

如何查看系统的 ip 地址

- windows: ipconfig
- Linus: ifconfig 或者 ip addr (如果没有这两个命令, 可以自行安装 net-tools 和 iproute2)

运行 ip addr

```
root@test:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether fa:16:3e:c7:79:75 brd ff:ff:ff:ff:ff:ff
    inet 10.100.122.2/24 brd 10.100.122.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fec7:7975/64 scope link
        valid_lft forever preferred_lft forever
```

IP 地址是一个网卡在网络世界的通讯地址, 相当于我们现实世界的门牌号码。

既然是类似门牌号码, 那么就不能冲突, 如果大家都叫 102 号, 那就找不到地方了。我们有时候遇到的网络地址冲突, 上不去网, 多半是 IP 地址冲突了。

我们来看一个 ip 地址

10.100.122.2: 这个地址被 . 分隔成了四个部分, 每个部分 8 个 bit, 所以 IP 地址总共是 32 位, 但是这样设计的地址很快就不够用了, 所以又产生了 IPV6 地址。

2001:0000:3238:00E1:0063:0000:0000:FEFB: IPV6 地址由 8 个 16 位的块组成, 总共 128 位, 目前看来是够用了, 但是将来也不一定。更多 IPV6 介绍可以看文章 [IPv6笔记-地址结构与分类](#)

32 位的 IPV4 地址被分成了 5 类

| | | | |
|----|-----------|------------|-----------|
| A类 | 0 | 网络号 (7位) | 主机号 (24位) |
| B类 | 1 0 | 网络号 (14位) | 主机号 (16位) |
| C类 | 1 1 0 | 网络号 (21位) | 主机号 (8位) |
| D类 | 1 1 1 0 | 多播组号 (28位) | |
| E类 | 1 1 1 1 0 | 留待后用 (27位) | |

当时设计的时候 A B C 类主要分两部分，前面一部分是**网络号**，后面一部分是**主机号**，可以理解为大家都是六单元 1001 号，我是小区 A 的六单元 1001，而你是小区 B 的六单元 1001。

下面表格表示了 A B C 类地址所能包含的主机的数量

| 类别 | IP地址范围 | 最大主机数 | 私有IP地址范围 |
|----|---------------------------|----------|-----------------------------|
| A | 0.0.0.0-127.255.255.255 | 16777214 | 10.0.0.0-10.255.255.255 |
| B | 128.0.0.0-191.255.255.255 | 65534 | 172.16.0.0-172.31.255.255 |
| C | 192.0.0.0-223.255.255.255 | 254 | 192.168.0.0-192.168.255.255 |

怎么计算的？

注意：全 0 和 全 1 的地址是保留地址，“0” 作为网络号，“1” 作为广播号，所以主机数需要减去 2

由于 C 类地址包含的最大主机数太少了，只有 254 个，而 B 类地址能包含的最大主机数又太多了，所以有了一个折中的方案，**无类型域间选路(CIDR)**，

无类型域间选路(CIDR)

这种做法打破了原来设计的几类地址的做法，将 32 位的 IP 地址一分为二，前面是**网络号**，后面是**主机号**。观察这个 IP `10.100.122.2/24` 这个 IP 地址有一个斜杠，斜杠前面有个数字是 24，这种地址表示形式就是 CIDR，24 的意思是，在 32 位地址中，前 24 位是网络号，后 8 位是主机号。

伴随着 CIDR 存在的，一个是**广播地址**，`10.100.122.255` (**网络号不变，主机号全 1**)。如果发送这个地址，所有 `10.100.122` 网络里面的机器都可以收到。另一个是**子网掩码** `255.255.255.0` (**网络号全 1，主机号全 0**)

总结：广播地址：网络号不变，主机号全 1；子网掩码：网络号全 1，主机号全 0

将子网掩码和 IP 地址进行 AND 计算，由于前三个都是 255，也就是全 1，1 和任何数值 AND 都是 1，所以前三个数不变，为 10.100.122，最后一个数是 0，它和任何数 AND 都是 0，这就是网络号。

将子网掩码和 IP 地址进行按位与运算就得到了网络号。

公有 IP 和私有 IP 地址

日常工作中，几乎不用划分 A 类，B 类或 C 类，但是有一点需要注意，就是公有 IP 地址和私有 IP 地址。

| 类别 | IP地址范围 | 最大主机数 | 私有IP地址范围 |
|----|---------------------------|----------|-----------------------------|
| A | 0.0.0.0-127.255.255.255 | 16777214 | 10.0.0.0-10.255.255.255 |
| B | 128.0.0.0-191.255.255.255 | 65534 | 172.16.0.0-172.31.255.255 |
| C | 192.0.0.0-223.255.255.255 | 254 | 192.168.0.0-192.168.255.255 |

上面表格最右侧就是私有 IP 地址，平时在家里学校或者办公室的 IP 地址，一般都是私有 IP 地址段。这些地址允许组织内部的 IT 人员进行自己管理，自己分配，而且可以重复。就像每个小区都有楼编号和门牌号，你们小区可以叫 6 栋，我们小区也叫 6 栋，这没有任何问题，但是出了小区，就要使用公有 IP 地址，比如**码农路 666 号**，是国家统一分配的。

公有 IP 地址有个组织统一分配，你需要去买，如果你搭建一个网站给学校里人用，让学校 IT 给你分配一个 IP 地址就行。但是如果你要做一个网站让全世界的人都可以访问，就需要公有 IP

