокол DHCP работает по следующему принципу. Во время старта системы компьютер, являющийся DHCP-клиентом, посылает в сеть широковещательный запрос на получение IP-адреса. DHCP-сервер откликается и посылает сообщение-ответ, содержащее IP-адрес и остальные конфигурационные параметры.

В случае, если в сети существуют, например, 2 DHCP-сервера, каждый из них получит запрос и сможет выдать в ответ параметры конфигурации сетевого интерфейса для запросившего их компьютера. При этом DHCP-серверы могут иметь различающиеся настройки, в частности, узел, желающий выполнить атаку типа «человек посередине» может выдать в качестве шлюза по умолчанию собственный IP-адрес, вследствие чего пакеты, отправляемые в интернет будут сначала проходить через него, а затем отправляться на реальное устройство, служащее интернет-шлюзом. Перенаправив трафик таким образом, атакующий получит доступ ко всем данным передаваемым с компьютера жертвы в интернет, в том числе к данным форм авторизации, содержащим логины и пароли пользователей.

## Порядок выполнения работы

Соберите сеть в соответствии со схемой, изображённой на рис. 3.1.



Рисунок 3.1 — Схема сети

2. Настройте подключение к сети интернет на компьютере № 1 и откройте общий доступ к этому подключению. Для этого необходимо открыть свойства сетевого подключения (рис. 3.2) нажав «Пуск» → «Панель управления» → «Центр управления сетями и общим доступом» → «Изменение параметров адаптера», далее нажать правой кнопкой на значок «Подключение по локальной сети» и выбрать пункт контекстного меню «Свойства».

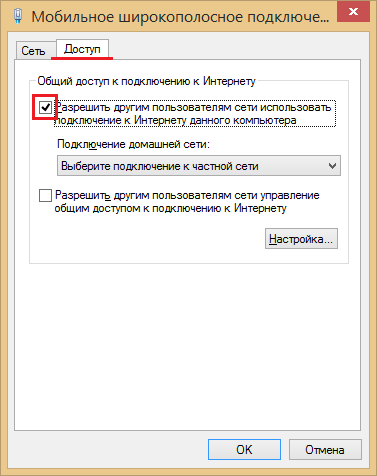


Рисунок 3.2 — Настройка общего доступа к соединению

3. Запустите DHCP-сервер на компьютере № 1.

3.1. В папке C:\OpenDHCPServer откройте конфигурационный файл opendhcpserver.ini.

3.2. В секции [RANGE\_SET] установите следующие настройки:

DHCPRange=192.168.1.1-192.168.1.254

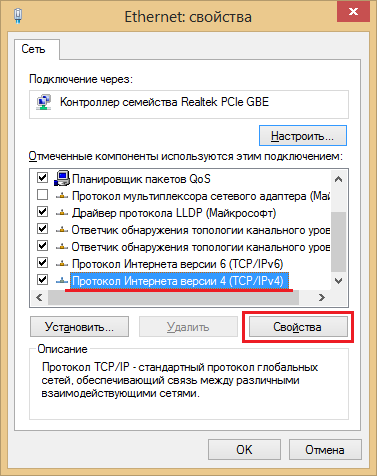
SubnetMask=255.255.255.0

DomainServer=192.168.1.1

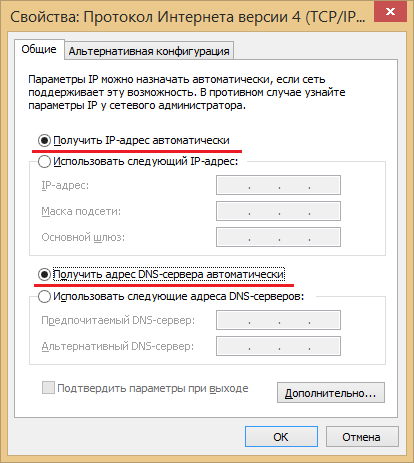
Router=192.168.1.1

3.3. Запустите сервер с помощью файла runstandalone.bat.

4. На компьютерах 2 и 3 настройте сетевые интерфейсы на автоматическое получение IP-адресов. Для этого необходимо открыть свойства сетевого подключения (рис. 3) нажав «Пуск» → «Панель управления» → «Центр управления сетями и общим доступом» → «Изменение параметров адаптера», далее нажать правой кнопкой на значок «Подключение по локальной сети» и выбрать пункт контекстного меню «Свойства».



а)



б)

Рисунок 3.3 — Настройка сетевого подключения: а — свойства подключения; б — настройки протокола IPv4

5. Проверьте правильность выдачи IP-адресов и наличие доступа в сеть интернет со всех компьютеров в сети. Для этого откройте командную строку («Пуск» → «Выполнить…» или «Win»+R) и введите команды ipconfig и ping 8.8.8.8. Результаты выполнения команд должны соответствовать приведённым на рис. 3.4.

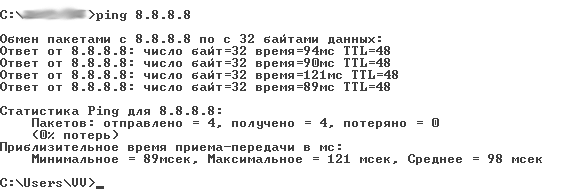


Рисунок 3.4 — Результаты выполнения команды ping

Если все настройки были выполнены верно, то компьютеры локальной сети должны получить доступ в интернет, причём пакеты будут передаваться в соответствии со схемой, приведённой на рис. 3.5.



Рисунок 3.5 — Схема прохождения пакетов

6. По аналогии с компьютером № 1, на компьютере № 3 установите статический IP-адрес 192.168.1.3, запустите DHCP-сервер и программу-сниффер (WireShark).

7. Включите на компьютере № 3 пересылку пакетов, предназначенных не ему. Для этого найдите в реестре следующий ключ:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\

\Parameters\IPEnableRouter

и измените его значение на 1.

8. Обновите сетевое соединение на компьютере № 2 и проверьте полученные настройки. Значение шлюза по умолчанию теперь будет зависеть от компьютера выдавшего настройки: если настройки будут выданы DHCP-сервером компьютера № 1, то 192.168.1.1, если DHCP-сервером компьютера № 3, то 192.168.1.3. Конкретный компьютер, выдавший настройки невозможно заранее предсказать, т.к. это зависит от большого числа факторов, в основном, загруженности каждого компьютера и времени прохождения пакетов к нему.

9. После получения компьютером № 2 в качестве шлюза по умолчанию IP-адреса компьютера № 3 (192.168.1.3), запустите веб-браузер, откройте, например, сайт ntb.donstu.ru и введите логин/пароль в форму авторизации.

10. На компьютере № 3 в окне программы WireShark обратите внимание на пополнившийся список захваченных пакетов, прошедших через этот компьютер. Полный путь пакетов представлен на рис. 3.6.



Рисунок 3.6 — Маршрут пакетов после подмены шлюза по умолчанию

11. Среди пакетов найдите пакет протокола TCP соответствующий представленному на рис. 3.7 и обратите внимание на строку «username=… &password=…» в которой указаны логин и пароль, введённые на сайте.

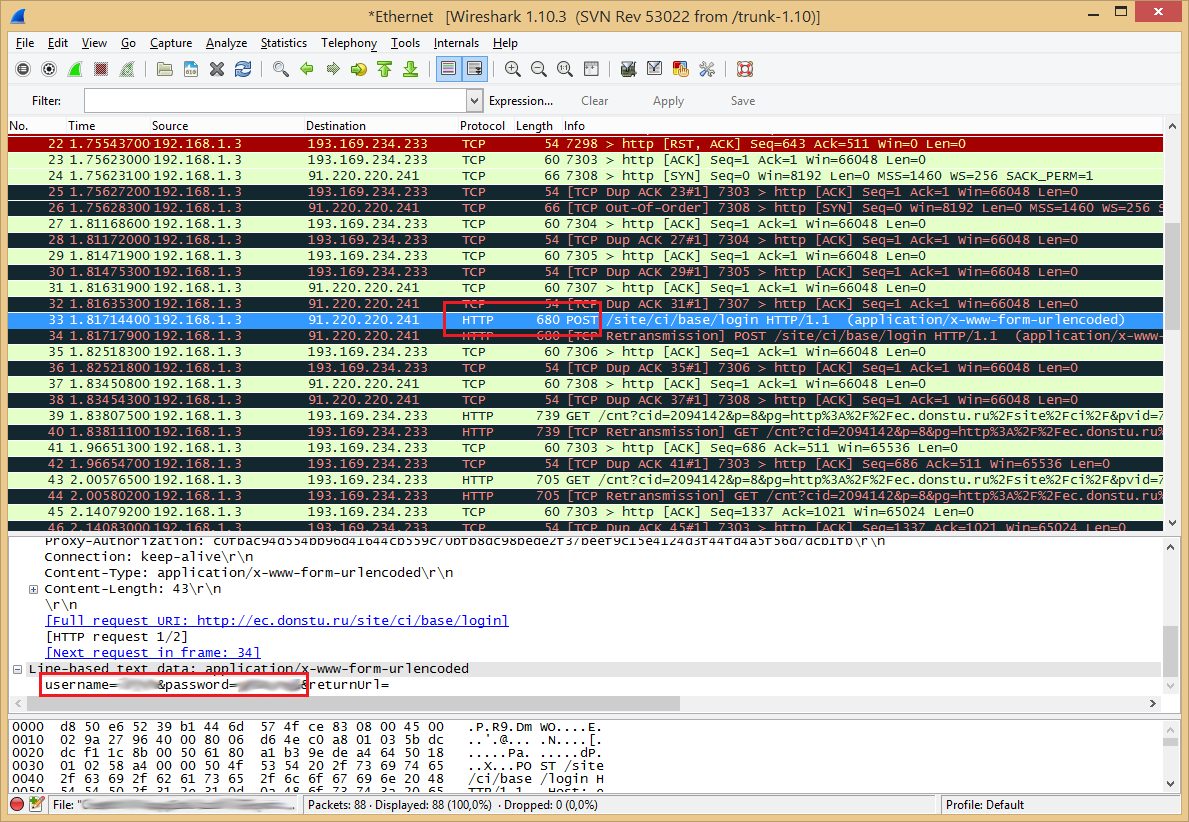


Рисунок 7 — Захваченные пакеты

## Контрольные вопросы

1. Какие функции выполняет протокол DHCP?
2. По какому принципу работает протокол DHCP?
3. Что такое шлюз по умолчанию?
4. Что такое атака типа «человек посередине»?
5. Как выполняется подмена адреса шлюза по умолчанию с помощью альтернативного DHCP-сервера?

