Linux操作系统及应用技术

防火墙与Squid代理服务器的搭建及应用

防火墙的本义是指古代构筑和使用木制结构房屋的时候，为防止火灾的发生和蔓延，人们将坚固的石块堆砌在房屋周围作为屏障，这种防护构筑物就被称之为“防火墙”。我们通常所说的**网络防火墙**是借鉴了古代真正用于防火的防火墙的喻义，它指的是**隔离在本地网络与外界网络之间的一道防御系统。**防火墙可以使企业内部局域网（LAN）网络与Internet之间或者与其他外部网络互相隔离、限制网络互访用来保护内部网络。

其实与防火墙一起起作用的就是“门”。如果没有门，各房间的人如何沟通呢，这些房间的人又如何进去呢？当火灾发生时，这些人又如何逃离现场呢？这个门就相当于防火墙的“安全策略”，所以在此我们所说的防火墙实际并不是一堵实心墙，而是带有一些小孔的墙。这些小孔就是用来留给那些允许进行的通信，在这些小孔中安装了过滤机制，也就是“单向导通性”。

Squid是一个**缓存Internet数据**的软件，其接收用户的下载申请，并自动处理所下载的数据。当一个用户想要下载一个主页时，可以向Squid**发出一个申请**，要Squid代替其进行**下载**，然后Squid连接所申请网站并请求该主页，接着把该主页传给用户同时保留一个备份，当别的用户申请同样的页面时，Squid将保存的备份立即传给用户，使用户觉得速度相当快。Squid可以代理HTTP、FTP、GOPHER、SSL和WAIS等协议，并且Squid可以自动地进行处理，可以根据自己的需要设置Squid，使之过滤掉不想要的东西。

目录

本章要点

目录

本章要点

14.1.1 认识防火墙（firewall）

1.什么是防火墙

防火墙——是指设置在**不同网络**（如可信任的企业内部网和不可信的公共网）或网络安全域之间的一系列部件的组合。它是不同网络或网络安全域之间信息的唯一出入口，能根据企业的安全策略控制（允许、拒绝、监测）出入网络的信息流，且本身具有较强的抗攻击能力。

在逻辑上，防火墙是一个**分离器、限制器和分析器，**它能有效地监控内部网和Internet之间的任何活动，保证了内部网络的安全，如图14-1所示。

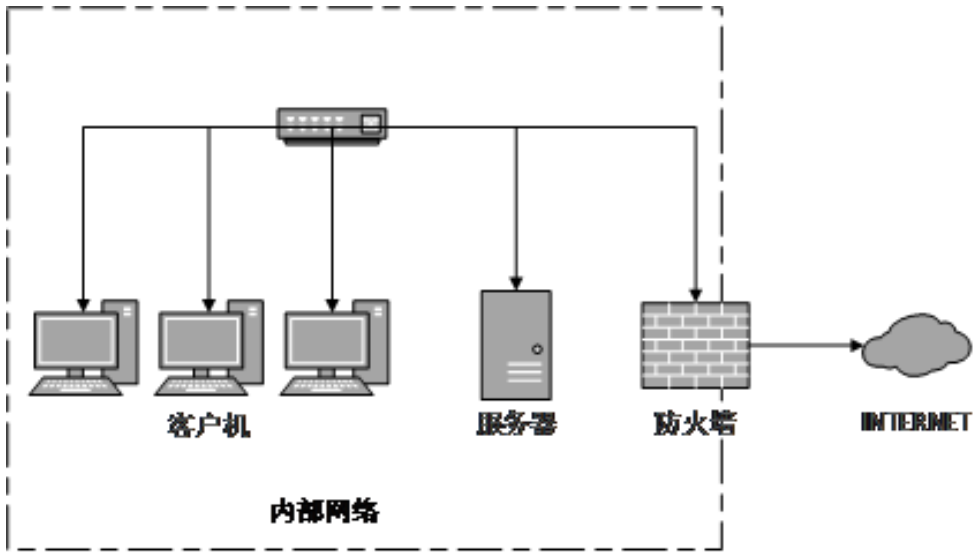


图14-1 防火墙

14.1.1 认识防火墙（firewall）

2．防火墙的功能

防火墙的主要功能如下：

- **过滤**进出网络的数据包，封堵某些禁止的访问行为。
- 对进出网络的访问行为作出日志记录，并提供网络使用情况的统计数据，实现对网络存取和访问的监控审计。
- 对**网络攻击**进行检测和告警。
- 防火墙可以保护网络免受基于路由的攻击，如IP选项中的源路由攻击和ICMP重定向中的重定向路径，并通知防火墙管理员。
- 提供数据包的路由选择和网络地址**转换（NAT）**，从而解决局域网中主机使用内部IP地址也能够顺利访问外部网络的应用需求。

