二 Linux 系统管理

实验目的

1、 掌握 Linux 网络管理相关操作；

2、 掌握 Linux 系统管理相关操作；

3、 掌握 Linux 磁盘管理相关操作。

实验环境

安装有 Linux 操作系统的计算机。

实验步骤

1. Linux 网络管理

Linux 操作系统网络管理主要包含网络相关信息查看、网络配置、网络监控能内容，如今部分 Linux 提供了图形化界面配置方式，但通过远程终端配置仍通过命令修改。

（1） 设置静态 IP 地址

不通 Linux 发行版配置静态 IP 不同，且差异较大，修改 IP 时需确定系统类型。

<1> RedHat/CentOS 系列设置静态 IP

首先需通过 ifconfig 命令查看网卡相关信息，找到网卡名，如下图所示，本实验实例中网卡名为“ens33”。

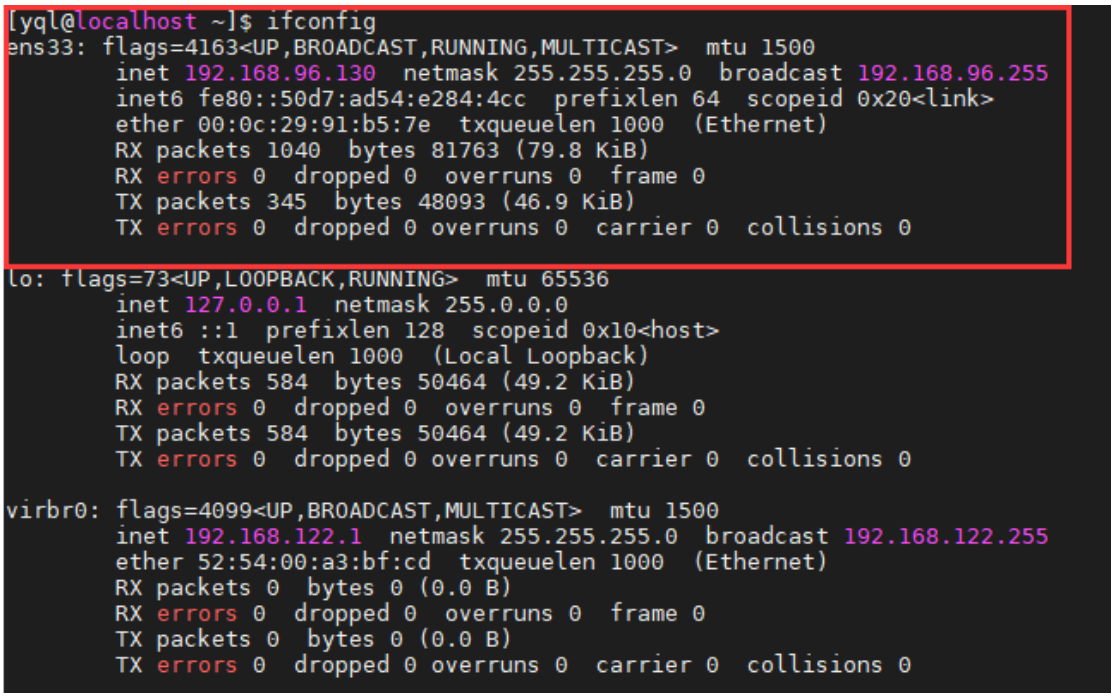

图 3.1 ifconfig 查看网卡信息

通过“vi”命令编辑网卡配置文件，如下图所示为网卡配置文件路径，图中“ifcfg-ens33”即对应上图中的网卡。

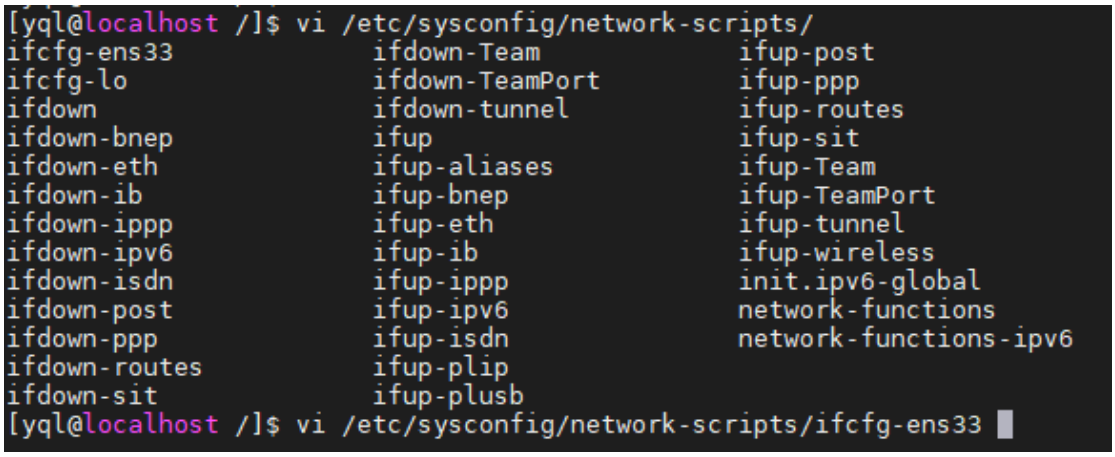

图 3.2 网卡配置文件路径

如下图所示，为网卡 DHCP 模式下的配置文件。

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=b55a2500-28b7-4b38-9594-594a56d9e175
DEVICE=ens33
ONBOOT=yes
```

图 3.3 DHCP 模式下的网卡配置文件

修改本机 IP，如本实例中设置静态 IP 为“192.168.96.130”，请对比上图与下图中的修改部分。

编辑配置文件可通过“vi”命令，保存配置后需重启网络服务。

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=b55a2500-28b7-4b38-9594-594a56d9e175
DEVICE=ens33
ONBOOT=yes

IPADDR=192.168.96.130
NETMASK=255.255.255.0
GATEWAY=192.168.96.1
DNS1=8.8.8.8
```

图 3.4 配置静态 IP

重启网络服务通过“service network restart”命令。

```
[yql@localhost /]$ vi /etc/sysconfig/network-scripts/ifcfg-ens33
[yql@localhost /]$ sudo service network restart
[sudo] password for yql:
Restarting network (via systemctl):                        [  OK  ]
```

图 3.5 重启网络服务

<2> Debian/Ubuntu 系列设置静态 IP

查看网卡名，如下图所示，本机网卡名为 eth0

```
root@bogon:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.72.133  netmask 255.255.255.0  broadcast 192.168.72.255
        inet6 fe80::20c:29ff:fe18:54c2  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:18:54:c2  txqueuelen 1000  (Ethernet)
        RX packets 4830  bytes 331178 (323.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5113  bytes 599795 (585.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x2000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 16  bytes 708 (708.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 708 (708.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

图 3.6 查看网卡名

修改配置文件，如 Kali 的网卡配置文件则是/etc/network/interfaces，编辑配置文件，如下图所示，分别配置 IP、子网掩码、网关信息。

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback



auto eth0
iface eth0 inet static
address 192.168.72.176
netmask 255.255.255.0
gateway 192.168.72.2
~
```

图 3.7 编辑网卡配置文件

配置 DNS，kali 网卡配置文件是 /etc/resolv.conf,如下所示，追加 DNS 202.99.160.68.

```
File  Actions  Edit  View  Help
# Generated by NetworkManager
search localdomain