# Лабораторная работа № 4 – Защита от атак типа DHCP-spoofing

## Цель работы: изучить практические аспекты сетевой атаки DHCP-spoofing и способы защиты от неё.

## Теоретическая часть

***Конфигурация DHCP на маршрутизаторах Cisco***

DHCP (англ. Dynamic Host Configuration Protocol — протокол динамической настройки узла) — сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие настройки, необходимые для работы в сети, такие как маска подсети, шлюз по умолчанию, адрес DNS-сервера.

Централизованное распределение IP-адресов позволяет не только избежать ручного ввода настроек сети на каждом компьютере, но и гарантирует отсутствие повторяющихся адресов внутри сети.

Для настройки DHCP используются следующие команды, вводить которые необходимо в режиме конфигурации маршрутизатора:

ip dhcp excluded-address адрес — позволяет исключить адрес из выдачи по протоколу DHCP.

ip dhcp pool название — создаёт пул (множество) IP-адресов для выдачи узлам сети и переходит в режим его конфигурирования.

В режиме конфигурации пула адресов:

default-router адрес — задаёт адрес, который будет назначен узлам сети в качестве шлюза по умолчанию.

network адрес маска — указывает сеть, из которой должны браться адреса для выдачи узлам сети.

***DHCP snooping***

DHCP snooping — функция коммутатора, предназначенная для защиты от атак с использованием протокола DHCP.

DHCP snooping регулирует только сообщения DHCP и не может повлиять напрямую на трафик пользователей или другие протоколы, однако некоторые функции коммутаторов, не имеющие непосредственного отношения к DHCP, могут выполнять проверки на основании таблицы привязок DHCP snooping. В их числе:

* Dynamic ARP Protection (Inspection) — проверка ARP-пакетов, направленная на борьбу с ARP-spoofing;
* IP Source Guard — выполняет проверку IP-адреса отправителя в IP-пакетах, предназначенная для борьбы с IP-spoofing’ом.

DHCP-snooping позволяет:

* защитить клиентов в сети от получения адреса от неавторизованного DHCP-сервера,
* регулировать какие сообщения протокола DHCP отбрасывать, какие перенаправлять и на какие порты.

Для правильной работы DHCP-snooping, необходимо указать какие порты коммутатора будут доверенными, а какие — нет:

* Недоверенные (Untrusted) — порты, к которым подключены клиенты. DHCP-ответы, приходящие с этих портов, отбрасываются коммутатором. Для ненадёжных портов выполняется ряд проверок сообщений DHCP и создаётся база данных привязки DHCP (DHCP snooping binding database).
* Доверенные (Trusted) — порты коммутатора, к которым подключен другой коммутатор или DHCP-сервер. DHCP-пакеты, полученные с доверенных портов, не отбрасываются.

Коммутатор с включённой функцией DHCP-snooping отбрасывает пакеты, пришедшие на недоверенный порт, если они относятся к одному из типов пакетов, отправляемых DHCP-сервером. Таким образом настройка данной функции сводится к указанию в качестве доверенного, того порта коммутатора, к которому подключён маршрутизатор или компьютер, выполняющие функции DHCP-сервера. Для это используются следующие команды выполняемые в режиме конфигурации:

switch(config)#ip dhcp snooping — включение функции DHCP-snooping

switch(config)#ip dhcp snooping vlan 1 — указание конкретного VLAN для которого данная функция будет использоваться.

switch(config)#interface fa0/1 — переход в режим настройки порта fa0/1.

switch(config-if)#ip dhcp snooping trust — обозначение порта в качестве доверенного.

После завершения всех настроек полезной может быть команда для просмотра информации о состоянии DHCP-snooping:

switch#show ip dhcp snooping

## Порядок выполнения работы

1. Постройте в Cisco Packet Tracer модель сети, полностью соответствующую показанной на рисунке 4.1.

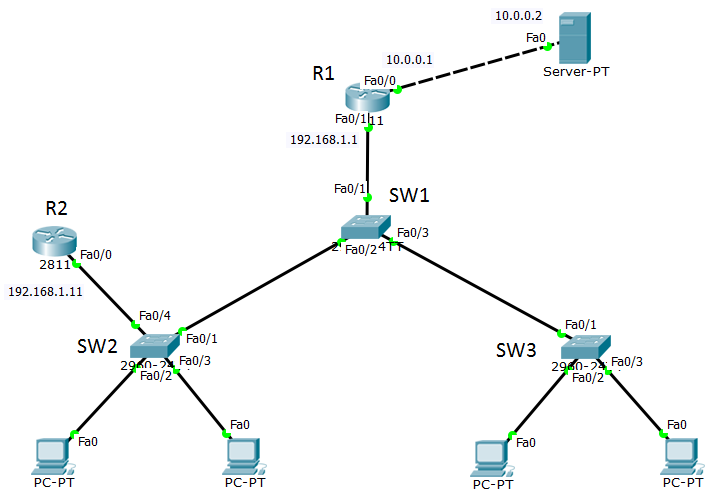


Рисунок 4.1 — Схема сети

3. На маршрутизаторе R1 настройте службу DHCP. При настройке в качестве шлюза по умолчанию укажите адрес 192.168.1.1.

4. Проверьте правильность выдачи сетевых настроек компьютерам по протоколу DHCP, а также путь пакета от любого компьютера к серверу.

5. Настройте службу DHCP на маршрутизаторе R2, указав в качестве шлюза по умолчанию адрес 192.168.1.11.

6. Проведите несколько экспериментов, переключая способ получения компьютерами сетевых настроек: статическое задание или получение по DHCP.

Определите, какие компьютеры и при каких условиях получают в качестве шлюза по умолчанию адрес 192.168.1.1, а какие 192.168.1.11.

Проверьте путь пакетов к серверу от компьютеров, использующих в качестве шлюза по умолчанию адрес 192.168.1.11.

7. На коммутаторе SW2 активируйте функцию DHCP-snooping, указав в качестве доверенного порта интерфейс fa0/1.

Убедитесь, что ни один компьютер в сети не получает по протоколу DHCP в сетевых настройках адрес шлюза по умолчанию 192.168.1.11.

## Контрольные вопросы

1. Как настраивается DHCP в маршрутизаторах Cisco?
2. Что такое DHCP-snooping?
3. Как работает DHCP-snooping?
4. Какие виды портов используются при настройке DHCP-snooping?

# Лабораторная работа № 5 – Атака ARP-Poisoning

## Цель работы: изучить методику выполнения атаки путём подмены MAC-адреса шлюза по умолчания в пакетах протокола ARP и возможности по перехвату сетевого трафика.

## Теоретическая часть

***Протокол ARP***

ARP (англ. Address Resolution Protocol — протокол определения адреса) — протокол в компьютерных сетях, предназначенный для определения MAC-адреса по известному IP-адресу.

Рассмотрим суть функционирования ARP на простом примере. Компьютер А (IP-адрес 10.0.0.1) и компьютер Б (IP-адрес 10.22.22.2) соединены сетью Ethernet. Компьютер А желает переслать пакет данных на компьютер Б, IP-адрес компьютера Б ему известен. Однако сеть Ethernet, которой они соединены, не работает с IP-адресами. Поэтому компьютеру А для осуществления передачи через Ethernet требуется узнать адрес компьютера Б в сети Ethernet (MAC-адрес в терминах Ethernet). Для этой задачи и используется протокол ARP. По этому протоколу компьютер А отправляет широковещательный запрос, адресованный всем компьютерам в одном с ним широковещательном домене. Суть запроса: «компьютер с IP-адресом 10.22.22.2, сообщите свой MAC-адрес компьютеру с IP-адресом 10.0.0.1». Сеть Ethernet доставляет этот запрос всем устройствам в том же сегменте Ethernet, в том числе и компьютеру Б. Компьютер Б отвечает компьютеру А на запрос и сообщает свой MAC-адрес (напр. 00:ea:d1:11:f1:11) Теперь, получив MAC-адрес компьютера Б, компьютер А может передавать ему любые данные через сеть Ethernet.

Наибольшее распространение ARP получил благодаря повсеместности сетей IP, построенных поверх Ethernet, поскольку практически в 100 % случаев при таком сочетании используется ARP.

***Атаки типа «человек посередине»***

Атака посредника («человек посередине», MITM-атака (Man in the middle)) — термин в криптографии, обозначающий ситуацию, когда криптоаналитик (атакующий) способен читать и видоизменять по своей воле сообщения, которыми обмениваются корреспонденты, причём ни один из последних не может догадаться о его присутствии в канале.

Атака обычно начинается с прослушивания канала связи и заканчивается тем, что криптоаналитик пытается подменить перехваченное сообщение, извлечь из него полезную информацию, перенаправить его на какой-нибудь внешний ресурс.

Предположим, объект A планирует передать объекту B некую информацию. Объект C обладает знаниями о структуре и свойствах используемого метода передачи данных, а также о факте планируемой передачи собственно информации, которую С планирует перехватить. Для совершения атаки С «представляется» объекту А как В, а объекту В — как А. Объект А, ошибочно полагая, что он направляет информацию В, посылает её объекту С. Объект С, получив информацию, и совершив с ней некоторые действия (например, скопировав или модифицировав в своих целях) пересылает данные собственно получателю — В; объект В, в свою очередь, считает, что информация была получена им напрямую от А.