14.1.1 认识防火墙（firewall）

3．防火墙的类型

（1）按采用的技术划分

① 包过滤型防火墙

在网络层或传输层对经过的数据包进行筛选。筛选的依据是系统内设置的过滤规则，通过检查数据流中每个数据包的IP源地址、IP目的地址、传输协议（TCP、UDP、ICMP等）、TCP/UDP端口号等因素，来决定是否允许该数据包通过。（包的大小1500字节）

② 代理服务器型防火墙

是运行在防火墙之上的一种应用层服务器程序，它通过对每种应用服务编制专门的代理程序，实现监视和控制应用层数据流的作用。

14.1.1 认识防火墙（firewall）

3．防火墙的类型

（2）按实现的环境划分

① 软件防火墙：

普通计算机+通用的操作系统（如：Linux），学校、网吧中常用。

② 硬件（芯片级）防火墙：

基于专门的硬件平台和固化在ASIC芯片来执行防火墙的策略和数据加解密，具有速度快、处理能力强、性能高、价格比较昂贵的特点，如NetScreen、FortiNet等。通常有3个以上网卡接口，如图14-2所示：

14.1.1 认识防火墙（firewall）

3．防火墙的类型

- 外网接口：用于连接Internet网；
- 内网接口：用于连接代理服务器或内部网络；
- DMZ接口（非军事化区）：专用于连接提供服务的服务器群。

图14-2 CheckPoint UTM-1 450

14.1.2 Linux防火墙概述

1．Linux防火墙的历史

从1.1内核开始，Linux系统就已经具有包过滤功能了，随着Linux内核版本的不断升级，Linux下的包过滤系统经历了如下3个阶段：

- 在2.0内核中，包过滤的机制是ipfw，管理防火墙的命令工具是ipfwadm。
- 在2.2内核中，包过滤的机制是ipchain，管理防火墙的命令工具是ipchains。
- 在2.4之后的内核中，包过滤的机制是netfilter，防火墙的命令工具是iptables。

