ПРАКТИЧЕСКАЯ РАБОТА №3

Мониторинг сетевого трафика на хосте на примере работы с утилитами диагностики и мониторинга сетевых соединений в Linux

Цель работы: получить практические навыки по работе с анализаторами сетевого трафика. На практике ознакомиться с различиями в принципах работы активного сетевого оборудования. Уяснить особенности взаимодействия сетевого и канального уровней на примере стека TCP/IP. Выяснить отличия форматов кадров Ethernet. Познакомиться с консольными утилитами диагностики и анализа сетевых соединений.

Необходимо: Компьютер с установленной средой виртуализации Virtual Box. Виртуальные машины Linux. Административные учетные записи на виртуальных машинах. Сетевое подключение по протоколу IP. Доступ к глобальной сети Интернет. Программный пакет Wireshark.

Краткие теоретические сведения

Linux – UNIX-подобная, многозадачная операционная система. Основным для нее является текстовый интерфейс, хотя для Linux разработаны (или портированы) графические оболочки, такие как KDE или Gnome.

Обычно в Linux запускаются несколько консолей, переключаться между которыми можно по кнопкам Alt + F1 для перовой консоли, Alt + F2 для второй и т. д.

Краткую справку по каждой команде можно получить с помощью команды man, краткую с помощью ключа –h (--help). Например: man ifconfig. Также полезными для получения справки могут оказаться команды apropos и whatis.

Если утилиты окажутся недоступны, то их можно установить через менеджера пакетов. Например, так: yum install netload.

Чтобы узнать, к какому пакету относится та или иная утилита можно воспользоваться командой:

yum whatprovides имя_утилиты.

Для диагностики сетевых соединений служит протокол ICMP. Его используют консольные утилиты ping, traceroute, mtr. Эти утилиты позволяют проверять доступность удаленного хоста и диагностировать соединение.

Для мониторинга интерфейсов используются множество утилит. Среди них nload, iftop, bmon. Для сбора статистики канального уровня используются демон vnstat. Диагностировать соединения приложений позволяют такие утилиты как nethogs .

Для анализа соединений с сетевыми сервисами служат утилиты консольные утилиты netstat, ss, lsof, позволяющие получить информацию о открытых и задействованных сетевых сокетах.

Для установления соединений, передачи сообщений и файлов, сканирования портов используется утилита nc из пакета netcat.

Для того, чтобы разрешить запуск службы и запустить ее используются команды:

systemctl enable ИмяСервиса

systemctl start ИмяСервиса

Для перехвата и анализа трафика на отдельном хосте используются программы «Анализаторы трафика», или «снифферы». Эти программы позволяют осуществить перехват всего трафика по выбранному сетевому интерфейсу и его деинкапсуляцию до прикладного уровня. Как правило, они обладают средствами фильтрации и поиска в перехваченном наборе кадров. Наиболее известным кроссплатформенным решением является Wireshark. Самый распространенный консольный сниффер для Linux – tcpdump.

Снифферы предназначены для анализа текущих соединений на хосте и поиск неисправностей при сетевом взаимодействии.

Инструментальные средства:

Утилиты для работы:	ip, ss, lsof, ping, mtr, ping, nload, iftop, bmon, nethogs, traceroute, vnstat, nc, Wireshark
Утилиты работы с текстом:	echo, grep, sed
Редакторы:	vi, nano

Порядок выполнения работы:

Далее описан порядок выполнения работы. Пункты работы, результаты которых прямо или косвенно используются в отчете, помечены знаком **(!)**.

Часть 1. Настройка инфраструктуры

1. Подготовьте две виртуальные машины с ОС Linux. Одну машину назовите c7-1, другой c7-2.
2. На обоих машинах сетевые интерфейсы настройте в режим Сеть NAT с включенным неразборчивым режимом, внутри машин получение адресов – автоматически с DHCP сервера VirtualBox.
3. Определите полученные адреса для машин c7-1 и c7-2.
4. Установите на реальном хосте программу Wireshark (https://www.wireshark.org). Если вы используете WiFi при инсталляции npcap включите поддержку IEEE 802.11 .
5. На хосте c7-1 с помощью утилиты ping проверьте доступность внешней сети.
6. Проверьте на c7-1 наличие перечисленных утилит. В случае, если утилиты, упомянутые в работе отсутствуют на хосте, их следует установить.
    a. bmon (еще есть аналоги nload, iftop)
    b. nethogs
    c. mtr

d. traceroute
e. vnstat
f. nc

Часть 2. Диагностика соединения

1. Познакомитесь с ключами утилиты ping.
2. На машине c7-2 напишите команду ping, которая **(!)** интервалом 10 секунд отправляет 5 пакетов размером 1500 байт на машину c7-1
3. Выясните что означат использование ключа -f (используйте его **только** при использовании утилиты ping между хостами c7-1 и c7-2)
4. Познакомитесь с ключами утилиты mtr. С ее помощью с хоста c7-1 соберите статистику соединения с хостом www.itmo.ru.
5. Определите значение всех параметров, выводимых утилитой mtr.
6. Напишите команду, которая сохранит в файл расширенную статистику работы mtr при отправке 40 пакетов (!).

