Занятие 19.

Тема: **Сетевое администрирование Linux. ICMP.**

Вид занятия: лекция, практическое занятие.

Учебные вопросы:

1. Протокол ICMP. Типы пакетов.
2. Утилиты ping, traceroute, tcptraceroute.
3. Утилиты управления сетью. Nmap. NatCat. Netstat.

Время: 90 минут

Литература:

1. Cysco systems и др. - Руководство по технологиям объедененных сетей, 3-е издание. : Пер. с англ. - М. : Издательский дом “Вильямс”, 2002. - 1040 с. : ил. - парал. тит. англ.

2. Кирх. О, Доусон Т. - Linux для профессионалов. Руководство администратора сети, второе издание. - СПб.: Питер, 2001. - 496 с.; ил.

Ход занятия.

1. Протокол ICMP (Internet Control Message Protocol, протокол контроля сообщений в интернет) – это Internet-протокол третьего (сетевого) уровня модели OSI, создающий пакеты с сообщениями об ошибках и другой информацией об обработке IP-пакетов.

В протоколе ICMP описаны несколько типов пакетов, таких как:

эхо-запрос – пакет, отправляемый для проверки связи с удаленным узлом

эхо-ответ – пакет, получаемый источником в ответ на пакет типа эхо-запрос

назначение недоступно – получатель такого пакета информируется о недоступности сети/узла/порта назначения.

Редиректт – пакет информирует узел об изменении маршрута до назначения. В современных ОС данный тип пакета игнорируется.

Время вышло – узел-источник получает этот тип пакета, если IP-пакет был уничтожен в связи с обнулением поля TimeToLive.

Остальные типы (а также подтипы) пакетов ICMP можно посмотреть в заголовочном файле /usr/include/linux/icmp.h.

2. Утилита ping служит для проверки факта наличия связи с удаленным узлом, а также надежности и скорости связи. Иногда утилита используется для раскрытия ip-адресов хоста по его имени.

Синтаксис ее таков:

ping [параметры] имя_или_ip-адрес_хоста

С помощью параметра -c вы можете указать количество запросов, которые выполнит ping. Если параметр -c опущен, то ping будет подавать запросы до тех пор, пока не получит сигнал или пользователь не нажмет <ctrl-c>.

```
[gserg@WebMedia linux]$ ping -c3 www.ya.ru
PING ya.ru (213.180.204.8) 56(84) bytes of data.
64 bytes from ya.ru (213.180.204.8): icmp_seq=0 ttl=49 time=26.7 ms
64 bytes from ya.ru (213.180.204.8): icmp_seq=1 ttl=49 time=25.6 ms
64 bytes from ya.ru (213.180.204.8): icmp_seq=2 ttl=49 time=24.7 ms

--- ya.ru ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2019ms
rtt min/avg/max/mdev = 24.741/25.718/26.724/0.809 ms, pipe 2
[gserg@WebMedia linux]$
```

С помощью параметра -f можно проверить быстродействие сети (работает только из под root). При использовании этого параметра ping посылает около 1000 пакетов в секунду.

```
[gserg@WebMedia gserg]$ ping -f 192.168.2.254
PING 192.168.2.254 (192.168.2.254) 56(84) bytes of data.
ping: cannot flood; minimal interval, allowed for user, is 200ms
[gserg@WebMedia gserg]$ su -c "ping -f 192.168.2.254"
Password:
PING 192.168.2.254 (192.168.2.254) 56(84) bytes of data.

--- 192.168.2.254 ping statistics ---
238 packets transmitted, 238 received, 0% packet loss, time 4379ms
rtt min/avg/max/mdev = 0.147/0.177/0.478/0.031 ms, pipe 2, ipg/ewma 18.478/0.179 ms
[gserg@WebMedia gserg]$
```

Параметром -s может быть указан размер пакеты в байтах (не менее 64 и не более 65000), в ином случае размер пакета составит 64 байта.

```
[gserg@WebMedia linux]$ ping -s 65000 192.168.2.254
PING 192.168.2.254 (192.168.2.254) 65000(65028) bytes of data.
65008 bytes from 192.168.2.254: icmp_seq=0 ttl=64 time=12.0 ms
65008 bytes from 192.168.2.254: icmp_seq=1 ttl=64 time=12.0 ms
65008 bytes from 192.168.2.254: icmp_seq=2 ttl=64 time=11.9 ms
65008 bytes from 192.168.2.254: icmp_seq=3 ttl=64 time=11.9 ms

--- 192.168.2.254 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3026ms
rtt min/avg/max/mdev = 11.988/12.029/12.083/0.086 ms, pipe 2
[gserg@WebMedia linux]$
```

Дополнительную информацию можно получить из справочного руководства man.

