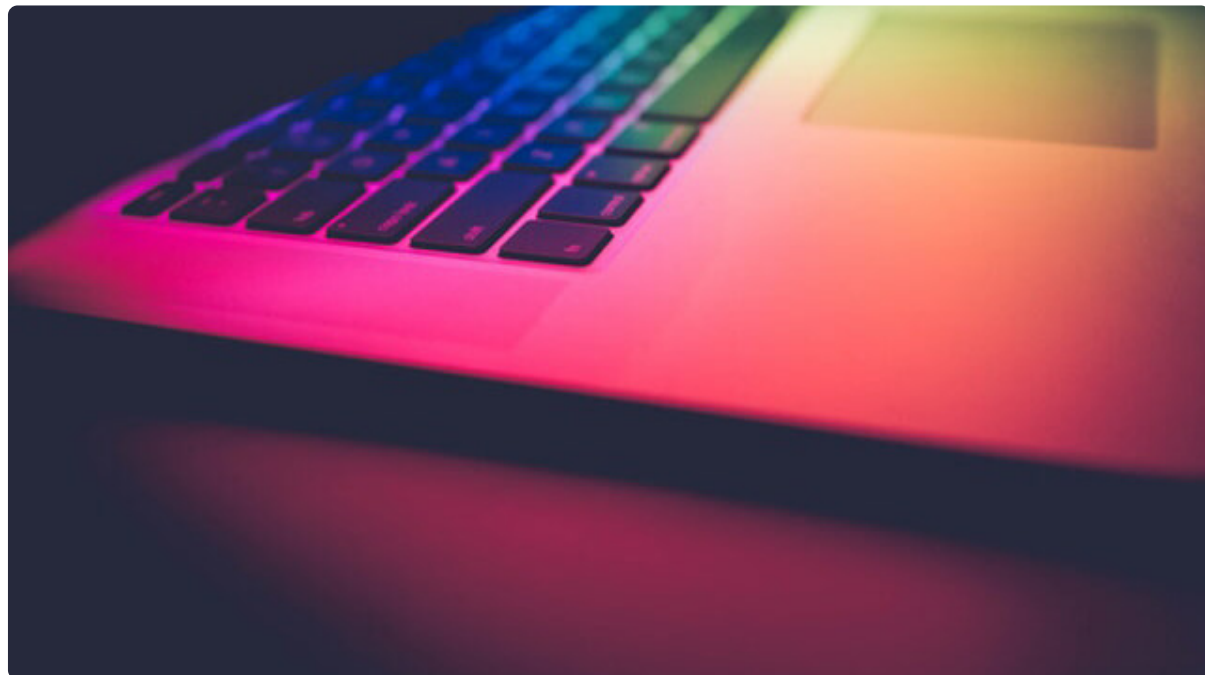


35 IP地址和分析网络

更新时间：2019-07-31 09:33:44



“

学习从来无捷径，循序渐进登高峰。

—— 高永祚

”

内容简介

1. 前言
2. host 和 whois 命令：告诉我你是谁
3. ifconfig 和 netstat 命令：控制和分析网络流量
4. 总结
5. 第四部分第六课预告

1. 前言

上一课 [带你玩转Linux和Shell编程 | 第四部分第四课：文件传输，潇洒同步](#) 比较轻松，这一课有点难度。

俗语说得好：“爱国爱家爱师妹，防火防盗防师兄”

虽然说，Linux 本身挺安全。你看 Linux 发行版，基本不需要安装杀毒软件，基于修改的 Unix 系统的苹果 macOS 系统也基本不需要装杀毒软件。

相反，如果 Windows 不装个杀毒软件，简直不好意思出门跟别人打招呼（日常黑 Windows 任务完成 :P）。开个玩笑啦，这也是因为 Windows 使用者多，所以针对 Windows 的病毒也多。