2．Linux防火墙的架构

Linux防火墙系统由以下两个组件组成。

（1）netfilter

netfilter是集成在内核中的一部分，作用是定义、保存相应的过滤规则。提供了一系列的表，每个表由若干个链组成，而每条链可以由一条或若干条规则组成。

netfilter是**表的容器**，表是链的容器，而链又是规则的容器。表→链→规则的分层结构来组织规则，如图14-3所示。

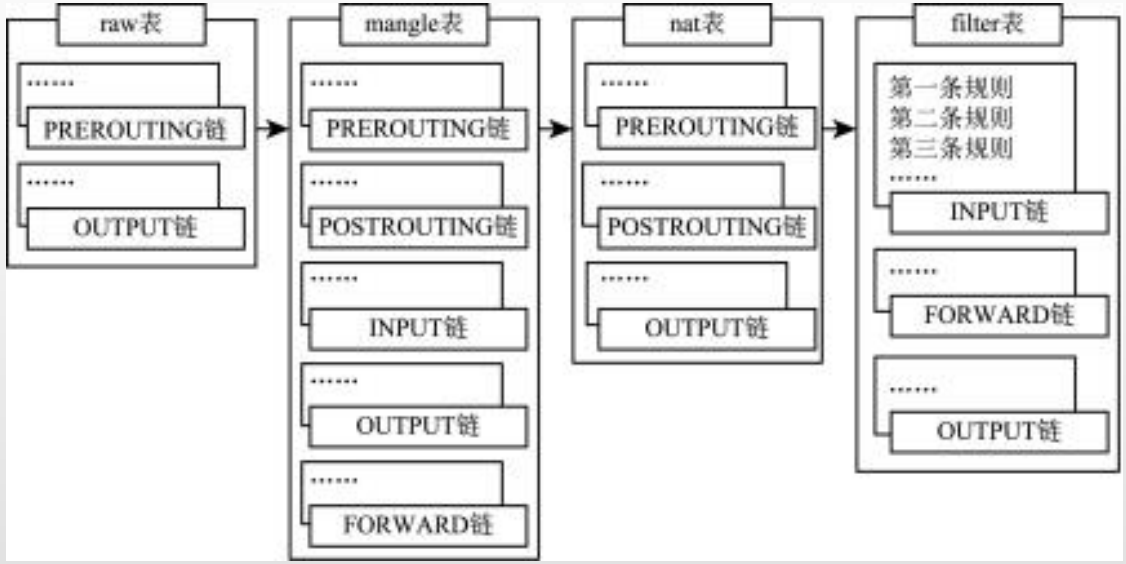


图14-3 netfilter结构

14.1.2 Linux防火墙概述

（2）iptables

iptables是Linux系统为用户提供的管理netfilter的一种工具，是编辑、修改防火墙（过滤）规则的编辑器。通过这些规则及其他配置，告诉内核的netfilter对来自某些源、前往某些目的地或具有某些协议类型的数据包如何处理。这些规则会保存在内核空间之中。

14.1.3 iptables规则的分层结构

1 表（tables）

这些表专表专用，作用分别如下：

filter表	nat表	mangle表	raw表
包过滤，含INPUT、FORWARD、OUTPUT三个链。	包地址修改：用于修改数据包的IP地址和端口号，即进行网络地址转换。含PREROUTING、POSTROUTING、OUTPUT三个链。	包重构：修改包的服务类型、生存周期以及为数据包设置Mark标记，以实现Qos（服务质量）、策略路由和网络流量整形等特殊应用。含PREROUTING、POSTROUTING、INPUT、OUTPUT和FORWARD五个链，	数据跟踪：用于数据包是否被状态跟踪机制处理，包含PREROUTING、OUTPUT两个链。

14.1.3 iptables规则的分层结构

2 链（chains）——处理的数据包流向的不同

1. **INPUT链**——当数据包源自外界并前往防火墙所在的本机（入站）时，即数据包的目的地址是本机时，则应用此链中的规则

2. **OUTPUT链**——当数据包源自防火墙所在的主机并要向外发送（出站）时，即数据包的源地址是本机时，则应用此链中的规则。

3. **FORWARD链**——当数据包源自外部系统，并经过防火墙所在主机前往另一个外部系统（转发）时，则应用此链中的规则。

4. **PREROUTING链**——当数据包到达防火墙所在的主机在作路由选择之前，且其源地址要被修改（源地址转换）时，则应用此链中的规则。

5. **POSTROUTING链**——当数据包在路由选择之后即将离开防火墙所在主机，且其目的地址要被修改（目的地址转换）时，则应用此链中的规则。

6. **用户自定义链。**

14.1.3 iptables规则的分层结构

3 规则（rules）

规则其实就是**网管员预定义的过滤筛选数据包的条件**。规则一般的定义为“如果数据包头符合这样的条件，就这样处理这个数据包”。当数据包与规则匹配时，iptables就根据规则所定义的动作来处理这些数据包（如放行、丢弃和拒绝等）。

配置防火墙的主要工作就是添加、修改和删除这些规则。学习防火墙就是学习这些规则如何去写。

14.1.4 数据包过滤匹配流程

表间的优先顺序，依次为：raw、mangle、nat、filter。

链间的匹配顺序：

- 入站数据：PREROUTING、INPUT
- 出站数据：OUTPUT、POSTROUTING
- 转发数据：PREROUTING、FORWARD、POSTROUTING

链内规则的匹配顺序：

- 按顺序依次进行检查，找到相匹配的规则即停止（LOG策略会有例外）
- 若在该链内找不到相匹配的规则，则按该链的默认策略处理

14.1.5 代理服务器Squid

1 Squid代理服务器的作用

Squid除了具有防火墙的代理、共享上网等功能外，还有以下特别的作用：

- 加快访问速度，节约通信带宽
- 多因素限制用户访问，记录用户行为

2 Squid代理服务器的工作流程

当**客户机**通过代理来请求**Web页面**时，指定的代理服务器会先检查自己的缓存，如果**缓存中**已经有客户机需要的页面，则直接将缓存中的页面内容反馈给客户机；如果缓存中没有客户机要访问的页面，则由**代理服务器**向Internet中发送访问请求，当获得返回的Web页面以后，将网页数据保存到缓存中并发送给客户机。