nameserver 192.168.72.2
nameserver 202.99.168.68
~
```

图 3.8 配置 DNS

配置完毕后需重启网络服务：/etc/init.d/networking restart

重启网络服务后可测试网络。

（2） 查看网络连接状态

Linux 使用过程中需经常监控网络连接状态，如开放端口、端口连接状态等。

如下图所示为通过“netstat”命令查看系统中开放的 TCP 和 UDP 端口，其中使用的具体参数含义请自行通过帮助命令查询，“netstat”不同的参数可定义不同的显示结果。

```
[yql@localhost /]$ sudo netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      1/systemd
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN      7330/X
tcp        0      0 192.168.122.1:53        0.0.0.0:*               LISTEN      7536/dnsmasq
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      7058/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      7061/cupsd
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      7370/master
tcp        0      0 127.0.0.1:6010          0.0.0.0:*               LISTEN      64716/sshd: yql@pts
tcp6       0      0 :::111                  :::*                    LISTEN      1/systemd
tcp6       0      0 :::6000                 :::*                    LISTEN      7330/X
tcp6       0      0 :::22                   :::*                    LISTEN      7058/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      7061/cupsd
tcp6       0      0 ::1:25                  :::*                    LISTEN      7370/master
tcp6       0      0 ::1:6010                :::*                    LISTEN      64716/sshd: yql@pts
udp        0      0 192.168.122.1:53        0.0.0.0:*                           7536/dnsmasq
udp        0      0 0.0.0.0:67              0.0.0.0:*                           7536/dnsmasq
udp        0      0 0.0.0.0:68              0.0.0.0:*                           123922/dhclient
udp        0      0 0.0.0.0:58452           0.0.0.0:*                           6582/avahi-daemon:
udp        0      0 0.0.0.0:111             0.0.0.0:*                           1/systemd
udp        0      0 0.0.0.0:5353            0.0.0.0:*                           6582/avahi-daemon:
udp        0      0 0.0.0.0:789             0.0.0.0:*                           6572/rpcbind
udp        0      0 127.0.0.1:323           0.0.0.0:*                           6588/chronyd
udp6       0      0 :::111                  :::*                                1/systemd
udp6       0      0 :::789                  :::*                                6572/rpcbind
udp6       0      0 ::1:323                 :::*                                6588/chronyd
```