不过，在网络上穿行，还是小心为妙。毕竟网上有不怀好意的黑客（或普通人），想方设法获取我们的隐私，或者控制我们的电脑。

这一课和下一课很重要，我们来学习几个厉害的命令，能帮助我们分析网络，再来学习如何使用轻量级的防火墙。

我们将一起学习经由电脑的网络通信是怎么回事。

事实上，当你的电脑连接到互联网（Internet）时，电脑里总有那么一些软件会连上网，然后下载或上传信息，很多时候还是偷偷地在后台进行。

如何监控此等“大逆不道”的行径呢？如何知道哪个软件正在与网络通信，在哪个端口上进行呢？就是本课要探究的。

还有，懂得配置防火墙是很必要的，不管是在你的家用电脑上还是在你租用的服务器上。因为这能有效保护电脑，免得一些软件未经你同意就和互联网交换信息。

使用防火墙是一个重要的安全防护措施，特别是每个系统管理员（System Administrator），更是不能逃避，总要学习。

好了，废话了这么多，开篇入正题吧。

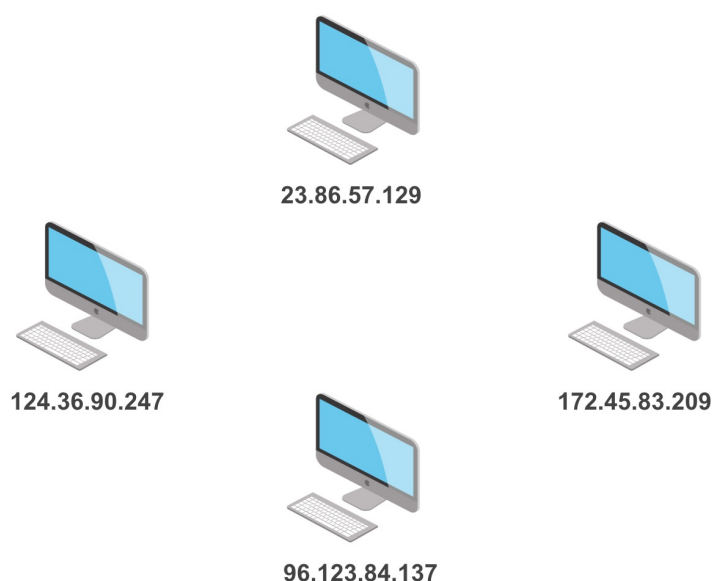
2. host 和 whois 命令：告诉我你是谁

我们知道，每台连上网的电脑都会被一个 IP 地址所标识，这个 IP 地址是全球唯一的。

目前，大多数 IP 地址是这样的形式：

```
43.136.208.71
```

这样的形式被称为 IPv4 格式（以小数点分隔），分隔开的部分的取值范围是 0 ~ 255（因为是 8 位二进制数， $2^8 - 1 = 256 - 1 = 255$ ）。IP 是 Internet Protocol（“网际协议”或“互联网协议”）的缩写，v 是 version（“版本”）的首字母，4 表示这是“第四版”。



不过，因为 IPv4 的地址不够用的关系，现在已经有不少 IP 地址的格式是 IPv6 的格式了，已有的 IPv4 地址也慢慢被替换成 IPv6。IPv6 的 IP 地址类似这样：

```
ef70::103:72cd:cb50:17d2
```

上面是简写形式（两个冒号 :: 用于合并连续的几组 0），完整的形式是：

```
ef70:0000:0000:0000:0103:72cd:cb50:17d2
```

我们看到，IPv6 的 IP 地址是以冒号分隔的 8 组 4 位的 16 进制（也就是 8 组 16 位二进制，因为 2 的 4 次方是 16），所以 $8 * 16 = 128$ 。

IPv6 把 IP 地址由 IPv4 的 32 位（4 个 2 的 8 次方， $4 * 8 = 32$ ）增加到 128 位。

2 的 32 次方是 4294967296（近 43 亿）。随着互联网的快速成长，IPv4 的近 43 亿个地址的分配最终于 2011 年 2 月 3 日用尽。

而 IPv6 的 128 位可以提供的 IP 地址数量可达 $2^{128} = 3.402823669 \times 10^{38}$ (10 的 38 次方) 个。届时每个人家中的每件电器，每样东西，甚至地球上每一粒沙子都可以拥有好几个 IP 地址。

根本用不完，IP 多就是这么任性~

我们可以为每一个 IP 地址绑定一个主机名，主机名的英语是 host name。

注意：

这里的主机名其实应该被称为“完整主机名”：Fully Qualified Domain Name (FQDN)。FQDN 是由主机名（host name）和域名（domain name）一起构成。我们一般会用主机名或域名来代称 FQDN，其实不是那么准确。