***Подмена MAC-адреса шлюза по умолчанию***

ARP-spoofing (ARP-poisoning) — техника сетевой атаки, применяемая преимущественно в Ethernet, но возможная и в других, использующих протокол ARP сетях, основанная на использовании недостатков протокола ARP и позволяющая перехватывать трафик между узлами, которые расположены в пределах одного широковещательного домена. Относится к числу spoofing-атак.

Протокол ARP является абсолютно незащищённым. Он не обладает никакими способами проверки подлинности пакетов: как запросов, так и ответов. Ситуация становится ещё более сложной, когда может использоваться самопроизвольный ARP (gratuitous ARP).

Самопроизвольный ARP — такое поведение ARP, когда ARP-ответ присылается, когда в этом (с точки зрения получателя) нет особой необходимости. Самопроизвольный ARP-ответ это пакет-ответ ARP, присланный без запроса. Он применяется для определения конфликтов IP-адресов в сети: как только станция получает адрес по DHCP или адрес присваивается вручную, рассылается ARP-ответ gratuitous ARP.

Самопроизвольный ARP может быть полезен в следующих случаях:

* Обновление ARP-таблиц, в частности, в кластерных системах;
* Информирование коммутаторов;
* Извещение о включении сетевого интерфейса.

Несмотря на эффективность самопроизвольного ARP, он является особенно небезопасным, поскольку с его помощью можно уверить удалённый узел в том, что MAC-адрес какой-либо системы, находящейся с ней в одной сети, изменился и указать, какой адрес используется теперь.

## Порядок выполнения работы

1. Соберите сеть в соответствии со схемой, изображённой на рис. 5.1.



Рисунок 5.1 — Схема сети

2. Настройте подключение к сети интернет на компьютере № 1 и откройте общий доступ к этому подключению. Для этого необходимо открыть свойства сетевого подключения (рис. 2) нажав «Пуск» → «Панель управления» → «Центр управления сетями и общим доступом» → «Изменение параметров адаптера», далее нажать правой кнопкой на значок «Подключение по локальной сети» и выбрать пункт контекстного меню «Свойства».

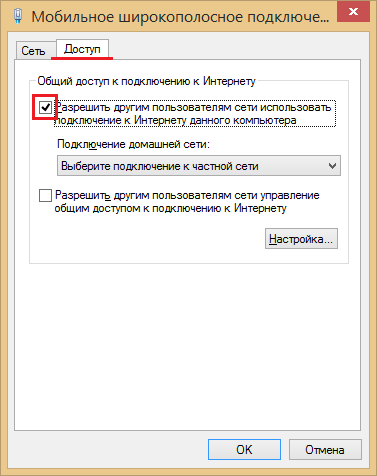


Рисунок 5.2 — Настройка общего доступа к соединению

5. Задайте статические IP-адреса в соответствии со схемой, изображённой на рис. 5 и проверьте наличие доступа в сеть интернет со всех компьютеров в сети. Для этого откройте командную строку («Пуск» → «Выполнить…» или «Win»+R) и введите команды ipconfig и ping 8.8.8.8. Результаты выполнения команд должны соответствовать приведённым на рис. 5.3.

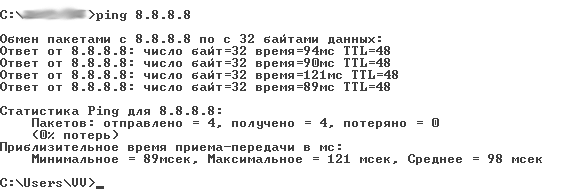


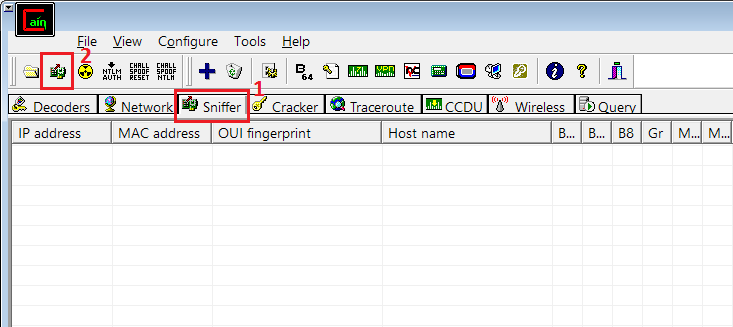
Рисунок 5.3 — Результаты выполнения команды ping

Если все настройки были выполнены верно, то компьютеры локальной сети должны получить доступ в интернет, причём пакеты будут передаваться в соответствии со схемой, приведённой на рис. 5.4.



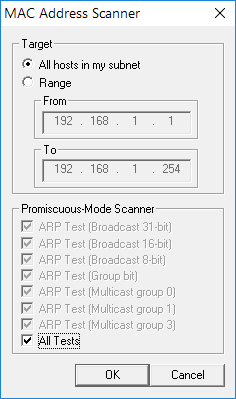
Рисунок 5.5 — Схема прохождения пакетов.

6. На компьютере № 3 откройте сетевую утилиту Cain. В открывшемся окне перейдите на вкладку «Sniffer», запустите сниффер нажав на кнопку «Start/Stop Sniffer».

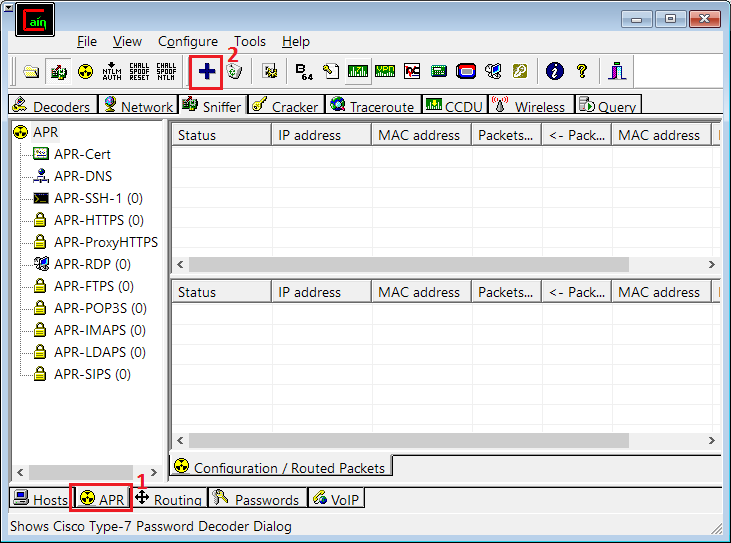


Затем в открывшемся окне выберите сетевой интерфейс компьютера и нажмите «Ok».

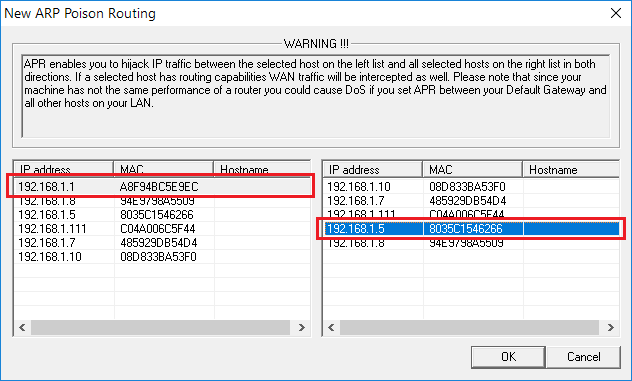
После этого, в основном окне программы нажмите правой кнопкой и в появившемся контекстном меню выберите пункт «Scan MAC Addresses». В появившемся окне можно указать диапазон сканируемых адресов или оставить выбор по умолчанию, при котором будут просканированы все узлы в данной подсети, а в разделе «Promiscuous-mode Scanner» необходимо выбрать пункт «All Tests» и нажать «Ok».



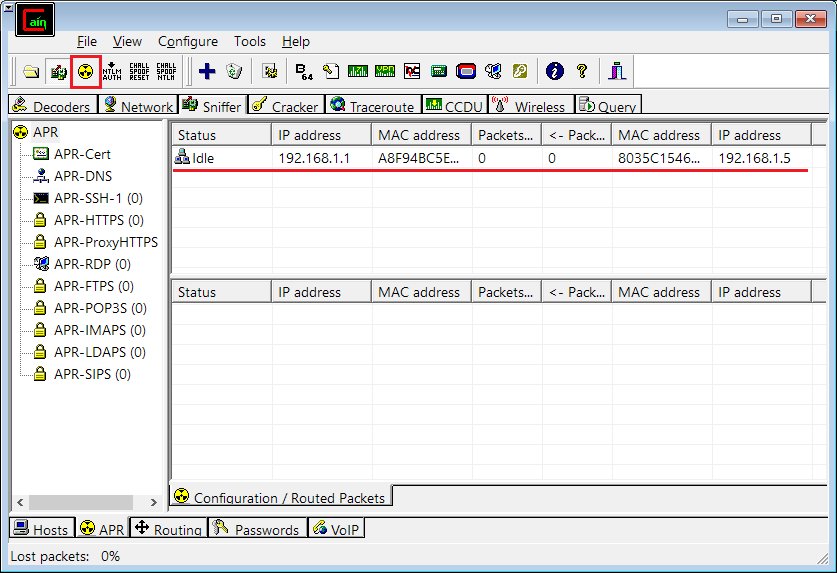
В результате сканирования должны найтись MAC- и IP-адреса всех узлов внутренней (локальной) сети.



7. Перейдите на вкладку ARP и нажмите кнопку «Add to list». Появится окно со списком компьютеров, в котором необходимо выбрать, в передачу данных между какими компьютерами будет встраиваться данная программа.



8. Выберите слева адрес компьютера № 1, а справа компьютера № 2 и нажмите кнопку «Start/Stop ARP».