表中的 192.168.0.x 是最常用的私有 IP 地址，一般家里的上网设备不会超过 256 个，所以 /24 基本就够了，有时候也会见到 /16，这两种是最常见的。

对于 /24 来说，整个网络中的第一个地址 192.168.0.1 就是你这个私有网络的**出口地址**，例如家里连接 WIFI，WIFI 地址就是 192.168.0.1，而 192.168.0.255 就是广播地址。一旦发送这个地址，整个 192.168.0 网络里的机器都能收到。**但是也不总是这种情况，其他情况可能难以理解，还会出错。**(理解了下面的算法才能真正理解 CIDR 是如何划分的)

例：一个容易“犯错”的 CIDR

看一下 16.158.165.91/22 这个 CIDR，求以下这个网络的第一个地址，子网掩码和广播地址。

/22 不是 8 的整数倍，不好算，需要先转成 2 进制，16.158 前 16 位不用算，165 转成二进制是 10100101,这 8 位中，前 6 位是网络号，16.158.<101001>,而 <01>.91 是机器号。

所以第一个地址是 16.158.<101001><00>.1，即 16.158.164.1，子网掩码是 255.255.<111111><00>.0，即 255.255.252.0，广播地址为 16.158.<101001><11>.255,即 16.158.167.255

D 类地址表示**组播地址**，使用这一类地址，属于某个组的机器都可以收到。类似于在公司加了一个小组讨论群，发送消息，小组的人都可以收到。

网卡的 scope

再看看 ip addr 的输出内容

```
root@test:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether fa:16:3e:c7:79:75 brd ff:ff:ff:ff:ff:ff
    inet 10.100.122.2/24 brd 10.100.122.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fec7:7975/64 scope link
        valid_lft forever preferred_lft forever
```

在 IP 地址的后面有个 scope，对于 eth0 这张网卡来讲，是 global，说明这张网卡是可以对外的，可以接收来自各个地方的包，对于 lo 来说是 host，说明这张网卡仅仅可以供本机相互通信。

lo 全称 `loopback` (环回接口)，往往会被分配 127.0.0.1 这个地址，这个地址用于本机通信，经过内核处理后直接返回，不会出现在网络中。

MAC 地址

在 IP 地址的上一行是 `link/ether fa:16:3e:c7:79:75 brd ff:ff:ff:ff:ff:ff` 这个被称为 MAC 地址，是一个网卡的物理地址，用十六进制，6 个 byte 表示。

MAC 地址号称全局唯一，不会有两个网卡有相同的 MAC 地址（实际只要子网内唯一就可以），而且自网卡生产出来就带着这个地址。那么，整个互联网的通信可以直接用 MAC 地址吗，只要知道对方的 MAC 地址，就可以把信息传递过去？

当然是不行的，一个网络包要从一个地方传到另一个地方，除了要有确定的地址，还需要有定位功能。而有门牌号码属性的 IP 地址才具有指路功能。

比如：你去上海三新北路1800弄 C 栋 6 层找小明，你在路上问路，可能被问的人不知道 C 栋在那，但是可以给你指 三新北路 1800 弄在那，但是如果你指着小明的身份证号码问一个人小明在那，肯定没有人知道。

MAC 地址更像是身份证，是一个唯一的标识。它的唯一性设计是为了组网的时候，不同的网卡放在一个网络的时候，可以不用担心冲突。

MAC 地址是有一定的定位功能的，不过局限于一个子网内，所以 MAC 地址的通信范围比较小，局限在一个子网里面。例如，从 192.168.0.2/24 访问 192.168.0.3/24 可以通 MAC 地址。但是一旦跨子网，比如从 192.168.0.2/24 访问 192.168.1.2/24，MAC 地址就不行了，需要 IP 地址起作用。

网络设备的状态标识

`<BROADCAST,MULTICAST,UP,LOWER_UP>` 叫做 **net_device_flags**, 网络设备状态

- **UP** 表示网卡处于启动的状态。
- **BROADCAST** 表示这个网卡有广播地址，可以发送广播包。
- **MULTICAST** 表示网卡可以发送多播包。
- **LOWER_UP** 表示 L1 是启动的，也就是插着网线。
- **MTU1500**：最大传输单元为 1500，这是以太网的默认值。
 - MTU 是二层 MAC 层的概念，以太网规定二层不包括 MAC 头 14 字节和 MAC 尾 4 字节，剩下的内容合起来**不允许超过 1500 个字节**，正文里面有 IP 的头、TCP 的头，HTTP 的头，如果放不下，就需要分片来传输。
- **qdisc pfifo_fast**：qdisc 全称是 `queueing discipline`，中文叫**排队规则**。内核如果需要通过某个网络接口发送数据包，它都需要按照为这个接口配置的 qdisc 把数据包加入队列。
 - 最简单的排队规则是 pfifo，它不对进入的数据包作任何处理，数据包采用先入先出的方式通过队列。
 - pfifo_fast 稍微复杂一点，它的队列包含三个波段(band)。在每个波段里面，采用先进先出规则。三个波段的优先级也不同，band0 最高，band2 最低。如果 band0 里面有数据包，系统就不会处理 band1，band1 和 band2 关系类推。数据包是按照服务类型(**Type of Service TOS**) 被分配到三个波段里的。TOS 是 IP 头里的一个字段，代表了当前的包是高优先级的，还是低优先级的。