Команда ***traceroute*** поможет Вам проверить путь до удаленного хоста, выдавая все промежуточные маршрутизаторы. С помощью этой команды становится просто

диагностировать проблему в большой сети, быстро выявляя сбойный участок. Синтаксис команды такой же, как и у ping, за исключением параметров командной строки.

```
[gserg@WebMedia linux]$ traceroute www.ya.ru
traceroute to ya.ru (213.180.204.8), 30 hops max, 38 byte packets
1  ns.edu.vologda.ru (192.168.2.2)  0.281 ms  0.192 ms  0.162 ms
2  80.92.3.3 (80.92.3.3)  1.169 ms  1.257 ms  1.103 ms
3  sv-vol00ra-s2-0.severttk.ru (80.92.3.1)  3.140 ms  3.250 ms  3.111 ms
4   sv-yar00rb-s1-0-0.severttk.ru (80.92.0.69)  34.511 ms sv-yar00rb-s2-5-0.severttk.ru (80.92.0.97)
25.161 ms sv-yar00rb-s1-0-0.severttk.ru (80.92.0.69)  10.718 ms
5  MSK41-F000.145.gw.transtelecom.net (217.150.38.142)  28.066 ms  21.430 ms  31.571 ms
6  MSK41.TRANSTELECOM.NET (193.232.244.211)  25.680 ms  18.921 ms  20.999 ms
     MPLS Label=350 CoS=6 TTL=255 S=1
7  ix2-m9.yandex.net (193.232.244.93)  23.085 ms  21.548 ms  35.076 ms
8  c3-vlan3.yandex.net (213.180.192.171)  25.177 ms  23.104 ms  22.918 ms
9  ya.ru (213.180.204.8)  33.519 ms  23.189 ms  32.496 ms
```

Команда ***tcptraceroute*** практически аналогична предыдущей, однако в качестве базового протокола она использует протокол tcp. Это позволяет проверять доступ к службам tcp и путь до них.

```
gserg@ADM:~$ tcptraceroute smtp.mail.ru 25
Selected device eth0, address 10.52.2.1, port 45217 for outgoing packets
Tracing the path to smtp.mail.ru (194.67.23.111) on TCP port 25 (smtp), 30 hops max
 1  10.52.0.5  0.383 ms  0.231 ms  0.219 ms
 2  b229.vologda.ru (193.19.67.229)  1248.475 ms  345.180 ms  106.562 ms
 3  spb-vgld.ptn.ru (212.48.195.9)  78.138 ms  65.802 ms  157.785 ms
 4  spb-bbn1-ge-2-1-0-88.rt-comm.ru (213.59.5.141)  120.521 ms  114.442 ms  131.549 ms
 5  msk-bgw1-ge1-0-0-0.rt-comm.ru (217.106.0.14)  76.536 ms  177.440 ms  130.447 ms
 6  msk-bgw1-ge1-0-0-0.rt-comm.ru (217.106.0.14)  143.422 ms  90.071 ms  115.207 ms
 7  195.161.165.246  172.163 ms  133.969 ms  110.668 ms
 8  cat03.Moscow.gldn.net (195.239.10.189)  79.129 ms  77.774 ms  93.654 ms
 9  cat12.Moscow.gldn.net (194.186.159.238)  76.082 ms  162.883 ms  77.045 ms
10  mailru-KK12-2-gw.Moscow.gldn.net (195.239.8.90)  105.042 ms  75.285 ms  89.922 ms
11  * * *
12  smtp.mail.ru (194.67.23.111) [open]  78.527 ms  81.314 ms  80.269 ms
```

В большинстве случаев, параметры командной строки не приходится использовать, однако их все же следует изучить с помощью справочного руководства man.

3. Утилиты управления сетью существуют для диагностистики как сетевых проблем, так и диагностики проблем каждого конкретного хоста.

Наиболее популярной утилитой, с помощь которой диагностируются проблемы хостов является nmap, который входит в поставку практически всех современных дистрибутивов операционной системы Linux.

Nmap обладает множеством способностей, но нас в первую очередь интересует его возможности определения активных сервисов удаленного хоста, а также операционной системы.