Host name 和 domain name 是不同的概念，稍微有点复杂。

例如，Github 的服务器的完整主机名是 www.github.com，这里的 [www](http://www.github.com) 是主机名，[github.com](http://www.github.com) 是域名，对应的 IP 地址是 140.82.118.4。

下面的课程中，我们会用主机名或域名代指 FQDN。

相比于记忆一串数字（IPv4）或数字+字母（IPv6）的 IP 地址，人脑更擅长记住主机名。如下图，IP 地址和主机名对应：



IP 地址和主机名的互相转换

使用 `host` (`host` 是英语“主机”的意思) 这个命令我们可以实现 IP 地址和主机名的互相转换, 例如:

- `host github.com` (或 `host www.github.com`) 会得到 IP 地址: 140.82.118.4 (不一定每次获得的 IP 地址都一样, 我有时获得的是 140.82.118.3);
- 而 `host 140.82.118.4` 则会得到主机名的信息。

`host` 命令的其它参数选项, 可以用 `man host` 来查考。

自定义解析

IP 地址和主机名的解析是由 DNS (Domain Name System, “域名解析系统”) 服务器完成的。我们不深入了解 DNS 的具体工作原理了, 因为一时半会解释不完。

因此, 当你在浏览器的地址栏里输入 `github.com` 时, 你就能收到由网络上的 DNS 服务器帮你解析出来的 IP 地址, 你就可以访问 Github 的官网了。

你当然不能修改 DNS 服务器上的主机名和 IP 地址的对应关系列表 (因为有风险, 会影响到很多人)。但是你可以自定义你电脑上主机名和 IP 地址的对应关系。

以 `root` 身份打开 `/etc/hosts` 文件:

```
sudo nano /etc/hosts
```

```
oscar@oscar-laptop: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 /etc/hosts  
  
127.0.0.1    localhost  
127.0.1.1    oscar-laptop  
  
# The following lines are desirable for IPv6 capable hosts  
::1         ip6-localhost ip6-loopback  
fe00::0     ip6-localnet  
ff00::0     ip6-mcastprefix  
ff02::1     ip6-allnodes  
ff02::2     ip6-allrouters
```

可以看到一些 IP 地址和主机名的对应关系，例如：

- 127.0.0.1 对应了 localhost；
- 127.0.1.1 对应了 oscar-laptop，就是我的电脑的主机名。

下面还有一些 IPv6 格式的 IP 地址。

你可以往里面添加自己的解析对。写在一行里，在 IP 地址和主机名之间至少要留一个空格。

那么自定义 IP 地址和主机名之间的解析关系，有什么用呢？

当然有用。有的时候，网络上负责解析域名的 DNS 服务器可能暂时出现故障（很少见），那这时你在地址栏直接输入 IP 地址可以访问对应网站，但是输入主机名（域名）则不能访问了，因为没有 DNS 服务器解析，域名是不能被识别的。

这时，你自定义在本地 Linux 系统的 /etc/hosts 文件中的解析规则就可以发挥作用了，你还是可以访问网站。

例如你可以加一条 Github 的解析对：

```
140.82.118.4 github.com
```

但是这样的方法也有缺陷：假如网站的主机的 IP 地址换了，那你输入域名就不能上那个网站了。所以，一般来说，我们还是靠经常更新解析对的网络上的 DNS 服务器更好（一般一天更新一次）。

对于局域网，这个 /etc/hosts 文件也是很有用的。

假设在自己家建立的局域网里面，有你的电脑，还有你老爸的电脑。那你不想知道老爸的电脑的 IP 地址，怎么办呢？可以添加一条解析对，例如：

```
192.168.0.5 father-laptop
```

这样，老爸的电脑的 IP 地址对应的主机名在你这里就成为 father-laptop 了，你就可以更方便地访问老爸的电脑。

例如，如果在老爸的电脑里你有一个账户叫 oscar，那么用之前学的 SSH 来登录老爸电脑上的 oscar 账户，就是：

```
ssh oscar@father-laptop
```