Часть 3. Работа с Wireshark

1. Настройте перехват трафика на реальном интерфейсе, так чтобы он завершился после сбора 5 Мб (для увеличения интенсивности генерации кадров открыть любой сайт в браузере).
2. Используя инструментарий статистики, определите **(!)**:
   a. Узел с максимальной активностью (по объему переданных данных),
   b. Узел, осуществивший наибольшее количество широковещательных рассылок,
   c. Самый активный TCP-порт на хосте (по количеству переданных пакетов)
   d. Постройте на одной координатной сетке постройте графики интенсивности TCP и UDP трафика (пункт Io Graphs).
   e. Постройте диаграмму связей только для пакетов, содержащих сообщения протокола HTTPS (пункт Flow Graph)
3. Напишите фильтры, которые выделяют из общего числа пакеты **(!)**:
   a. Отбирающие сообщения протокола DNS (53 порт udp и tcp) относящиеся **только** к взаимодействию DNS клиента на хосте и внешних серверов. То есть в случае, если на вашем компьютере будет запущен и DNS-сервер, фильтр должен отбирать только трафик от и к DNS-клиенту, игнорируя трафик от и к DNS-сервера. Для генерации DNS запросов на виртуальной машине можно использовать утилиту dig (dig www.itmo.ru).
   b. Все кадры Ethernet, отправленные с сетевого интерфейса хоста.
   c. Напишите фильтр, отбирающий только широковещательные сообщения. Определите назначение 3-х широковещательных рассылок разных протоколов (или тех, которые удалось обнаружить).

4. На основании анализа адресов отправителя и получателя в перехваченных пакетах, их вида и распределения, определите к какому типу коммутационного оборудования подключен используемый компьютер (концентратор, коммутатор или маршрутизатор).

Часть 4. Определение маршрута прохождения пакета

1. Познакомитесь с ключами утилиты traceroute.
2. На машине c7-1 напишите команды traceroute, которые **(!)**:
    a. определяют маршрут до хоста 8.8.8.8 с помощью ICMP
    b. определяют маршрут до хоста 8.8.8.8 с помощью UDP
    c. определяют маршрут до хоста 8.8.8.8 с помощью TCP
    d. позволяют определить используется ли по маршруту фрагментация IPv4

Часть 5. Текущий мониторинг сетевых интерфейсов

1. С хоста c7-2 запустите отправку запросов утилитой ping в режиме flood на внутренний интерфейс c7-1.
2. На хосте c7-1 последовательно с помощью утилиты bmon или ее аналогов получите данные о загрузке интерфейса, на который отправляет трафик хост c7-2 **(!)**.
3. Изменяйте размер пакета, передаваемой утилитой ping пакета от 100 до 60100 с шагом 10000. Определите, как меняется загрузка на сетевом интерфейсе **(!)**.

Часть 6. Сбор статистики о загрузки сетевого интерфейса

1. На хосте c7-1 запустите демон vnstat.
2. Поставите на мониторинг интерфейс, через который машина c7-1 подключена к c7-2
3. С хоста c7-2 запустите отправку запросов утилитой ping в режиме flood, так чтобы работа утилиты прекратилась после отправки 500 пакетов.
4. Выведите статистику собранного трафика **(!)**.

    *Примечание: На Centos 7 vnstat не обновляет базу данных сам. Это надо делать вручную или добавить запуск в крон (*/5 * * * * /usr/bin/vnstat -u >/dev/null 2>&1)*

Часть 7. Диагностика работы приложений через сеть

1. Установите несколько соединений с SSH сервером на хосте c7-1 с хоста c7-2. Для простоты можно открыть несколько физических консолей или запускать ssh клиент в скрипте, передавая пароль в явном виде с помощью утилиты sshpass (sshpass -p MyPlainPassword_DontBeatMeSecurityMamager ssh username@host_address). Никогда не поступайте так в реальной жизни! Если нужно используйте

аутентификацию по ключам.

2. Используя утилиту netstat или lsof на c7-1 вывести все активные (прослушиваемые) порты. **(!)**
3. Используя утилиту netstat или ss все установленные соединения **(!).**
4. Напишите скрипт, которой выводит список IP-адресов и количество подключений с них к нашему хосту через порт, задаваемый параметрами скрипта (значение по умолчанию 22). Список упорядочить по количеству соединений с IP адреса. Ради большей наглядности результатов вы можете дополнительно подключиться по SSH к c7-1 с основного хоста или с дополнительных виртуальных машин. Для выполнения задания вам могут понадобиться утилиты grep, awk, cut, sort и uniq, но в выборе инструментов вы не ограничены. **(!)**
5. Закройте все соединения по ssh с хостом c7-1.
6. Познакомитесь с ключами утилиты nethogs.
7. С хоста c7-2 подключитесь по ssh к машине c7-1. В терминале ssh запустите утилиту top.
8. На хосте c7-1 с помощью утилиты nethogs определите **(!)**
    a. Среднюю скорость передачи данных до sshd.
    b. PID процесса sshd.