Полноценно использовать nmap можно только при наличии прав суперпользователя. Вот те параметры nmap, которые на будут интересовать:

-sS – скрытое сканирование TCP-портов

-sT – открытое сканирование TCP-портов

-sU – открытое сканирование UDP-портов

-sP – ping-сканирование (поиск активных хостов в сети)

-sF, -sN, -sX – разные скрытые типы сканирования TCP-портов.

-O – определение типа операционной системы (не работает с -sP)

-p – указание диапазона портов для сканирования (12345, 12345-23333)

-v – подробный вывод.

```
[root@WebMedia linux]# nmap -v -sS -O 192.168.2.84

Starting nmap 3.48 ( http://www.insecure.org/nmap/ ) at 2005-04-23 17:24 MSD
Host 192.168.2.84 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.2.84 at 17:24
Adding open port 135/tcp
Adding open port 139/tcp
Adding open port 445/tcp
The SYN Stealth Scan took 0 seconds to scan 1657 ports.
For OSScan assuming that port 135 is open and port 1 is closed and neither are firewalled
Interesting ports on 192.168.2.84:
(The 1654 ports scanned but not shown below are in state: closed)
```

```
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
Device type: general purpose
Running: Microsoft Windows 95/98/ME|NT/2K/XP
OS details: Microsoft Windows Millennium Edition (Me), Windows 2000 Professional or Advanced Server,
or Windows XP
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=9641 (Worthy challenge)
IPID Sequence Generation: Incremental

Nmap run completed -- 1 IP address (1 host up) scanned in 2.391 seconds
```

Команда NetCat (nc) – это еще более широко используемая команда. Ее предназначение, в первую очередь, это проверка работоспособности сети на конкретном узле и выявление ошибок при передаче. Команда nc может открыть порт и выводить информацию из него на экран (или в файл, другое приложение по каналу). Эту возможность часто используют для проверки работы TCP. Для открытия порта служит ключ -l, а порт можно указать параметром -p:

```
1-я консоль
[root@WebMedia linux]# nc -l -p 25 127.0.0.1
test
test double
 punt!

2-я консоль
[gserg@WebMedia gserg]$ telnet 127.0.0.1 25
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
test
test double
Connection closed by foreign host.
[gserg@WebMedia gserg]$
```

Однако и с помощью nc можно просканировать открытые порты. Правда, nc не обладает таким широким набором возможностей как nmap, и делает сканирование намного медленнее. Для этого используется параметр -z. Параметр -v позволяет нам получить дополнительную информацию:

```
[root@WebMedia linux]# nc -z -v 127.0.0.1 600-700
localhost.localdomain [127.0.0.1] 631 (ipp) open
```

Команда netstat позволяет просмотреть открытые соединения на текущем компьютере. Введенная без параметров, она покажет все соединения, включая открытые сокеты unix. Команда принимает параметры, наиболее часто используемыми являются:

-n – показывает числовые значения номеров портов вместо имени из /etc/services и числовые ip-адреса вместо доменных имен;

-p – показывает pid процесса, использующего соединение;

-t – показывать только TCP-сокеты

-u – показывать только UDP-сокеты

-i – показать список интерфейсов и статистику трафика на них

-l – только слушающие порты

Таким образом, просмотр всех программ, слушающих TCP-порты:

```
root@ADM:/var/mail# netstat  -tpl
Активные соединения с интернетом (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State       PID/Program name
tcp        0      0 localhost:2208          *:*                     LISTEN     4785/hpiod
tcp        0      0 *:print01               *:*                     LISTEN     4864/inetd
tcp        0      0 *:51762                 *:*                     LISTEN     26729/sim
tcp        0      0 *:telnet                *:*                     LISTEN     4864/inetd
tcp        0      0 localhost:ipp           *:*                     LISTEN     4761/cupsd
tcp        0      0 localhost:2207          *:*                     LISTEN     4788/python
tcp6       0      0 *:5800                  *:*                     LISTEN     9066/kded [kdeinit]
tcp6       0      0 *:5900                  *:*                     LISTEN     9066/kded [kdeinit]
tcp6       0      0 *:65005                 *:*                     LISTEN     9519/notes2w
tcp6       0      0 *:57177                 *:*                     LISTEN     9519/notes2w
```

Просмотр всех интерфейсов:

```
root@ADM:/var/mail# netstat  -i
Таблица интерфейсов ядра
Iface MTU Met RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0   1500 0    238445      0      0      0   200095      0      0      0 BMRU
lo    16436 0       282      0      0      0      282      0      0      0 LRU
```