就不用输入以下命令了：

```
ssh oscar@192.168.0.5
```

whois: 了解有关域名的信息

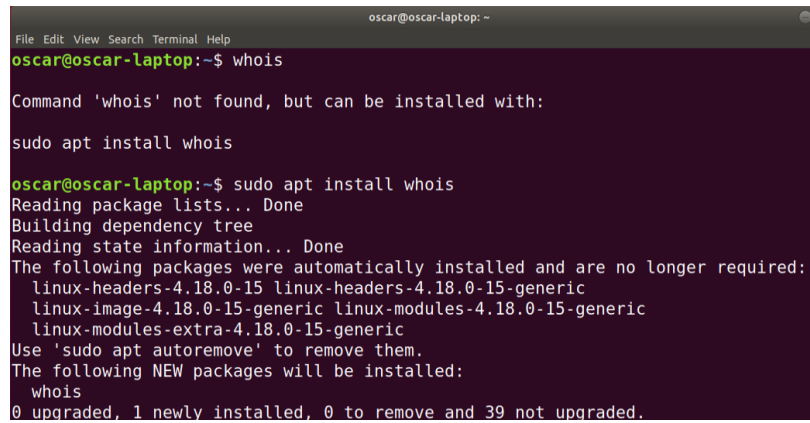
whois 这个命令很好记，由 who 和 is 两部分组成，who 是英语“谁”的意思，is 是英语“是”的第三人称单数形式。所以连起来表示：“是谁”。

每一个域名在登记注册时都须要填写：姓、名、联系地址、联系方式等等，这是规定。

whois 命令就可以帮助我们轻松获得域名背后的这些信息。

如果你的 Ubuntu 系统里没有安装 whois 命令，可以用以下命令来安装：

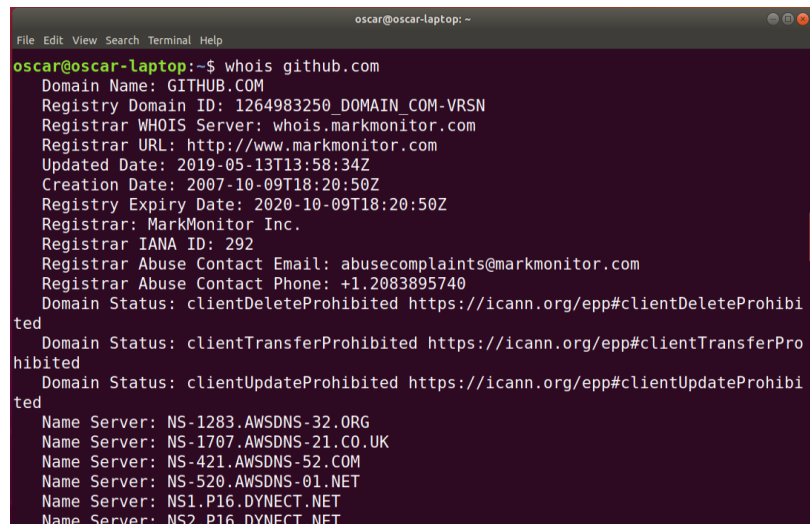
```
sudo apt install whois
```



```
oscar@oscar-laptop: ~  
File Edit View Search Terminal Help  
oscar@oscar-laptop:~$ whois  
Command 'whois' not found, but can be installed with:  
  
sudo apt install whois  
  
oscar@oscar-laptop:~$ sudo apt install whois  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  linux-headers-4.18.0-15 linux-headers-4.18.0-15-generic  
  linux-image-4.18.0-15-generic linux-modules-4.18.0-15-generic  
  linux-modules-extra-4.18.0-15-generic  
Use 'sudo apt autoremove' to remove them.  
The following NEW packages will be installed:  
  whois  
0 upgraded, 1 newly installed, 0 to remove and 39 not upgraded.
```