9. На компьютере № 3 запустите Wireshark.

10. На компьютере № 2 запустите веб-браузер, откройте, например, сайт ntb.donstu.ru и введите логин/пароль в форму авторизации.

11. На компьютере № 3 в окне программы WireShark обратите внимание на пополнившийся список захваченных пакетов, прошедших через этот компьютер. Найдите среди них пакет содержащий строку «username=… &password=…» в которой указаны логин и пароль, введённые на сайте.

## Контрольные вопросы

1. Какие функции выполняет протокол ARP?
2. В каких случаях компьютер посылает ARP-запрос?
3. Что такое самопроизвольный ARP?
4. Как происходит подмена записей в ARP-таблице?

# Лабораторная работа № 6 – Защита от атаки ARP-spoofing

## Цель работы: изучить возможности технологии с использованием Dynamic ARP Inspection для защиты от атаки путём подмены MAC-адресов в пакетах протокола ARP.

## Теоретическая часть

***Технология Dynamic ARP Inspection***

Dynamic ARP Inspection — функция коммутатора, предназначенная для защиты от атак с использованием протокола ARP.

Чтобы эффективно предотвращать ARP-poisoning, коммутатор должен иметь информацию о связке MAC-адрес/IP-адрес. Данная информация хранится в таблице DHCP snooping, поэтому для корректной конфигурации эти две технологии практически всегда использутся вместе.

Для правильной работы Dynamic ARP Inspection, необходимо указать какие порты коммутатора будут доверенными (trusted), а какие — недоверенными (untrusted):

Если порт недоверенный, то коммутатор:

— перехватывает все ARP-запросы и ARP-ответы прежде чем перенаправлять их;

— проверяет соответствие MAC-адреса IP-адресу на ненадёжных портах.

Проверка соответствия MAC-адреса IP-адресу может выполняться на основе:

— статических записей,

— базы данных, полученной в результате работы DHCP-snooping.

После активации DHCP snooping, коммутатор начинает следить за DHCP пакетами в сети и запоминать соответствие выданных IP адресов с MAC-адресами запрашивающих устройств, в таблицу DHCP snooping binding. Таким образом Dynamic ARP Inspection не работает в сетях без DHCP.

## Порядок выполнения работы

1. Соберите сеть в соответствии со схемой, изображённой на рис. 6.1. В качестве коммутатора используйте Cisco Catalyst 2950 или Cisco Catalyst 2960.



Рисунок 6.1 — Схема сети.

2. На компьютере 1 настройте и запустите DHCP-сервер в соответствии с инструкциями из лабораторной работы № 2 «Сетевая атака DHCP-spoofing».

3. На коммутаторе настройте функцию DHCP-Snooping в соответствии с инструкциями из лабораторной работы № 3 «DHCP-spoofing в Cisco Packet Tracer и защита от него».

4. Выполните атаку ARP-Poisoning в соответствии с инструкциями из лабораторной работы № 4. Убедитесь, что её выполнение возможно.

5. Настройте на коммутаторе функцию Dynamic ARP Inspection. Используйте для этого следующие команды:

Включение для указанного VLAN:

Switch(config)# ip arp inspection vlan 1

Настройка доверенного порта:

Switch(config)# interface Fa0/1

Switch(config-if)# ip arp inspection trust

Посмотреть запомненные коммутатором в процессе выдачи DHCP-сервером сопоставления IP и MAC адресов можно командой:

Switch# show ip dhcp snooping binding

6. Убедитесь, что атаку ARP-Poisoning выполнить больше невозможно.

## Контрольные вопросы

1. Какие существуют способы защиты от ARP-poisoning?
2. Что такое Dynamic ARP Inspection?
3. Какие условия должны выполняться, чтобы можно было использовать Dynamic ARP Inspection?
4. На каких устройствах настраивается Dynamic ARP Inspection?

# Лабораторная работа № 7 – Подмена маршрутов в протоколе RIP

## Цель работы: изучить принципы и методы программной реализации внесения ложных записей в таблицы маршрутизации устройств, работающих по протоколу RIP.

## Теоретическая часть

Протокол маршрутной информации (англ. Routing Information Protocol, RIP) — один из простых протоколов маршрутизации, который применяется в небольших компьютерных сетях и позволяет маршрутизаторам динамически обновлять маршрутную информацию. Относится к дистанционно-векторным протоколам.

Будучи одним из наиболее ранних протоколов обмена маршрутной информацией, он до сих пор чрезвычайно распространен в вычислительных сетях ввиду простоты реализации. В настоящее время протокол RIP для IP представлен двумя версиями. RIP v.1 не поддерживает маски, т. е. он распространяет между маршрутизаторами информацию только о номерах сетей и расстояниях до них, но не о масках этих сетей, считая, что все адреса принадлежат к стандартным классам A, B или С. RIP v.2 передает данные о масках сетей.

В качестве расстояния до сети стандарты протокола RIP разрешают использовать различные виды метрик: число транзитных узлов, пропускную способность, вносимые задержки и надежность сетей, а также любые их комбинации, однако чаще всего на практике в качестве метрики используется именно количество транзитных узлов, то есть промежуточных маршрутизаторов.

На рисунке ниже представлена структура RIP-пакета для версии 1. В нём содержится следующая информация:

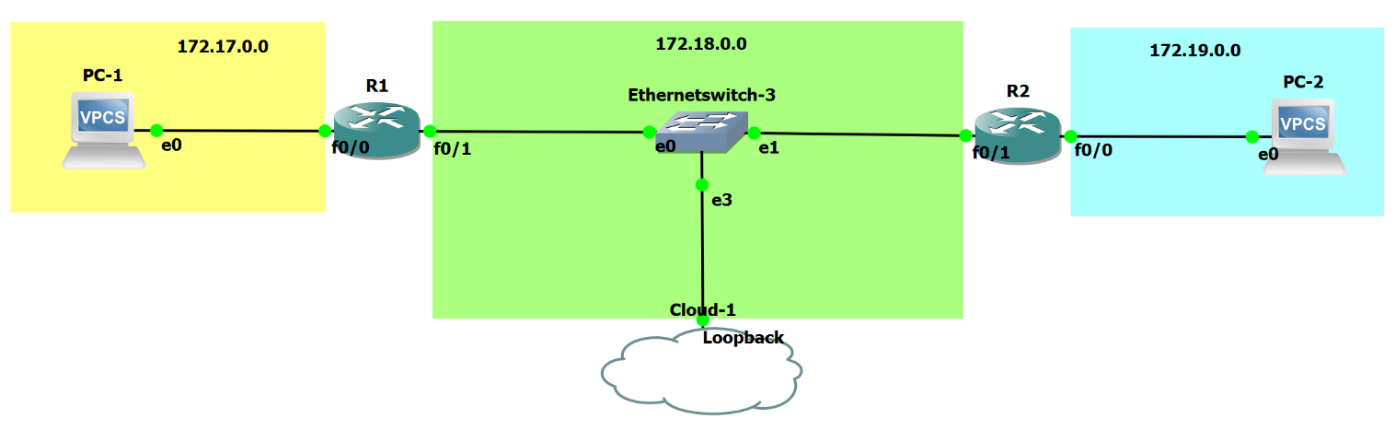
* Command — команда, определяет назначение пакета: 1 — request (запрос); 2 — response (ответ).
* Version — номер версии, в зависимости от версии, определяется формат пакета.
* Routing Domain — идентификатор RIP-системы, к которой принадлежит данное сообщение; часто — номер автономной системы. Используется, когда к одному физическому каналу подключены маршрутизаторы из нескольких автономных систем, в каждой автономной системе поддерживается своя таблица маршрутов. Поле использовалось короткое время в версии протокола RIP-2. В протоколе RIP-1 и в текущей версии RIP-2 не используется.



* RIP Entry (RTE) — запись маршрутной информации RIP. RIP пакет может содержать от 1 до 25 записей RIP Entry. Каждая RIP Entry содержит в себе:
  + адрес сети, информация о которой передаётся,
  + её маску (только для версии 2),
  + адрес следующего маршрутизатора (только для версии 2),
  + метрику.

