Шамшин Ю.В.

Методические указания к лабораторной работе

Порядок разрешения адресов узлов в сети.

1. Цель работы.

В ip-сетях используется три типа сетевых адресов узлов: mac-адрес, сетевой адрес и доменное имя. Они используются на разных уровнях сетевой модели для идентификации хостов.

Рассмотреть схему адресации узлов в ip-сетях. Получить представление о порядке разрешения адресов, используемых на различных уровнях стека TCP/IP. При разрешении имён на разных уровнях задействуются файл hosts, протоколы dns и arp

2. Задание к работе.

1. Определить физический (MAC) и сетевой (IP) адреса локального хоста и его доменное имя (DNS).

2.1. Протокол arp.

2. Просмотреть таблицу преобразования физических адресов arp -a. Сохранить полученную информацию в файле **arp2.txt**.
3. Командой ping проверить доступность следующих узлов:
   - 127.0.0.1;
   - locahost;
   - example.com;
   - двух-трех соседних компьютеров;
   - шлюза локальной сети.
4. Просмотреть таблицу преобразования адресов arp, сохранить результат в файле **arp4.txt** и сравнить ее с результатами, полученными в задании 1.
5. Сделать перерыв в сетевой активности на несколько минут, после которого повторить предыдущий пункт, сохранить результат в файле **arp5.txt**.

Пункты в рамке выполняются от имени суперпользователя.

6. Добавить в таблицу *статическую* запись arp –s nnn.nnn.nnn.nnn xx-xx-xx-xx-xx-xx (действительные аппаратный (x) и сетевой (n) адреса одной из соседних машин).
7. Выполнить ping добавленного в предыдущем пункте сетевого адреса.
8. Добавить в таблицу arp следующие записи (пары "mac-адрес — ip-адрес"):
   - действительный mac-адрес — недействительный сетевой адрес;
   - недействительный mac-адрес — действительный сетевой адрес;
9. Проверить доступность (ping) добавленных узлов. Объяснить полученные результаты.
10. Просмотреть таблицу arp и сохранить ее в файле **arp10.txt**.
11. Перезагрузить компьютер просмотреть кэш arp и сохранить в **arp11.txt**. Сравнить с результатами в п.10. Что стало с записями, добавленными в заданиях 6 и 8?

2.2. Файл hosts.

12. Добавить в файл hosts (путь к файлу в UNIX: /etc/hosts, в ОС Windows: %systemroot%\System32\Drivers\etc\hosts) следующую запись (где ip соответствует адресу одной из соседних машин локальной сети):

    nnn.nnn.nnn.nnn abra.cadabra

13. Выполнить ping узла abra.cadabra
14. Определить по arp таблице mac-адрес узла abra.cadabra.
15. Добавьте в hosts запись

    nnn.nnn.nnn.nnn microsoft.com

16. Проверьте доступность web-сервера microsoft.com.

2.3. Протокол dns.

17. Определить все ip-адреса (публичные) одного из известных доменов: inbox.lv, ya.ru, google.com или им подобного.
18. Определите имя и ip-адрес первичного DNS-сервера зоны lv.
19. Ответить на контрольные вопросы

3. Указания к работе.

3.1. Преобразование адресов.

Для сопоставления сетевого адреса с аппаратным адресом интерфейса в стеке TCP/IP имеются специализированные протоколы типа *arp* - address resolution protocol, RFC-826 (http://tools.ietf.org/html/rfc826 или http://rfc.com.ru/rfc826.htm). Это позволяет использовать сетевые протоколы стека поверх различных протоколов канального уровня. Все операции преобразования выполняются прозрачно для протоколов верхних уровней. Результаты преобразований кэшируются и сохраняются на некоторый интервал времени, что позволяет не выполнять преобразование при повторном обращении к ранее взаимодействовавшим узлам.

Кэш arp представлен в виде таблицы, заполненной записями примерно такого вида:

"сетевой адрес — MAC-адрес — интерфейс — способ назначения"

Эта таблица формируется *динамически*, при любом сетевом взаимодействии узла. Для просмотра кэша arp используется одноименная команда — arp. Эта же команда позволяет формировать таблицу MAC-адресов *статически*, передавая записи через список аргументов. Команда arp используется как в UNIX, так и в Windows-системах.

Основной способ заполнения таблицы преобразований — динамический, при котором записи добавляются по мере участия узла в сетевом обмене. Это означает, что в отсутствие сетевой активности кэш arp пуст (если не задано статических записей). Для выполнения заданий к этой работе вам необходимо организовать некоторое сетевое взаимодействие. Пожалуй, самым доступным способом для этого является использование команды *ping*.

Команда ping использует протокол ICMP (Internet Control Message Protocol; RFC-792, RFC-1256) для отправки запросов датаграммного типа (ECHO_REQUEST) и ожидает ответ (ECHO_RESPONSE) от запрашиваемого хоста или шлюза.

ECHO_REQUEST — это датаграмма, имеющая заголовок IP и ICMP. Поле данных заполнено некоторым количеством произвольной информации. Для анализа сети выполняется отправка определенного количество таких датаграмм. По результатам анализа можно судить о доступности запрашиваемого хоста и некоторых аспектах работы сети в целом.

