nmap -sS 103.10.87.148

nmap -sS 103.10.87.1-255

nmap -sT 103.10.87.1-255

nmap -sU 103.10.87.1-255

nmap -sU -p 103.10.87.1-255

nmap -sT -v 103.10.87.1-255

**1、TCP扫描（-sT）**

这是一种最为普通的扫描方法，这种扫描方法的特点是：扫描的速度快，准确性高，对操作者没有权限上的要求，但是容易被防火墙和IDS(防入侵系统)发现

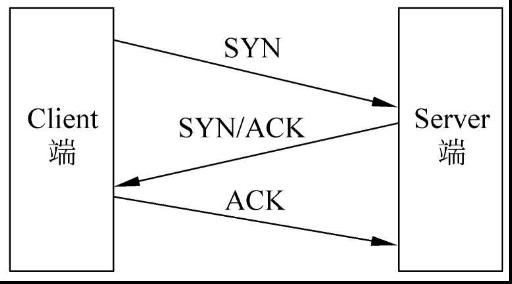
运行的原理：通过建立TCP的三次握手连接来进行信息的传递

① Client端发送SYN；

② Server端返回SYN/ACK，表明端口开放；

③ Client端返回ACK，表明连接已建立；

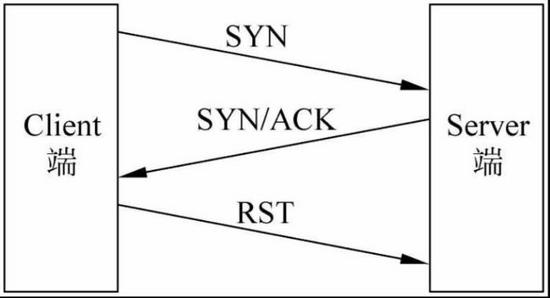
④ Client端主动断开连接。



**2、SYN扫描（-sS）**

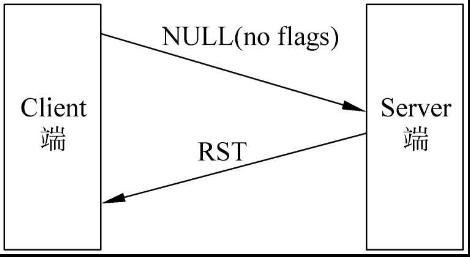
这是一种秘密的扫描方式之一，因为在SYN扫描中Client端和Server端没有形成3次握手，所以没有建立一个正常的TCP连接，因此不被防火墙和日志所记录，一般不会再目标主机上留下任何的痕迹，但是这种扫描是需要root权限（对于windows用户来说，是没有root权限这个概念的，root权限是linux的最高权限，对应windows的管理员权限）

运行的原理图如下：



**3、NULL扫描**

NULL扫描是一种反向的扫描方法，通过发送一个没有任何标志位的数据包给服务器，然后等待服务器的返回内容。**这种扫描的方法比前面提及的扫描方法要隐蔽很多，但是这种方法的准确度也是较低的**， 主要的**用途是用来判断操作系统是否为windows**，因为windows不遵守RFC 793标准，不论端口是开启还是关闭的都返回RST包



但是虽然NULL具有这样的一些用处，但是本人却认为不宜使用NULL

1、NULL方法的精确度不高，端口的状态返回的不是很准确

2、要获取目标主机的运行系统，可以使用参数(-O),来获取对于一些操作系统无法准确判断的，可以加上参数(-osscan-guess)

3、NULL扫描易被过滤

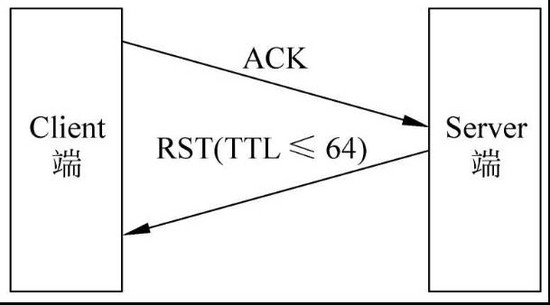
**4、FIN扫描**

FIN扫描的原理与NULL扫描的原理基本上是一样的在这里就不重复了

**5、ACK扫描**

ACK扫描的原理是发送一个ACK包给目标主机，不论目标主机的端口是否开启，都会返回相应的RST包，通过判断RST包中的TTL来判断端口是否开启

运行原理图：



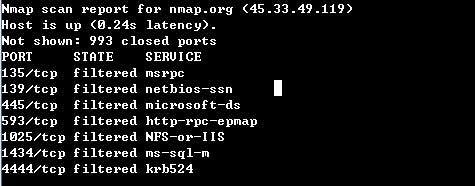
TTL值小于64端口开启，大于64端口关闭

大致上主要的扫描方法就是这些，除了我们可以按照这样些参数去执行扫描外，还可以自己定义一个TCP扫描包

**6、自定义TCP扫描包的参数为（--scanflags）**

例如：定制一个包含ACK扫描和SYN扫描的安装包

命令：nmap --scanflags ACKSYN nmap.org



 好了，接下来还有各种扫描方法的端口列表参数

**-PS 端口列表用,隔开[tcp80 syn 扫描]**  
**-PA 端口列表用,隔开[ack扫描](PS+PA测试状态包过滤防火墙【非状态的PA可以过】)【默认扫描端口1-1024】**  
**-PU 端口列表用,隔开[udp高端口扫描 穿越只过滤tcp的防火墙]**