## Порядок выполнения работы

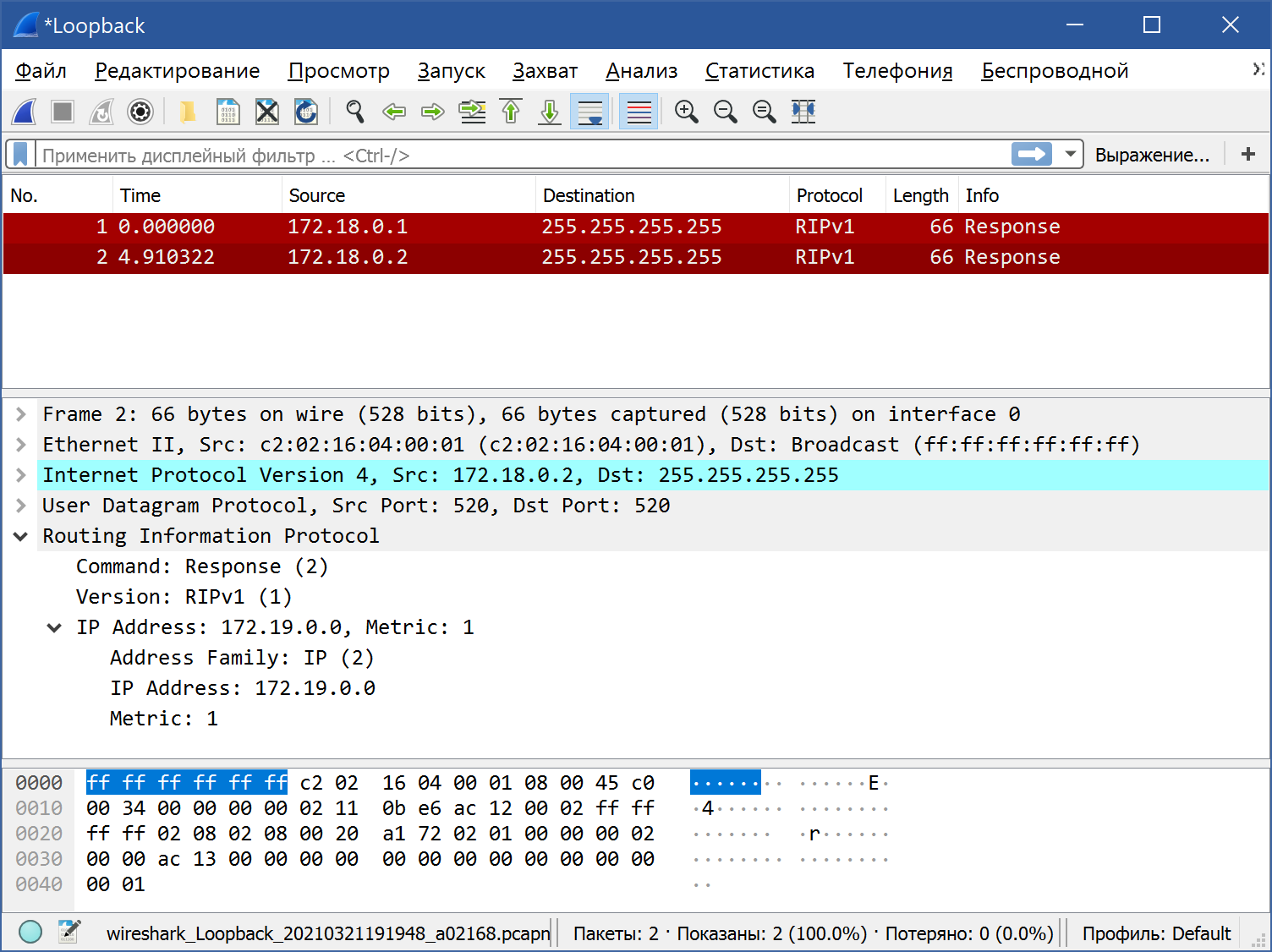
1. В GNS3 постройте сеть, соответствующую изображённой на рисунке.



2. Задайте IP-адреса всем интерфейсам маршрутизаторов и компьютеров на схеме, включая интерфейс реального компьютера (на схеме обозначен как Loopback).

3. Настройте на маршрутизаторах протокол RIP, а на виртуальных компьютерах шлюзы по умолчанию. С помощью команды ping убедитесь, что связь между виртуальными компьютерами работает.

4. На реальном компьютере запустите Wireshark и дождитесь RIP-пакета от какого-либо маршрутизатора. Изучите структуру этого пакета.

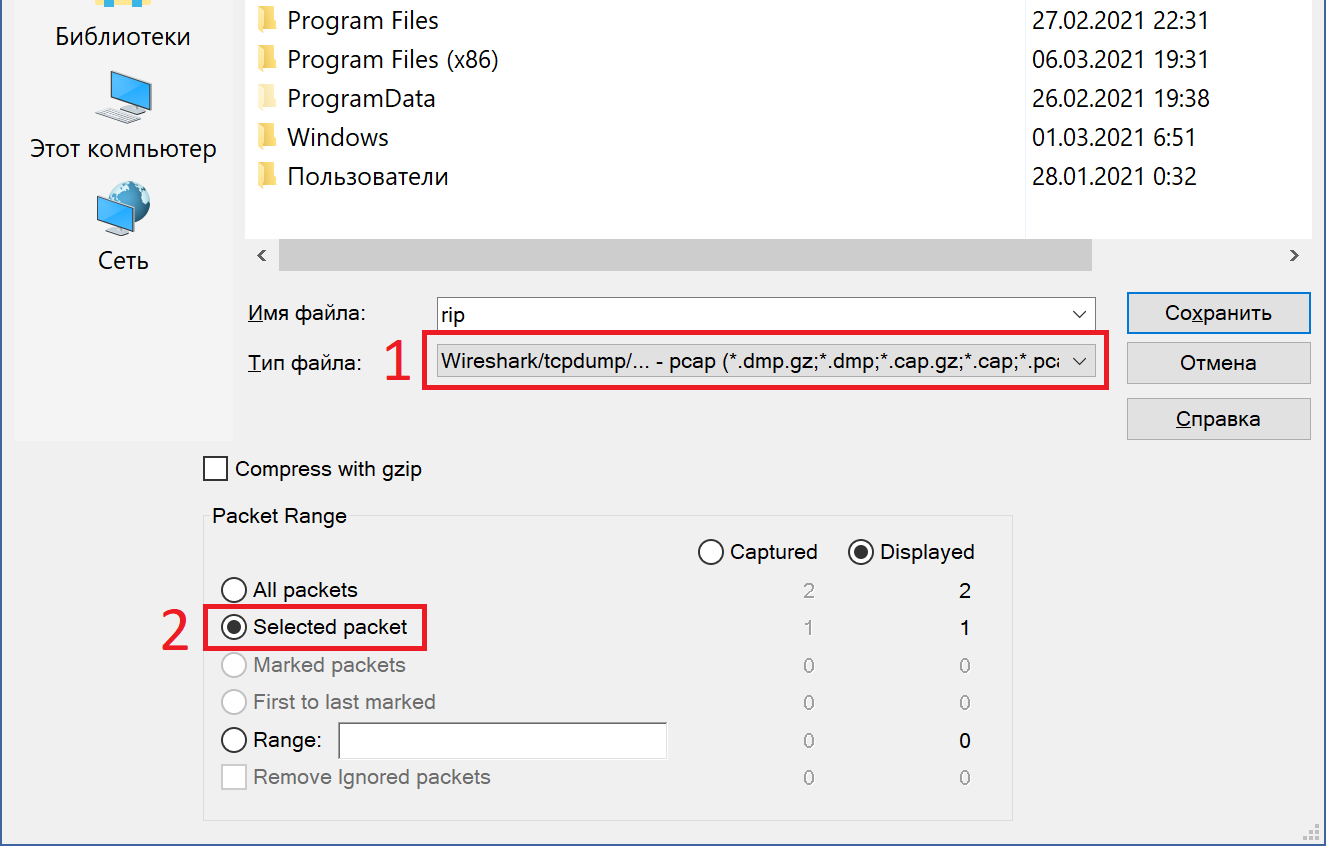


5. Сохраните один из пакетов в файл. Для это в меню «Файл» выберите пункт «Экспортировать Указанные Пакеты…» и в появившемся окне задайте настройки в соответствии в рисунком:

Тип файла: Wireshark/tcpdump/… - pcap

Установите переключатель «Selected packet»

Запомните расположение сохранённого файла.



6. Создайте новый проект в Visual Studio и добавьте к нему библиотеки SharpPCap и Packet.NET. При выполнении этого и следующих пунктов используйте в качестве основы лабораторную работу № 5 «Программные средства анализа трафика» из дисциплины «Информационная безопасность телекоммуникационных систем».

7. Реализуйте в программе загрузку сохранённого пакета и извлечение из него данных, относящихся к протоколу RIP.

7.1 Так как библиотека Packet.NET не поддерживает протокол RIP, то хранить данные RIP-пакета необходимо в обычном массиве байт:

byte[] RIPPacket;

7.2 Для загрузки пакетов из pcap-файла используется метод CaptureFileReaderDevice(filename) из библиотеки LibPcap, которая является частью SharpPCap. Сам объект, отвечающий за загрузку относится к тому же классу ICaptureDevice, что и объект, отвечающий за захват пакетов с сетевого интерфейса:

ICaptureDevice loadDevice;

loadDevice = new SharpPcap.LibPcap.CaptureFileReaderDevice(@"С:\rip.pcap");

Далее необходимо открыть созданное устройство и связать с ним обработчик события извлечения очередного пакета из файла:

loadDevice.Open();

loadDevice.OnPacketArrival += PacketLoad;

loadDevice.StartCapture();

7.3 После этого нужно создать метод PacketLoad(object sender, CaptureEventArgs e) и внутри него последовательно извлекать данные из считываемых из pcap-файла пакетов:

Packet packet = Packet.ParsePacket(e.Packet.LinkLayerType, e.Packet.Data);

UdpPacket udpPacket = (UdpPacket)packet.Extract(typeof(UdpPacket));

RIPPacket = new byte[udpPacket.PayloadData.Length];

for(int i=0; i< udpPacket.PayloadData.Length; i++)

{

RIPPacket[i] = udpPacket.PayloadData[i];

}

8. Напишите программный код для создания собственного RIP-пакета на основе загруженного. Для этого необходимо сначала заменить необходимые байты в считанных данных, хранимых в массиве RIPPacket.

Сред них наибольший интерес представляют байты со следующими номерами:

[8 – 11] — адрес сети;

[21 – 23] — метрика.

Далее массив RIPPacket нужно поместить в поле PayloadData пакета UDP:

UdpPacket newUdpPacket = new UdpPacket(520, 520);

newUdpPacket.PayloadData = RIPPacket;

Числа 520 обозначают номер порта, закреплённого за протоколом RIP.

Далее этот UDP-пакет нужно последовательно поместить в IP-пакет, а затем в Ethernet-пакет, используя в качестве IP и MAC адресов получателя широковещательные адреса 255.255.255.255 и FFFFFFFFFFFF соответственно.

После формирования пакета его можно отправлять в сеть:

sendDevice.SendPacket(newEthPacket);

9. С использованием созданной программы отправьте виртуальным маршрутизаторам в GNS3 RIP-пакет с маршрутом в сеть 172.19.0.0 и метрикой 0 указав в качестве отправителя адрес реального компьютера.

Признаком успешного выполнения атаки будет появление в таблицах маршрутизации маршрута в сеть 172.19.0.0 через реальный компьютер.