可以选一个主机名来试试，例如：

```
whois github.com
```



```
oscar@oscar-laptop: ~  
File Edit View Search Terminal Help  
oscar@oscar-laptop:~$ whois github.com  
Domain Name: GITHUB.COM  
Registry Domain ID: 1264983250_DOMAIN_COM-VRSN  
Registrar WHOIS Server: whois.markmonitor.com  
Registrar URL: http://www.markmonitor.com  
Updated Date: 2019-05-13T13:58:34Z  
Creation Date: 2007-10-09T18:20:50Z  
Registry Expiry Date: 2020-10-09T18:20:50Z  
Registrar: MarkMonitor Inc.  
Registrar IANA ID: 292  
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com  
Registrar Abuse Contact Phone: +1.2083895740  
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited  
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited  
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited  
Name Server: NS-1283.AWSDNS-32.ORG  
Name Server: NS-1707.AWSDNS-21.CO.UK  
Name Server: NS-421.AWSDNS-52.COM  
Name Server: NS-520.AWSDNS-01.NET  
Name Server: NS1.P16.DYNECT.NET  
Name Server: NS2.P16.DYNECT.NET
```

whois 会输出很多信息，一个屏幕还显示不完。

3. ifconfig 和 netstat 命令：控制和分析网络流量

我们接着学习两个非常强大的命令：ifconfig 和 netstat。

ifconfig: 列出网络接口

ifconfig 是 Linux 中用于显示或配置网络设备（网络接口卡）的命令，英文全称是 Network Interface Configuration（意为“网络接口配置”）。network 表示“网络”；if 是 Interface 的缩写，表示“接口”；config 是 configuration 的缩写，表示“配置”。

一般来说，你的电脑拥有好几个网络接口，也就是说有多种连接网络的方式。

如果你的 Ubuntu 系统里没有 ifconfig 这个命令，那么可以运行以下命令来安装：

```
sudo apt install net-tools
```

我们运行 `ifconfig` 命令。在旧版的 Ubuntu 系统（16.04 以前）里，会显示类似如下的信息：


```
ooscar@ooscar-laptop: ~  
File Edit View Terminal Help  
ooscar@ooscar-laptop:~$ ifconfig  
eth0      Link encap:Ethernet  HWaddr f0:de:f1:7b:09:00  
          inet addr:172.16.38.249  Bcast:172.16.38.255  Mask:255.255.255.0  
          inet6 addr: fe80::f2de:f1ff:fe7b:900/64 Scope:Link  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:4853 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:4821 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:3474961 (3.4 MB)  TX bytes:618173 (618.1 KB)  
          Interrupt:30 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:148 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:148 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:0  
          RX bytes:12088 (12.0 KB)  TX bytes:12088 (12.0 KB)  
  
wlan0     Link encap:Ethernet  HWaddr 38:59:f9:dc:45:5b  
          inet6 addr: fe80::3a59:f9ff:fedc:455b/64 Scope:Link  
          UP BROADCAST MULTICAST  MTU:1500  Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)  
          Interrupt:19 Memory:f8360000-f8360100
```

可以看到，显示有三个网络接口，分别是：

- eth0
- lo
- wlan0

在你的电脑上可能网络接口多于三个，也可能少于三个，本不足为奇，取决于你电脑的设备。幸好我的电脑的三个网络接口都是最常见的三个，详述如下：

- eth0：对应有线连接（对应你的有线网卡），就是用网线来连接的上网（一般是 RJ45 网线，就是平时我们用的那种网线。见下图），如果你的电脑目前使用网线来上网，那就是在使用这个接口。eth 是 Ethernet 的缩写，表示“以太网”。有些电脑可能同时有好几条网线连着（有好几个有线接口），例如服务器，那么除了 eth0（第一块有线网卡），你还会看到例如 eth1、eth2等等。



- lo：本地回环（Local Loopback 的缩写，对应一个虚拟网卡）。可以看到它的 IP 地址是 127.0.0.1，似曾相识是吗？对啊，之前我们在 /etc/hosts 文件中看到对应 localhost 的就是这个 IP。每台电脑都应该有这个接口，因为

它对应着“连向自己的链接”。这也是被称之为“本地回环”的原因：所有经由这个接口发送的东西都会回到你自己的电脑。看起来好像并没有什么用，但有时为了某些缘故（可以百度或 Google 继续探究），我们需要连接自己。例如用来测试一个网络程序，但又不想让局域网或外网的用户能够查看，只能在此台主机上运行和查看所用的网络接口。比如把 HTTPD 服务器指定到回环地址，在浏览器输入 127.0.0.1 就能看到你所架设的 Web 网站了。但只有你自己能看得到，局域网的其它主机或用户无从知道。