图 3.9 查看系统开放端口

继续查询上图中端口所对应系统中的服务。Linux 操作系统端口与服务的对应关系定义与“/etc/services”文件中，如下图所示，为通过“grep”命令过滤“/etc/services”中的 111 端口所对应服务。

```
[yql@localhost /]$ sudo cat /etc/services | grep -w 111
[sudo] password for yql:
sunrpc          111/tcp         portmapper rpcbind      # RPC 4.0 portmapper TCP
sunrpc          111/udp         portmapper rpcbind      # RPC 4.0 portmapper UDP
```

图 3.10 查看端口所对应服务

2. 进程管理

（1） 通过“ps”命令查看进程

```
[yql@localhost /]$ ps
   PID TTY          TIME CMD
 64750 pts/1    00:00:00 bash
125104 pts/1    00:00:00 ps
```

图 3.11 查看进程

（2） 查看所有用户及所有进程信息

```
[yql@localhost /]$ ps -aux
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          1  0.0  0.5 201992  5264 ?        Ss   03:13   0:14 /usr/lib/systemd/sy
root          2  0.0  0.0      0     0 ?        S    03:13   0:00 [kthreadd]
root          3  0.0  0.0      0     0 ?        S    03:13   0:00 [ksoftirqd/0]
root          5  0.0  0.0      0     0 ?        S<   03:13   0:00 [kworker/0:0H]
root          7  0.0  0.0      0     0 ?        S    03:13   0:00 [migration/0]
root          8  0.0  0.0      0     0 ?        S    03:13   0:00 [rcu_bh]
root          9  0.0  0.0      0     0 ?        R    03:13   0:03 [rcu_sched]
root         10  0.0  0.0      0     0 ?        S<   03:13   0:00 [lru-add-drain]
root         11  0.0  0.0      0     0 ?        S    03:13   0:00 [watchdog/0]
root         13  0.0  0.0      0     0 ?        S    03:13   0:00 [kdevtmpfs]
root         14  0.0  0.0      0     0 ?        S<   03:13   0:00 [netns]
root         15  0.0  0.0      0     0 ?        S    03:13   0:00 [khungtaskd]
```

图 3.12 查看详细进程信息

字段说明：

USER:启动进程用户

PID：进程号

%CPU：CPU 占用率

%MEM：内存占用率

VSZ：占用虚拟内容大小（KB）

RSS：占用物理内存大小（KB）

TTY：进程建立时所对应的终端（？：不占用终端）

STAT：进程状态（S：睡眠；T：被跟踪或停止；Z：僵尸进程；W：无足够内存分页可分配；<：高优先级进程；N：低优先级进程；L：待续进程）

START：进程开始时间

TIME：进程执行时间

COMMAND：进程对应命令名

（3） 进程信息排序

实际使用中，常根据进程占用资源情况判断进程状态，“ps”命令可通过不同参数对进程进行排序。

如下图所示，为按内存占用情况对进程进行排序，请注意升序和降序使用参数的不同。

```
[yql@localhost /]$ ps -auxw --sort=rss
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          2  0.0  0.0      0     0 ?        S    03:13   0:00 [kthreadd]
root          3  0.0  0.0      0     0 ?        S    03:13   0:00 [ksoftirqd/0]
root          5  0.0  0.0      0     0 ?        S<   03:13   0:00 [kworker/0:0H]
root          7  0.0  0.0      0     0 ?        S    03:13   0:00 [migration/0]
root          8  0.0  0.0      0     0 ?        S    03:13   0:00 [rcu_bh]
root          9  0.0  0.0      0     0 ?        R    03:13   0:03 [rcu_sched]
root         10  0.0  0.0      0     0 ?        S<   03:13   0:00 [lru-add-drain]
```

图 3.13 按内存使用排序（升序）

```
[yql@localhost /]$ ps -auxw --sort=-rss
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
yql        8204  0.0 16.1 3055348 160516 ?      Sl   03:14   0:21 /usr/bin/gnome-shell
yql        9990  0.0  3.8 102222228 38272 ?     Sl   03:42   0:01 gnome-control-center
yql        8503  0.0  3.3 1106292 33276 ?       Sl   03:15   0:02 /usr/bin/gnome-softwa
root       7330  0.0  3.0 318844 30868 tty1     Ssl+ 03:13   0:03 /usr/bin/X :0 -backgr
yql        8485  0.0  1.5 1042556 15848 ?       Sl   03:15   0:00 nautilus-desktop --fo
yql        8840  0.0  1.3 670384 13768 ?        Sl   03:15   0:01 /usr/libexec/gnome-te
```