10. Выполните команду ping между виртуальными компьютерами и убедитесь, что пакеты проходят по изменённому маршруту через реальный компьютер.

## Контрольные вопросы

1. Какая информация передаётся в RIP-пакете?
2. Что такое метрика?
3. Как передать пакеты, захваченные Wireshark, в программу на C#?
4. Внутри какого протокола передаются RIP-пакеты?

# Лабораторная работа № 8 – Защита от подмены маршрутов в протоколе RIP и OSPF

## Цель работы: изучить способы защиты от атак, основанных на подмене маршрутов протоколов RIP и OSPF.

## Теоретическая часть

***Защита RIP***

Чтобы предотвратить изменение маршрутов не имеющими на это разрешение RIP-маршрутизаторами в среде с протоколом RIP версии 2, можно настроить интерфейсы маршрутизатора, использующие RIPv2, на простую парольную проверку подлинности. Получаемые объявления RIP с паролями, не совпадающими с заданным, будут отклоняться.

Настройка аутентификации в RIP включает в себя несколько этапов.

1. Создание цепочки ключей:

Router (config)#key chain RIPSec

Router(config-keychain)#key 1

Router(config-keychain-key)#key-string password

2. Включение аутентификации на интерфейсе, направленном в сторону другого маршрутизатора.

Передавать пароли можно в открытом виде и в виде MD5. По умолчанию используется открытая аутентификация, однако для обеспечения необходимого уровня безопасности следует применять исключительно аутентификацию MD5.

Router(config)#interface fastEthernet 0/1

Router(config-if)#ip rip authentication mode md5

Router(config-if)#ip rip authentication key-chain RIPSec

Включить показ отладочных сообщений о работе протокола RIP, в том числе сообщений о проблемах с аутентификацией, можно командой:

Router#debug ip rip

Помимо аутентификации, полезной настройкой, влияющей на безопасность протокола RIP является статическое указание соседства.

По умолчанию протокол RIP распространяет свои объявления с помощью широковещательной (RIPv1 и RIPv2) или многоадресной рассылки (RIPv2). Соответственно их получают все узлы сети, включая компьютер потенциального злоумышленника, что даёт ему широкие возможности по анализу данных обновлений для сбора информации о сети и конкретных настройках протокола RIP. Для того чтобы ограничить отправления обновлений в сети с множественным доступом нужно использовать команду neighbor. После её выполнения маршрутизатор будет отправлять RIP-обновления одноадресными (unicast) пакетами только указанным соседям. Для этого используются следующие команды:

Router(config)# router rip

Router(config-router)# passive-interface FastEthernet0/1

Router (config-router)# neighbor 172.18.0.2

Статическое указание соседства не защищает от подмены маршрутов, оно защищает только от получения таблиц маршрутизации третьими лицами.

***Защита OSPF***

В OSPF используется аутентификация для защиты сети от злоумышленников, рассылающих ложную маршрутную информацию с целью обмануть маршрутизаторы сети и получить возможность перехвата, анализа и модификации данных, либо просто нарушая работу сети. Аутентификация заключается в защите пересылаемой информации заранее внесенным паролем.

Когда аутентификация соседей корректно сконфигурирована, маршрутизатор проверяет подлинность отправителя каждого пакета, содержащего обновление маршрутной информации при помощи обмена ключами аутентификации (или паролями), известными всем участникам, но по умолчанию, маршрутизаторы не используют аутентификацию в OSPF.

OSPF поддерживает 2 типа аутентификации: открытым текстом и MD5 (Message Digest, не использует пароль в открытом виде). Механизм аутентификации с использованием MD5 добавляет в каждый пакет увеличивающийся номер последовательности, защищая тем самым участников обмена от атак, использующих повторную пересылку захваченных ранее пакетов.

Аутентификация в OSPF настраивается отдельно для каждого интерфейса:

Router(config-if)#ip ospf authentication-key password

Router(config-if)#ip ospf authentication message-digest

Включить показ отладочных сообщений о работе протокола OSPF, в том числе сообщений о проблемах с аутентификацией при установлении соседства, можно командой:

Router#debug ip ospf adj

## Порядок выполнения работы

1. Для сети, построенной в предыдущей лабораторной, настройте аутентификацию RIP.

2. Попытайтесь выполнить подмену маршрутов и убедитесь, что это невозможно.

3. Задайте статическое соседство на маршрутизаторах, перехватите передаваемый одним из них пакет и убедитесь, что в качестве адреса назначения в нём не указан широковещательный адрес.

4. Создайте сеть, аналогичную построенной в предыдущей лабораторной и настройте в ней маршрутизацию по протоколу OSPF, а также аутентификацию для OSPF.

## Контрольные вопросы

1. Какие средства используются для защиты от подмены маршрутов в протоколах RIP и OSPF?
2. Что такое хеш (или дайджест) сообщения?
3. Что такое статическое указание соседства? Для чего оно используется?
4. Какой тип аутентификации безопаснее: открытым текстом или MD5?

# Лабораторная работа № 9 – Настройка правил фильтрации трафика в сетевом экране Linux с помощью утилиты iptables

## Цель работы: изучить варианты использования и способы настройки межсетевого экрана Linux с использованием утилиты iptables.

## Теоретическая часть

Iptables — утилита командной строки, которая является стандартным интерфейсом управления работой межсетевого экрана (брандмауэра) ядра Linux. Иногда под словом iptables имеется в виду и сам межсетевой экран.

Все сетевые пакеты, которые проходят через компьютер, отправляются компьютером или предназначены компьютеру, ядро Linux направляет через фильтры iptables. Там эти пакеты поддаются проверкам и затем для каждой проверки, если она пройдена выполняется указанное в ней действие. Например, пакет передается дальше ядру для отправки целевой программе, или отбрасывается.

Все пакеты делятся на три типа: входящие, исходящие и проходящие. Входящие — это те, которые были отправлены на этот компьютер, исходящие — отправленные из этого компьютера в сеть. А проходящие (или транзитные) — это пакеты, которые должны быть пересланы дальше, например, если компьютер выступает в качестве маршрутизатора.

Соответственно в фильтре iptables все пакеты делятся на три аналогичные цепочки:

* INPUT — обрабатывает входящие пакеты и подключения.
* FORWARD — применяется для проходящих соединений. Сюда попадают пакеты, которые отправлены на компьютер, но не предназначены ему.
* OUTPUT — используется для исходящих пакетов и соединений.

Кроме перечисленных выше, есть еще две дополнительные цепочки правил:

* PREROUTING — в эту цепочку пакет попадает перед обработкой iptables, система еще не знает куда он будет отправлен, в INPUT, OUTPUT или FORWARD;
* POSTROUTING — сюда попадают все проходящие пакеты, которые уже прошли цепочку FORWARD.

Названия цепочек всегда пишутся заглавными буквами.

Для каждого типа пакетов можно установить набор правил, которые по очереди будут проверяться на соответствие с пакетом и если пакет соответствует, то применять к нему указанное в правиле действие. Правила образуют цепочку, поэтому input, output и forward называют цепочками правил. Действий может быть несколько:

* ACCEPT — разрешить прохождение пакета дальше по цепочке правил;
* DROP — удалить пакет;
* REJECT — отклонить пакет, отправителю будет отправлено сообщение, что пакет был отклонен;
* LOG — сделать запись о пакете в лог файл;
* QUEUE — отправить пакет пользовательскому приложению.

Правила могут проверять любые соответствия, например, по ip, по порту получателя или отправителя, заголовкам пакета и многому другому. Если пакет не подходит ни одному из правил, то к нему применяется действие по умолчанию, обычно ACCEPT.

Цепочки организованны в 4 таблицы:

* raw — просматривается до передачи пакета системе определения состояний. Используется редко, например, для маркировки пакетов, которые не должны обрабатываться системой определения состояний. Для этого в правиле указывается действие NOTRACK. Содержит цепочки PREROUTING и OUTPUT.
* mangle — содержит правила модификации (обычно заголовка) IP‐пакетов. Среди прочего, поддерживает действия TTL (Time to live), TOS (Type of Service), и MARK (для изменения полей TTL и TOS, и для изменения маркеров пакета). Редко необходима и может быть опасна. Содержит все пять стандартных цепочек.
* nat — просматривает только пакеты, создающие новое соединение (согласно системе определения состояний). Поддерживает действия DNAT, SNAT, MASQUERADE, REDIRECT. Содержит цепочки PREROUTING, OUTPUT, и POSTROUTING.
* filter — основная таблица, используется по умолчанию если название таблицы не указано. Содержит цепочки INPUT, FORWARD, и OUTPUT.

Последовательность прохождения пакета по таблицам и цепочкам показана на рисунке 1.