- wlan0: 对应 Wi-Fi 无线连接（对应你的无线网卡）。wlan 是 Wireless Local Area Network 的缩写，表示“无线局域网”。假如你有好几块无线网卡，那么会看到 wlan1、wlan2 等等。

仔细观察上面 ifconfig 输出的信息，你猜到我此时用的是哪种方式连网的吗？

聪明如你应该猜到了（可不要害怕阅读一堆英语哦）。是的，我此时用的是有线连接的方式上网。证明就在 eth0 的信息中：

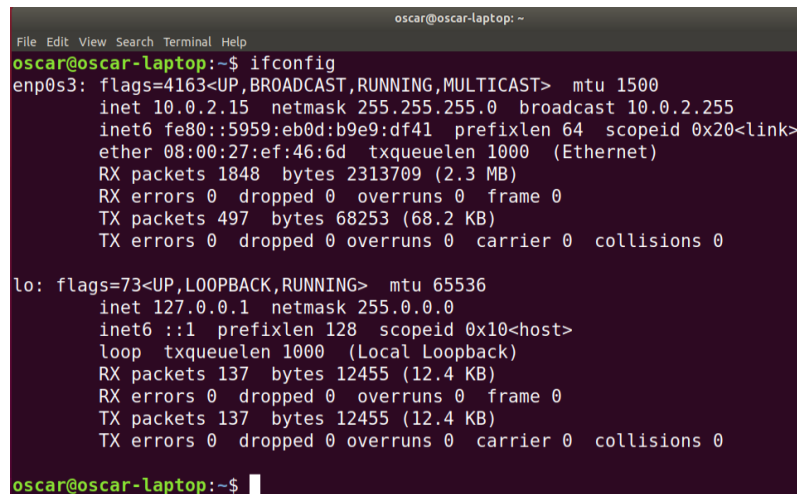
```
RX packets : 4853
TX packets : 4821
```

分别是接收包和发送包的数目。RX 是 receive（表示“接收”）的缩写，TX 是 transmit（表示“发送”）的缩写。packet 是英语“包，数据包”的意思。

而 wlan0 的信息中，这两行是 0。表示没有发送和接收任何数据。

当然，本地回环的这两行也有 148 呢。所以说，给自己发送东西也是会发生的。

如果你是在新版的 Ubuntu 系统里（从 16.04 开始），运行 ifconfig，应该会显示类似以下的信息：



```
File Edit View Search Terminal Help
oscar@oscar-laptop: ~
oscar@oscar-laptop:~$ ifconfig
enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::5959:eb0d:b9e9:df41 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:ef:46:6d txqueuelen 1000 (Ethernet)
    RX packets 1848 bytes 2313709 (2.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 497 bytes 68253 (68.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 137 bytes 12455 (12.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 137 bytes 12455 (12.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

oscar@oscar-laptop:~$
```

可以看到，以太网卡的名字不是我们以前熟悉的 eth0，而变成了不太好理解的 enp3s0。

这是因为新版用 systemd 替换掉了 initd 来引导系统。其带来的另外一个副作用（或者可以说是优点）就是网络接口的命名方式变了：


```

/*
 * Two character prefixes based on the type of interface:
 *   en – Ethernet
 *   sl – serial line IP (slip)
 *   wl – wlan
 *   ww – wwan
 *
 * Type of names:
 *   b<number>                – BCMA bus core number
 *   c<bus_id>                – CCW bus group name, without leading zeros [s390]
 *   o<index>[d<dev_port>]    – on-board device index number
 *   s<slot>[f<function>][d<dev_port>] – hotplug slot index number
 *   x<MAC>                  – MAC address
 *   [P<domain>]p<bus>s<slot>[f<function>][d<dev_port>]
 *                               – PCI geographical location
 *   [P<domain>]p<bus>s<slot>[f<function>][u<port>][.][c<config>][i<interface>]
 *                               – USB port number chain
 */

