如何用Iptables实现Linux下强大的NAT功能 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/264/2021_2022__E5_A6_82_E4_BD_95_E7_94_A8I_c103_264418.htm 本文主要介绍如何使用iptbales实现linux2.4下的强大的NAT功能。关于iptables的详细语法请参考“ 用iptales实现包过虑型防火墙 ”一文。需要申明的是，本文绝对不是 NAT-HOWTO的简单重复或是中文版，在整个的叙述过程中，作者都在试图用自己的语言来表达自己的理解，自己的思想。 一、概述 1. 什么是NAT 在传统的标准的TCP/IP通信过程中，所有的路由器仅仅是充当一个中间人的角色，也就是通常所说的存储转发，路由器并不会对转发的数据包进行修改，更为确切的说，除了将源MAC地址换成自己的MAC地址以外，路由器不会对转发的数据包做任何修改。NAT(Network Address Translation网络地址翻译)恰恰是出于某种特殊需要而对数据包的源ip地址、目的ip地址、源端口、目的端口进行改写的操作。 2. 为什么要进行NAT 我们来看看再什么情况下我们需要做NAT。 假设有一家ISP提供园区Internet接入服务，为了方便管理，该ISP分配给园区用户的IP地址都是伪IP，但是部分用户要求建立自己的WWW服务器对外发布信息，这时候我们就可以通过NAT来提供这种服务了。我们可以在防火墙的外部网卡上绑定多个合法IP地址，然后通过NAT技术使发给其中某一个IP地址的包转发至内部某一用户的WWW服务器上，然后再将该内部WWW服务器响应包伪装成该合法IP发出的包。 再比如使用拨号上网的网吧，因为只有一个合法的IP地址，必须采用某种手段让其他机器也可以上网，通常是采用代理服务器的方式，但是代

理服务器，尤其是应用层代理服务器，只能支持有限的协议，如果过了一段时间后又有新的服务出来，则只能等待代理服务器支持该新应用的升级版本。如果采用NAT来解决这个问题， 因为是在应用层以下进行处理，NAT不但可以获得很高的访问速度，而且可以无缝的支持任何新的服务或应用。还有一个方面的应用就是重定向，也就是当接收到一个包后，不是转发这个包，而是将其重定向到系统上的某一个应用程序。最常见的应用就是和squid配合使用成为透明代理，在对http流量进行缓存的同时，可以提供对Internet的无缝访问。 3. NAT的类型 在linux2.4的NAT-HOWTO中，作者从原理的角度将NAT分成了两种类型，即源NAT(SNAT)和目的NAT(DNAT)，顾名思义，所谓SNAT就是改变转发数据包的源地址，所谓DNAT就是改变转发数据包的目的地址。 二、原理 在 “ 用iptales实现包过虑型防火墙 ” 一文中我们说过，netfilter是Linux 核心中一个通用架构，它提供了一系列的"表"(tables)，每个表由若干"链"(chains)组成，而每条链中可以有一条或数条规则(rule)组成。并且系统缺省的表是"filter"。但是在使用NAT的时候，我们所使用的表不再是"filter"，而是"nat"表，所以我们必须使用"-t nat"选项来显式地指明这一点。因为系统缺省的表是"filter"，所以在使用filter功能时，我们没有必要显式的指明"-t filter"。 同filter表一样，nat表也有三条缺省的"链"(chains)，这三条链也是规则的容器，它们分别是: PREROUTING:可以在这里定义进行目的NAT的规则，因为路由器进行路由时只检查数据包的目的ip地址，所以为了使数据包得以正确路由，我们必须在路由之前就进行目的NAT. POSTROUTING:可以在这里定义进行源NAT的规则

，系统在决定了数据包的路由以后在执行该链中的规则。
OUTPUT:定义对本地产生的数据包的目的NAT规则。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com