Squid代理服务器的工作流程，如图14-4所示。

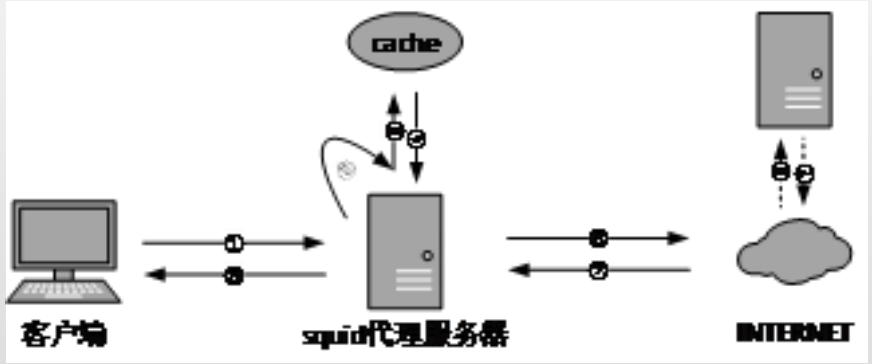


图14-4 Squid代理服务器的工作流程

14.1.5 代理服务器Squid

3 Squid代理服务器的分类及特点

Squid代理服务器按照代理的设置方式可以分为以下3种：

普通（标准）代理服务器

这种代理服务器需要在客户端的浏览器中设置代理服务器的地址和端口号。

透明代理服务器

透明代理是**NAT和代理**的完美结合，之所以称为透明，是因为在这种方式下用户感觉不到代理服务器的存在，不需要在浏览器或其他客户端工具（如网络快车、QQ、迅雷等）中作任何设置，客户机只需要将默认网关设置为代理服务器的IP地址便可。

反向代理服务器

普通代理和透明代理是为局域网用户访问Internet中的Web站点提供缓存代理，而反向代理恰恰相反，是为Internet中的用户访问企业局域网内的Web站点提供缓存加速。

目录

本章要点

1．安装iptables软件包

因为netfilter/iptables的netfilter组件是与内核集成在一起的，所以只需要安装iptables工具，默认情况下系统会安装该软件包，可通过下面命令检查是否已安装：

```
[root@Server1 ~]# rpm -qa |grep iptables
iptables-1.4.21-16.el7
```

因为netfilter/iptables的netfilter组件是与内核集成在一起的，所以只需要安装iptables工具，默认情况下系统会安装该软件包，可通过下面命令检查是否已安装：

2．iptables服务的常用管理命令

设置完成后可启动、停止或重启iptables服务了，有关命令如下：

（1）iptables服务的启动、停止或重新启动

```
service  iptables  start|stop|restart
```

（2）iptables服务的自动启动

```
# chkconfig  --level  345  iptables  on
# ntsysv
```

此命令打开文本图形界面，在iptables前面选中“*”，确定后即可实现开机自动加载iptables服务，否则取消掉“*”就不自动加载了。

iptables是对防火墙配置管理的核心命令，其基本格式为：

```
iptables [-t 表名] 命令选项 [链名] -[匹配条件] [-j 目标动作/跳转]
```

其中：

表名、链名——用于指定所操作的表和链，若未指定表名，则filter作为缺省表。

命令选项——指定管理规则的方式，常用的命令选项见表14-1。

表14-1 iptables命令的常用命令选项

命令选项	功能说明
-A或--append	在指定链的末尾添加一条新的规则
-D或--delete	删除指定链中的某一条规则，按规则序号或内容确定要删除的规则
-I或--insert	在指定链中插入一条新的规则，若未指定插入位置则默认在链的开头
-L或--list	列出指定链中所有规则以供查看，若未指定链名，则列出表中所有链的内容。若要同时显示规则在链中的序号，再加--line-numbers选项；若要以数字形式显示输出结果，则再加-n选项