```

上面一段是 systemd 源码的注释。所以 enp3s0 的意思是：

- en 代表以太网卡；
- p3s0 代表 PCI 接口的物理位置为 (3, 0)，其中横坐标代表 bus（“总线”），纵坐标代表 slot（“槽，插口”）。

详细信息可以参看 [FreeDesktop 的 Wiki](#)。

当然你也可以配置，把 enp3s0 重新改为以前的 eth0。可以参看这篇文章：[ifconfig不是熟悉的eth0，却是enp0s3](#)。

ifconfig 命令还可以用来配置网络接口。但这有些超出本课程的范围了，毕竟需要懂一些网络方面的知识才能配置啊，不然配错了就糟了。

但是，有一个简单的配置方式我们可以学习：接口的激活/关闭。

使用格式如下：

```
ifconfig interface state
```

替换：

- interface: 由你想要修改的网络接口名称代替（eth0 / enp3s0, wlan0 等等）。interface 是英语“接口”的意思；
- state: 由 up 或 down 代替，分别表示“激活”和“关闭”对应接口。up 是英语“向上”的意思，down 是英语“向下”的意思。state 是英语“状态”的意思。设置“激活”和“关闭”需要 root 权限，所以加上 sudo。

例如：

```

# 关闭 eth0 这个有线接口，之后就没有任何网络传输会在 eth0 上进行了。
sudo ifconfig eth0 down

```

```

# 激活 eth0 这个有线接口。
sudo ifconfig eth0 up

```

对于新版，激活和关闭以太网接口的对应命令就是：

```

# 关闭 enp3s0 这个有线接口，之后就没有任何网络传输会在 enp3s0 上进行了。
sudo ifconfig enp3s0 down

```

```
# 激活 enp3s0 这个有线接口。
sudo ifconfig enp3s0 up
```

```
oscar@oscar-laptop: ~
File Edit View Search Terminal Help
TX packets 277 bytes 66732 (66.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 133 bytes 11190 (11.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 133 bytes 11190 (11.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

oscar@oscar-laptop:~$ sudo ifconfig enp0s3 down
oscar@oscar-laptop:~$ ifconfig
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 141 bytes 11790 (11.7 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 141 bytes 11790 (11.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

oscar@oscar-laptop:~$
```

可以看到，我用 `sudo ifconfig enp3s0 down` 来关闭 `enp3s0` 接口之后，再运行 `ifconfig` 就只显示 `lo` 这个接口了。

我们可以用 `sudo ifconfig enp3s0 up` 来重新激活 `enp3s0` 接口：

```
oscar@oscar-laptop:~$ sudo ifconfig enp0s3 up
oscar@oscar-laptop:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
inet6 fe80::5959:eb0d:b9e9:df41 prefixlen 64 scopeid 0x20<link>
ether 08:00:27:ef:46:6d txqueuelen 1000 (Ethernet)
RX packets 423 bytes 370167 (370.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 310 bytes 70753 (70.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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 178 bytes 14479 (14.4 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 178 bytes 14479 (14.4 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

oscar@oscar-laptop:~$
```

配置 IP 地址

用 `ifconfig` 命令可以配置网络接口的 IP 地址和其它数据，例如：

```
ifconfig eth0 192.168.120.56 netmask 255.255.255.0 broadcast 192.168.120.255
```

上面的命令用于给 `eth0` 网卡配置 IP 地址（192.168.120.56），加上子网掩码（255.255.255.0），加上广播地址（192.168.120.255）。

对于新版，可以用：

```
ifconfig enp3s0 192.168.120.56 netmask 255.255.255.0 broadcast 192.168.120.255
```

还有更多命令选项就用 `man ifconfig` 来查看吧。

篇幅关系，暂时我们就学到这里，剩下的内容我们下一课继续学习。

4. 总结

1. 在 Internet（互联网）上，每台电脑都由 IP 地址来标识。例如：72.208.169.35;
2. 我们可以把一个主机名绑定到一个 IP 地址上，主机名相对 IP 地址更容易被记住。例如：github.com。在很多场合，用主机名就可以代替 IP 地址了；
3. host 命令可以从 IP 地址解析出对应主机名，或者从主机名解析出 IP 地址；
4. ifconfig 命令列出你电脑上的网络接口（对应虚拟网卡或实体网卡），可以配置和操作这些接口。

今天的课就到这里，一起加油吧！

← 34 文件传输，潇洒同步

36 统计网络，安全防火墙 →

精选留言 0

欢迎在这里发表留言，作者筛选后可公开显示



目前暂无任何讨论