Часть 8. Работа с утилитой nc (NetCat)

1. На машине c7-1 на отдельной консоли запустите tcpdump для сбора всего трафика с портов 9999 и 4444, так, чтобы на консоль выводилось **содержимое сообщения**, а не только информация из служебных заголовков (!).
2. Используя утилиту nc на обоих машинах передайте текстовый файл с произвольным текстовым содержимым (не мнее 20 слов) принимая файл на порту tcp 9999 (!).
3. Используя утилиту nc на обоих машинах организовать текстовый чат между машинами через порт udp 4444.
    a. Hi! How are you?
    b. Fine! And You?
    c. So am i!

    Завершите сессию (Cntrl+C) (!).

    Примечание: учтите, что, если у вас работает firewall, нужно будет его выключить (что плохо) или добавить разрешения на порты (что хорошо). Так, для FirewallD это можно сделать так: firewall-cmd --permanent --add-port=НОМЕР_ПОРТА/tcp.

4. Остановите работу tcpdump, проанализируйте перехваченные сообщения. Какие выводы можно сделать?

    Примечание: вывод tcpdump можно направить в файл с помощью ключа -w. Это будет файл стандарта pcap, который можно открыть в Wireshark для удобного анализа.

5. Этот пункт выполняется по желанию. С помощью nc можно организовать reverse shell. На машине с Linux Centos 7 с помощью ключа -e запустите команду /bin/bash с перенаправлением вывода-ввода на порт tcp 4445, так же как вы делали для организации чата. Со второй Linux машине подключитесь к порту 4445 и позадавайте команды bash, например получите версию ядра, адрес или hostname.

Содержание отчета

Требуется подготовить отчеты в формате DOC\DOCX или PDF. Отчет содержит титульный лист, артефакты выполнения и ответы на вопросы и задания.

Артефакты:

1. Тексты команд, консольный вывод и полученный файл из Части 2. п. 2,6
2. Графики, тексты фильтров и ответы на вопросы из Части 3. п. 2-3.
3. Тексты команд и консольный вывод из Части 4, п.2.
4. Тексты команд и консольный вывод из Части 5, п.2-3.
5. Тексты команд и консольный вывод из Части 6, п.4.
6. Тексты команд и консольный вывод (или его часть) из Части 7, п.2, 3, 8 и скрипт из п.4.
7. Тексты команд из части 8, п. 1-3, и, если выполнялся, п.4

Вопросы и задания:

1. По какому протоколу работает утилита mtr? Как это можно определить?
2. Опишите значения столбцов статистики, выводимой утилитой mtr. Какие еще статистики доступны в mtr кроме основных?
3. Какие типы кадров Ethernet бывают, в чем их отличия?
4. Какой тип кадров Ethernet используется в анализируемой сети? Почему именно его применение позволяет сети функционировать?
5. Как можно определить тип используемого коммутационного оборудования, используя сетевую статистику? Сделайте предположения о типе коммутационного оборудования использовался в сети на основании собранного трафика.
6. На какие адреса сетевого уровня осуществляются широковещательные рассылки?
7. На какой канальный адрес осуществляются широковещательные рассылки?

8. Для чего применяются перехваченные широковещательные рассылки в Части 3?
9. В Части 4 при разном использовании утилиты traceroute вы получили разные данные. Почему?
10. Как изменяется загрузка интерфейса в Части 5. п. 3? Почему?
11. Какие выводы вы сделали в Части 7, п.4?
12. На каком уровне модели OSI работает vnstat?

Понятийный минимум по работе

1. Broadcast трафик, адреса, назначение
2. Утилиты traceroute и mtr, смысл выводимых значений
3. Утилиты lsof, netstat, ss. Получение информации о прослушиваемых портах, об активных соединениях.
4. Понятие сокета
5. Инкапсуляция при передаче сообщений.
6. MAC адрес.
7. Простые фильтры по адресам и портам в Wireshark и tcpdump

Отчет выслать в течение 4-х недель на адрес edu-net@yandex.ru. В теме письма: №группы ФИО (латинскими буквами) №работы (например: 5555 Fedor Sumkin 3)

Поддержка работы

Дополнительные материалы по теме курса публикуются на Telegram-канале ITSMDao (t.me/itsmdao). Обсуждать работу и задавать вопросы можно в чате ITSMDaoChat (t.me/itsmdaochat).