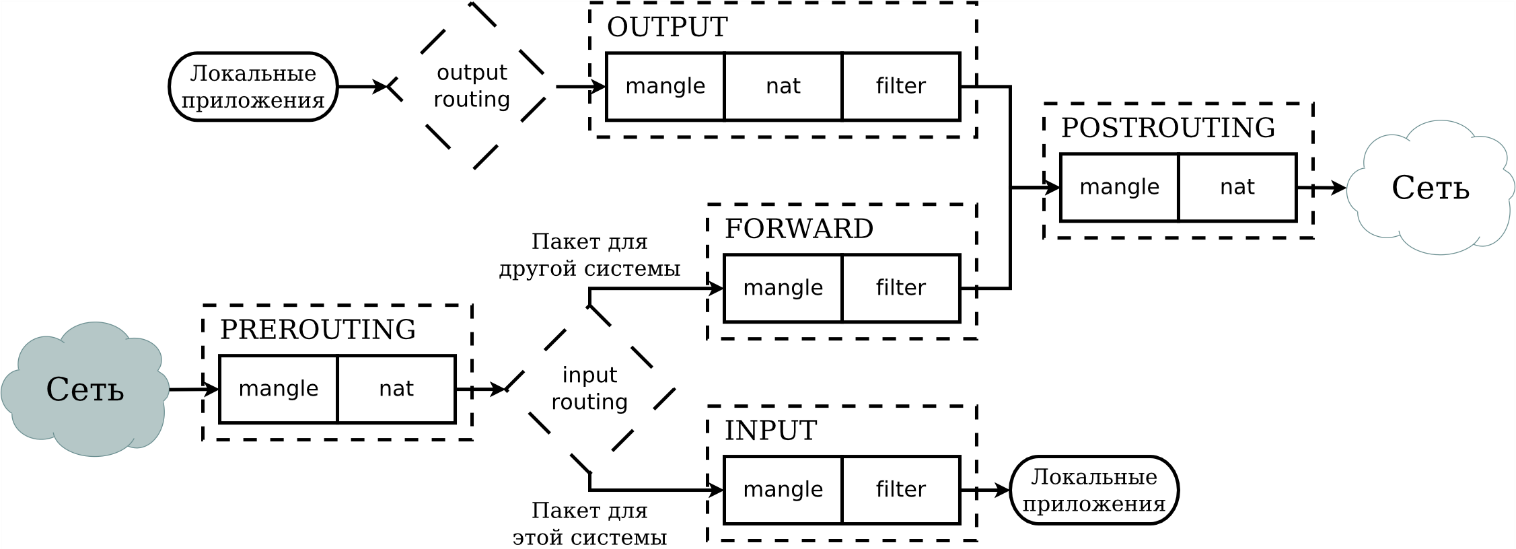


Рисунок 1 — Упрощённая диаграмма прохождения таблиц и цепочек.

Управление правилами, находящимися в цепочках и таблицах, осуществляется из терминала с использованием команд iptables, которые имеют следующую структуру:

iptables таблица операция цепочка критерии действие

Действие может быть одним из следующих:

* -A — добавить правило в цепочку;
* -С — проверить все правила;
* -D — удалить правило;
* -I — вставить правило с нужным номером;
* -L — вывести все правила в текущей цепочке;
* -S — вывести все правила;
* -F — очистить все правила;
* -N — создать цепочку;
* -X — удалить цепочку;
* -P — установить действие по умолчанию.

Дополнительные опции:

* -p — указать протокол, один из tcp, udp, icmp, icmpv6,esp, ah, sctp;
* -s — указать ip адрес устройства-отправителя пакета;
* -d — указать ip адрес получателя;
* -i — входной сетевой интерфейс;
* -o — исходящий сетевой интерфейс;
* -j — выбрать действие, если правило подошло.

Для вывода всех существующих правил iptables используется команда:

iptables –L

а для получения более подробной информации:

iptables –L -v

Полная очистка всех правил выполняется командой:

sudo iptables –F

отдельной цепочки:

sudo iptables -F ЦЕПОЧКА

Если для пакета не подходит ни одно правило, то для него применяется действие по умолчанию. Его можно задать с помощью команды:

sudo iptables -p ЦЕПОЧКА ДЕЙСТВИЕ

Например, чтобы полностью запретить пересылку пакетов, необходимо написать:

sudo iptables -p FORWARD DROP

Ниже приведены несколько примеров команд для решения наиболее типичных задач по фильтрации трафика.

Блокировка ip-адреса:

iptables –A FORWARD –d 8.8.8.8 –j DROP

Блокировка диапазона ip-адресов (подсети):

iptables –A FORWARD –d 8.8.0.0/16 –j DROP

Блокировка протокола (ICMP):

iptables –A FORWARD –p ICMP –j DROP

Блокировка сайта по доменному имени:

iptables -A FORWARD -m string --string "donstu.ru" --algo kmp --to 65535 -j DROP

## Порядок выполнения работы

1. Создайте виртуальную сеть из виртуальных машин в соответствии со схемой, приведённой ниже.



2. Настройте сетевые интерфейсы виртуальных машин следующим образом:

Linux: 192.168.0.1 255.255.255.0

Windows: 192.168.0.2 255.255.255.0

3. Обеспечьте для виртуальной машины с ОС Windows выход в интернет через виртуальную машину с ОС Linux.

3.1. Разрешите в Linux пересылку пакетов, не предназначенных данному компьютеру.

Откройте в текстовом редакторе файл /etc/sysctl.conf с помощью команды:

sudo gedit /etc/sysctl.conf

Найдите строчку:

#net.ipv4.ip\_forward=1

Уберите символ # в начале строки, чтобы раскомментировать её.

Выполните команду для включения трансляции сетевых адресов (NAT):

iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

Где eth0 — название интерфейса через который осуществляется в интернет.

4. С помощью утилиты iptables задайте правила обработки трафика в виртуальной машине с ОС Linux, чтобы обеспечить выполнение перечисленных ниже требований:

* Виртуальная машина с Linux должна отвечать на команды ping и tracert.
* Пакеты, посылаемые командами ping и tracert должны проходить из внутренней сети во внешнюю.
* На виртуальной машине с Linux должна быть возможность создавать общие папки, доступ к которым осуществляется по протоколу SMB (TCP-порт 445). Общие папки должны быть доступны только из внутренней сети.
* К виртуальной машине с Linux из любой сети должен быть доступ по протоколам HTTP и HTTPS.
* Никакой другой трафик не должен приниматься виртуальной машиной с Linux.
* Доступ в интернет должен быть только с компьютеров в сети чьи IP-адреса больше чем 192.168.0.64.
* Со всех компьютеров в сети 192.168.0.0 255.255.255.0, кроме 192.168.0.111 должен быть запрещён доступ к сайтам vk.com и odnoklassniki.ru.

## Контрольные вопросы

1. Для чего нужен межсетевой экран?
2. Какие цепочки имеются во встроенном сетевом экране Linux?
3. В какую цепочку попадают пакеты, проходящие через компьютер, но не предназначенные для него?
4. Из чего состоит правило iptables?
5. Какие параметры можно указывать при создании правила iptables?

# Лабораторная работа № 10 – Настройка прокси-сервера squid в Linux

## Цель работы: изучить функции прокси-сервера Squid по контролю и управлению сетевыми соединениями, а также методы его настройки.

## Теоретическая часть

Squid — это приложение кэширующего прокси сервера, которое предоставляет сервисы кэширования и прокси для HTTP, FTP и других популярных сетевых протоколов.

Squid имеет следующие основные возможности:

— Контроль доступа к сети Интернет. Ограничение доступа для групп или отдельных пользователей, доступ по расписанию, доступ к ограниченному числу ресурсов или создание “запрещённого списка” сайтов - для регуляции доступа сотрудников к “развлекательным” сайтам в рабочее время и защиты от проникновения во внутреннюю сеть посторонних клиентов;

— Контроль трафика. Множество вариантов настройки, позволяющие серверу обеспечивать доступ максимальному количеству клиентов без перегрузки возможностей сетевого оборудования, в том числе отдельные конфигурации для разных видов трафика. Организация открытой точки wi-fi без боязни за возможную нестабильность сети. Возможность выдавать отдельным клиентам или группам ограниченного объема трафика для работы;

— Мониторинг использования сетевых ресурсов. Различные статистические данные, которые собирает Squid, и созданные для него расширения будут незаменимы для оптимизации локальной сети, анализа нагрузки, исследования причин возможных сбоев сети и обнаружения атак на локальную сеть компании;

— Обратное кэширование. Возможность работы в режиме “обратного прокси” или “ускорителя” - для существенного снижения нагрузки на сервер путем кэширования запросов, к которым идет больше всего обращений от пользователей.

Если Squid отсутствует в дистрибутиве Linux, его можно установить командой

sudo apt install squid

Настройка Squid в Linux осуществляется с помощью его конфигурационного файла:

/etc/squid/squid.conf

В Ubuntu Linux для редактирования данного файла необходимо выполнить команду:

sudo nano /etc/squid/squid.conf

Однако, изначально данный файл содержит очень много текста, как правило, закомментированных примеров параметров и пояснения к ним. Для того, чтобы извлечь из этого файла только активные параметры можно выполнить команды:

cp /etc/squid/squid.conf /etc/squid/backup.conf

cat /etc/squid/backup.conf | grep -v '^\(#\|$\)' > /home/student/squid.conf

cp /home/student/squid.conf /etc/squid/squid.conf

В полученном файле squid.conf будут некоторые стандартные настройки. Важной из них является

http\_port число — порт, на котором прокси-сервер принимает соединения (обычно 3128). Этот порт понадобится для настройки клиентских браузеров.

Для того, чтобы изменения конфигурации Squid вступили в силу, используется команда

sudo squid -k reconfigure

Представленная выше команда не перезапускает процесс, а только применяет новые настройки. Если требуется полностью перезапустить прокси-сервер, можно использовать команду

sudo /etc/init.d/squid restart

Основной инструмент настройки Squid — это списки контроля доступа или ACL (Access Control List). ACL объявляются директивой, имеющей следующий синтаксис:

acl имя параметр элементы\_списка

Параметр даёт серверу понять тип элементов списка. АCL с параметром port содержит номера портов, а с параметром src — ip-адреса, с которых на сервер поступает запрос. Например:

acl localnet16 src 172.16.0.0/16

описывает локальную сеть с названием localnet16, адресом 172.16.0.0 и маской /16 или 255.255.0.0.

Следующая важная директива http\_access, имеющая формат:

http\_access действие имя\_acl

определяет действие (разрешить или запретить) с элементами указанного ACL. Например:

http\_access allow localnet16

разрешает выполнение запросов из описанной ранее сети.

В начальной конфигурации Squid содержаться несколько строк со стандартными директивами. Изменять их не рекомендуется. Среди таких директив есть, например, следующие:

acl SSL\_ports port 443

acl Safe\_ports port 80 # http

acl Safe\_ports port 21 # ftp

acl Safe\_ports port 443 # https

Они определяют список портов, часто используемых стандартными приложениями, которые считаются безопасными.