Обязательным параметром команды ping является сетевой адрес узла, заданный в числовом виде или в символьном представлении:

$ ping 192.0.32.10
$ ping example.com

Если задан символьный адрес, то ping попытается выполнить преобразование символьного имени в сетевой адрес. Для этого сначала будет перечитываться содержимое файла *hosts*, который является своего рода сервером DNS в масштабе отдельно взятого сетевого узла. Содержательно файл hosts — обычный текстовый файл, где прописано соответствие ip-адресов доменным именам. Его основное назначение — *ускорить преобразование имен компьютеров в сетевые адреса*. Формат файла:

```
#ip-address  hostname    aliaces
x.x.x.x      hostname    [aliace1 [aliace2 [... [aliaceN]]]]
```

Обычно в этом файле содержится единственная запись:

```
127.0.0.1       localhost
```

Если требуемое имя узла найдено в файле hosts, то возвращается соответствующий ему сетевой адрес. Иначе — выполняется запрос к внешнему серверу DNS, указанному в настройках сетевого интерфейса.

3.2. Как узнать MAC-адрес и ip-адрес своего узла?

Чтобы узнать физический адрес локального хоста и его ip-адрес нужно выполнить команду ifconfig в UNIX или ipconfig в Windows. Запущенная без параметров, команда ifconfig отображает информацию об имеющихся в системе сетевых интерфейсах и их физических и сетевых адресах:

```
$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:1D:92:A2:90:E7
          inet addr:192.168.1.250  Bcast:192.168.255.255  Mask:255.255.0.0
          inet6 addr: fe80::21d:92ff:fea2:90e7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:811957 errors:0 dropped:0 overruns:0 frame:0
          TX packets:446207 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:596559482 (568.9 Mb)  TX bytes:114698114 (109.3 Mb)
          Interrupt:28 Base address:0xe000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:24226 errors:0 dropped:0 overruns:0 frame:0
          TX packets:24226 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:50861906 (48.5 Mb)  TX bytes:50861906 (48.5 Mb)
```

3.3. Как узнать доменное имя своего узла?

Узнать доменное имя локального хоста можно командой hostname.

3.4. Как узнать адрес сервера DNS своего узла?

Узнать адрес сервера DNS можно разными способами, самый простой в UNIX — посмотреть содержимое файла resolv.conf:

```
$ cat /etc/resolv.conf
```

Расширенную информацию о сервере DNS можно получить используя специальные команды, такие как dig, host, nslookup в UNIX. В ОС Windows можно использовать утилиту nslookup.

Пример использования команды nslookup, жирным выделено имя отвечающего сервера имен (сравните с записью в resolv.conf):

```
$ nslookup
> set type=any
> inbox.lv
Server:         192.168.111.1
Address:    192.168.111.1#53
Non-authoritative answer:
inbox.lv        mail exchanger = 1 mx2.inbox.lv.
inbox.lv        nameserver = ns1.inbox.lv.
inbox.lv        nameserver = ns3.inbokss.com.
inbox.lv        text = "google-site-verification=iSLom3GwbyWs2_MH_Cl07gT5OSiCHgPFf9RRVw3uYc0"
inbox.lv        text = "v=spf1 ip4:194.152.32.7 ip4:194.152.32.80 ip4:194.152.32.81 ip4:194.152.32.82
ip4:194.152.32.83 ip4:194.152.32.84 ip4:194.152.32.85 ~all"
inbox.lv        mail exchanger = 1 mx1.inbox.lv.
Name: inbox.lv
Address: 194.152.32.40
inbox.lv        nameserver = ns2.inbox.lv.
inbox.lv
        origin = ns1.inbox.lv
        mail addr = support.inbox.lv
        serial = 2016091400
        refresh = 172800
        retry = 2048
        expire = 1048576
        minimum = 7200
Authoritative answers can be found from:
> exit
```

4. Отчёт.

4.1. Отчёт представляется в электронном виде.

4.2. Отчёт должен содержать листинги arp2.txt, arp4.txt, arp5.txt, arp10.txt, arp11.txt.

4.3. Результаты полученные в 17 и 18 заданиях.

4.4. Отчёт должен содержать краткие ответы на следующие контрольные вопросы:

4.4.1. Что и почему изменилось в таблице arp после выполнения задания 3?

4.4.2. Пояснить причины изменений (или отсутствия таковых) в таблице arp за время перерыва после задания 5.

4.4.3. Что произойдет, если в таблицу arp добавить две или более записей, в которых одному mac-адресу сопоставлены разные сетевые адреса?

4.4.4. Что произойдет, если в таблицу arp добавить две или более записей, в которых одному сетевому адресу сопоставлены разные аппаратные адреса?

4.4.5. Как отличается "время жизни" динамических и статических записей в таблице arp?

4.4.6. Почему в ip-сетях не используется прямое сопоставление символьного адреса (dns) физическому адресу (mac)?

4.4.7. Что произойдет, если в файл hosts записать два (или более) узла с одинаковыми именами (например, myhost.mydomain), но разными сетевыми адресами, а затем обратиться к ним по имени (например так: ping myhost.mydomain)?

4.4.8. Как можно использовать подмену в задании 16 для защиты узлов в локальной сети?

4.4.9. Какой порядок разрешения имён при выполнении ping www.microsoft.com?