图 3.14 按内存使用排序（降序）

如下图所示，为按 CPU 使用情况对进程进行排序，请注意升序和降序使用参数的不同。

```
[yql@localhost /]$ ps -auxw --sort=%cpu
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          1  0.0  0.5 201992  5264 ?        Ss   03:13   0:14 /usr/lib/systemd
root          2  0.0  0.0      0     0 ?        S    03:13   0:00 [kthreadd]
root          3  0.0  0.0      0     0 ?        S    03:13   0:00 [ksoftirqd/0]
root          5  0.0  0.0      0     0 ?        S<   03:13   0:00 [kworker/0:0H]
root          7  0.0  0.0      0     0 ?        S    03:13   0:00 [migration/0]
root          8  0.0  0.0      0     0 ?        S    03:13   0:00 [rcu_bh]
root          9  0.0  0.0      0     0 ?        R    03:13   0:04 [rcu_sched]
```

图 3.15 按 CPU 使用排序（升序）

```
[yql@localhost /]$ ps -auxw --sort=-%cpu
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          1  0.0  0.5 201992  5264 ?        Ss   03:13   0:14 /usr/lib/systemd
root          2  0.0  0.0      0     0 ?        S    03:13   0:00 [kthreadd]
root          3  0.0  0.0      0     0 ?        S    03:13   0:00 [ksoftirqd/0]
root          5  0.0  0.0      0     0 ?        S<   03:13   0:00 [kworker/0:0H]
root          7  0.0  0.0      0     0 ?        S    03:13   0:00 [migration/0]
root          8  0.0  0.0      0     0 ?        S    03:13   0:00 [rcu_bh]
root          9  0.0  0.0      0     0 ?        R    03:13   0:04 [rcu_sched]
```

图 3.16 按 CPU 使用排序（降序）

（4） 动态查看进程信息

```
[yql@localhost /]$ top
top - 21:50:37 up 18:37,  3 users,  load average: 0.00, 0.01, 0.05
Tasks: 216 total,   1 running, 215 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.3 sy,  0.0 ni, 99.7 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :   995896 total,    64408 free,   655320 used,   276168 buff/cache
KiB Swap:  2097148 total,  1927932 free,   169216 used.    82936 avail Mem

   PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
  6562 root      20   0  320024   2104   1744 S  0.3  0.2   0:56.10 vmtoolsd
  8511 yql       20   0  566600   7580   1872 S  0.3  0.8   1:01.61 vmtoolsd
125529 root      20   0       0      0      0 S  0.3  0.0   0:00.07 kworker/0:0
     1 root      20   0  201992   5264   2900 S  0.0  0.5   0:14.12 systemd
     2 root      20   0       0      0      0 S  0.0  0.0   0:00.04 kthreadd
     3 root      20   0       0      0      0 S  0.0  0.0   0:00.68 ksoftirqd/0
```

图 3.17 查看系统进程信息

（5） 终止进程

如下图所示，通过上述内容可查询，本实例中 PID 为 8973 的进程为切换 root 用户产生，可用于本示例中终止进程演示。

```
[yql@localhost /]$ ps -aux | grep 8973
root      8973  0.0  0.2 232132  2388 pts/0    S    03:15   0:00 su root
yql     127272  0.0  0.0 112708   976 pts/1    R+   22:15   0:00 grep --color=auto 8973
```

图 3.18 确认进程 PID

终止进程并确认。

```
[yql@localhost /]$ sudo kill 8973
[yql@localhost /]$ ps -aux | grep 8973
yql      127424  0.0  0.0 112708   976 pts/1    R+   22:17   0:00 grep --color=auto 8973
```

图 3.19 终止进程并确认

（6） 强制终止进程

强制终止进程为“kill -9 [pid]”,在此不做演示。

3. 磁盘管理

（1） 查看已挂在磁盘总容量、已使用、剩余容量

如图下图所示，为磁盘使用详细信息。其中“Filesystem”表示扇区。

```
[yql@localhost /]$ df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   17G  4.0G   14G  24% /
devtmpfs                 470M     0  470M   0% /dev
tmpfs                    487M     0  487M   0% /dev/shm
tmpfs                    487M   15M  472M   4% /run
tmpfs                    487M     0  487M   0% /sys/fs/cgroup
/dev/sda1               1014M  166M  849M  17% /boot
tmpfs                     98M  4.0K   98M   1% /run/user/42
tmpfs                     98M   36K   98M   1% /run/user/1000
```

图 3.20 查看磁盘使用情况

（2） 查看目录或文件所占空间

如下图所示为通过“du”命令查看目录所占用空间，具体参数意义可自行查询。

```
[yql@localhost /]$ du -s /home/yql/
77532   /home/yql/
[yql@localhost /]$ du -sm /home/yql/
76      /home/yql/
```

图 3.21 查看目录所占空间

实验内容

1、 完成上述实验演示，并记录结果。