Следующая стандартная директива запрещает доступ для всех небезопасных портов:

http\_access deny !Safe\_ports

это равносильно разрешению доступа для всех безопасных портов.

В ACL можно указывать не только IP-адреса сетей или узлов, но и доменные имена сайтов. Для в качестве параметра указывается «dstdomain», а в качестве элементов списка — URL сайтов. Например:

acl university dstdomain .donstu.ru

Если не поставить точку перед donstu, то правило будет распространяться только на доменное имя donstu.ru. При наличии точки оно также распространяется и на все поддомены данного домена.

Важной особенностью прокси-сервера Squid является возможность аутентификации пользователей. Её настройка включает в себя несколько этапов.

1. Создание файла со списком пользователей и паролей.

1.1. Установка утилит для хеширования паролей:

sudo apt-get install apache2-utils

1.2. Создание файла с паролями:

htpasswd -c /etc/squid/users имя\_пользователя

Ключ -с указывает, что необходимо создать новый файл с паролями. Если файл с таким именем уже существует, то он будет перезаписан.

2. Активация функции авторизации в настройках Squid. Для этого необходимо в файл конфигурации вписать следующие параметры:

auth\_param basic program /usr/lib/squid/basic\_ncsa\_auth /etc/squid/users

auth\_param basic children 5

auth\_param basic credentialsttl 8 hours

acl auth\_users proxy\_auth REQUIRED

3. Для пользователей, для которых необходимы различные правила доступа создать отдельные ACL командой:

acl название proxy\_auth имя\_пользователя

Например, если в файле /etc/squid/users был создан пользователь admin, то для него можно создать следующий ACL:

acl adm proxy\_auth admin

После этого созданные ACL можно будет использовать в директиве http\_access, например:

http\_access allow auth\_users — разрешает запросы от всех авторизованных пользователей.

http\_access allow adm — разрешает запросы от администратора.

Следующей функцией, имеющейся в Squid, является управление скоростью сетевых соединений. Для этого используется механизм Delay Pools (пулы задержек). Пулы задержек позволяют ограничить скорость получаемого прокси-сервером интернет-трафика в зависимости от разных условий, например в зависимости от того, какой пользователь запрашивает этот трафик, в какой подсети он находится, с какого сайта и т.д. В конфигурации Squid по умолчанию механизм пулов задержки выключен, поэтому при доступе к прокси-серверу между клиентами используется простая конкуренция, в результате которой некоторые пользователи могут занимать значительную часть пропускной способности интернет-канала, а некоторые практически не получать её. При этом, объекты, находящиеся в кэше, всегда отдаются клиенту с максимальной скоростью.

Для реализации ограничения скорости в squid используются следующие параметры конфигурационного файла: acl, http\_access, delay\_pools, delay\_access, delay\_parameters, delay\_class.

delay\_pools количество

задает количество delay pools.

delay\_class номер класс

определяет класс для каждого из delay\_pools. Существует несколько классов, у каждого класса существуют соответствующие ему настройки. Для каждого delay\_pools задается единственный класс.

delay\_access номер действие имя\_acl

задает попадание (allow) или не попадание (deny) пользователей в заданный пул (номер\_пула). Обратите внимание, что в данном случае слова allow и deny обозначают не действия (разрешить/запретить), а принадлежность пользователя к пулу задержек. При этом можно задавать не только пользователей, а любые сущности, доступные в acl, например адреса сетей, доменные имена сайтов и др. При этом, проверка принадлежности к данному классу происходит до первого совпадения. Если пользователь по какому-либо параметру совпадает с acl, и для него указано правило allow, то для него применяются параметры текущего пула, если задано правило deny, то на пользователя правила пула не применяются. Если запрос не попал ни в один delay pools то информация отправляется напрямую клиенту без задержек. Синтаксис использования данного параметра аналогичен http\_access.

delay\_parameters номер параметры

устанавливает параметры для указанного номера пула. При этом, в зависимости от класса заданного номера пула, параметры различаются. Для одного delay pools может использоваться только один delay\_parameters.

Существуют следующие классы delay pools:

— Класс 1 содержит ограничения для всех запросов, попадающих в пул.

— Класс 2 имеет ограничения: общие для всех (аналогично первому классу), плюс индивидуальные ограничения для отдельных хостов (биты 25 – 32 сетевого адреса IPv4).

— Класс 3 имеет ограничения: общие, индивидуальные для отдельных хостов (аналогично 2 классу). Плюс ограничения для сетей подсетей (биты 17 – 32 сетевого адреса IPv4).

— Класс 4 имеет ограничения: всё, что определено в классе 3, плюс добавлены ограничения для конкретного авторизованного пользователя.

Команда задания ограничений скорости для пула 4 класса выглядит следующим образом:

delay\_parameters pool total\_rest/total\_max net\_rest/net\_max host\_rest/host\_max user\_rest/user\_max

где:

* pool — номер пула, для которого определяются параметры;
* total — ширина канала на всех;
* net — ширина канала на подсеть;
* host — ширина канала на отдельный адрес;
* user — ширина канала на отдельного авторизованного пользователя;
* rest — скорость заполнения пула в байтах в секунду (значение –1 определяет отсутствие ограничений);
* max — объем пула в байтах.

В процессе обеспечения доступа в интернет прокси-сервер Sqiud может собирать и накапливать довольно информацию об устанавливаемых пользователями сетевых соединениях. Такая информация записывается в log-файл

/var/log/squid/access.log

Однако, в том виде, в котором данные представлены в этом файле, очень сложно анализировать вручную. Поэтому существуют утилиты, автоматизирующие процесс анализа лого Squid и генерирующие на их основе удобную для воcприятия статистику.

Одной из таких утилит является sarg (Squid Analysis Report Generator). Несмотря на то, что sqrg является самостоятельной утилитой и способен генерировать отчёты в виде html-файлов без дополнительных программ, для удобного просмотра полученной статистки в графическом интерфейсе требуется наличие веб-сервера.

Для Linux существует множество программ, способных выполнять функции веб-сервера, однако для простого просмотра отчётов sarg достаточно будет использовать простой веб-сервер lighthttpd.

При его отсутствии в составе операционной системы, установить данный веб-сервер можно командой

sudo apt install lighttpd

Настройка sarg осуществляется также с помощью конфигурационного файла

/etc/sarg/sarg.conf

В нём необходимо задать перечисленные ниже параметры. Некоторые из них уже присутствуют в конфигурационном файле и закомментированы (перед ними стоит символ #), некоторые нужно дописать самостоятельно в любое место файла.

graphs yes

graph\_days\_bytes\_bar\_color orange

output\_dir /var/www/html/squid-reports

charset UTF-8

После выполнения всех настроек необходимо запустить генерацию отчётов с помощью sarg командой

sudo /usr/bin/sarg

Затем можно проверить работоспособность sarg введя в браузер IP-адрес компьютера (или виртуальной машины), на котором установлен sarg и squid.

## Порядок выполнения работы

Настроить прокси-сервер в соответствии с перечисленными требованиями:

1. Для доступа в интернет клиенты должны проходить авторизацию.

Необходимо создать 3 пользователей:

* admin,
* user\_vip,
* user.

2. Должен быть заблокирован доступ к сайтам:

* для user\_vip недоступны развлекательные сайты (несколько на выбор);
* для user — те же развлекательные сайты, что и для user\_vip, а также социальные сети;
* для admin ограничений быть не должно.

3. Должны быть установлены ограничения скорости передачи данных:

* для user\_vip:
* если находится в сети 172.16.0.0, то 2 Мбит/с,
* если находится в сети 172.17.0.0, то 1 Мбит/с;
* для user:
* если находится в сети 172.16.0.0, то 1 Мбит/с,
* если находится в сети 172.17.0.0, то 512 кбит/с;
* для admin ограничений быть не должно;
* для сайта youtube.com у всех пользователей (кроме admin) должно быть ограничение скорости 5 Мбит/с.

4. Необходимо собирать статистику соединений каждого пользователя и настроить её отображение с использованием утилиты sarg и http-сервера lighttpd.

## Контрольные вопросы

1. Что такое прокси сервер?
2. Как выполняется HTTP-запрос через прокси-сервер?
3. В чём разница между NAT и прокси-сервером?
4. Как настраивать прокси-сервер Squid в Ubuntu Linux?
5. По какому принципу реализуются ограничения скорости для пользователей в прокси-сервере Squid?

# Перечень использованных информационных ресурсов

1. Щеглов А.Ю. Защита компьютерной информации от несанкциони-рованного доступа / А.Ю. Щеглов. – СПб.: Издательство «Наука и Техника», 2009. – 384 с.
2. Анин Б.А. Защита компьютерной информации. – СПб.: БХВ-Петербург. 2000. – 384 с.
3. Айвенс К. Компьютерные сети. Хитрости. — СПб.: Питер, 2006. — 298 с.
4. Пятибратов А.П. и др. Вычислительные системы, сети и телекоммуникации. М. 2008. — 284 с.
5. Лапонина О.Р. Основы сетевой безопасности. Учебное пособие. — 2005. URL: http://master.cmc.msu.ru/files/Laponina-1.pdf.
6. Хоффман Э. Безопасность сетевых сервисов. – СПб.: O`REILLY, 2022. – 336 с.
7. Лимончелли Т., Хоган К., Чейлап С. Системное и сетевое админис-трирование. Практическое руководство, 2-е издание. — Пер. с англ. — СПб: Символ-Плюс, 2009. — 944 с.
8. Алексеев В.А. Маршрутизация и защита сетевого трафика в сетях TCP/IP: учебно-методическое пособие. – Липецк: Липецкий государственный технический университет, 2013. – 35 с.
9. Бэндл Д. Защита и безопасность в локальных компьютерных сетях. – СПб.: Питер, 2012. – 480 с.