表14-1 iptables命令的常用命令选项

命令选项	功能说明
-n或--numeric	使用数字形式显示输出结果，如显示主机IP地址而不是主机名
--line-numbers	查看规则列表时，同时显示规则在链中的序号
-v或--verbose	查看规则列表时，显示数据包的个数、字节数等详细的信息
-R或-replace	替换指定链中的某一条规则，按规则序号或内容确定要替换的规则
-X或--delete-chain	删除指定表中的用户自定义链。该链必须没有被引用，如果被引用，在删除之前你必须删除或者替换与之有关的规则。如果没有给出参数，这条命令将试着删除每个用户自定义的链
-P或--policy	设置指定链的默认策略
-h或--help	查看iptables命令的帮助信息

匹配条件——用于指定对符合什么样的条件的包进行处理，常用条件匹配见表14-2。

表14-2 iptables命令的常用匹配条件

条件匹配	功能说明
-i或--in-interface [!] <网络接口名>	指定数据包从哪个网络接口进入，如ppp0、eth0、eth1也可以使用通配符，如eth+，表示所有以太网接口。!表示除去该接口之外的其他接口
-o或--out-interface [!] <网络接口名>	指定数据包从哪个网络接口输出
-p或--proto [!] <协议类型>	指定数据包匹配的协议，可以是/etc/protocols中定义的协议，如tcp、udp和icmp等
-s或--source [!] <源地址或子网>	指定数据包匹配的源IP地址或子网
-d或--dstination [!] <目的地址或子网>	指定数据包匹配的目的IP地址子网
--sport [!] <源端口号> [:<源端口号>]	指定数据包匹配的源端口号或端口范围
--dport [!] <目的端口号> [:<目的端口号>]	指定数据包匹配的目的端口号或端口范围

目标动作/跳转——用来指定内核对数据包的处理方式，如允许通过、拒绝、丢弃或跳转给其他链进行处理等，常用目标动作/跳转见表14-3。

表14-3 iptables命令的目标动作/跳转

目标动作/跳转	功能说明
ACCEPT	接受数据包
DROP	丢弃数据包，不给出任何回应信息
REJECT	丢弃数据包，并给数据发送端返回一个回应信息
REDIRECT	将数据包重定向到本机或另一台主机的某个端口，通常用于实现透明代理或向外网开放内网的某些服务
SNAT	源地址转换，即改变数据包的源IP地址
DNAT	目的地址转换，即改变数据包的目的IP地址
MASQUERADE	IP地址，即NAT技术，MASQUERADE只能用于ADSL等拨号上网的IP伪装，也就是IP地址是由ISP动态分配的，如果是静态固定的，则使用SNAT
LOG	将符号规则的数据包的相关信息记录在/var/log/messages目录的日志文件中，方便管理员进行分析和查错，然后继续匹配下一条规则

1) 添加、插入规则

```
# iptables  -t  filter  -A  INPUT  -p  tcp  -j  ACCEPT
# iptables  -I  INPUT  -p  udp  -j  ACCEPT
# iptables  -I  INPUT  2  -p  icmp  -j  ACCEPT
```

2) 查看规则

```
# iptables  -t filter -L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target      prot opt source               destination
1    ACCEPT      udp  --  anywhere             anywhere
2    ACCEPT      icmp --  anywhere             anywhere
3    ACCEPT      tcp  --  anywhere             anywhere
# iptables -vnL INPUT
```

注意

L选项要放在vn后，否则会将vn当成链名。

3) 创建、删除用户自定义链

```
# iptables  -t  filter  -N  hnwy
```

在filter表中创建一条用户自定义的链，链名为hnwy

```
# iptables  -t  filter  -X
```

清空filter表中所有自定义的链。

4) 删除、清空规则

```
# iptables   -D  INPUT 3
# iptables  -F
# iptables   -t  nat   -F
```

5] 设置内置链的默认策略

当数据包与链中所有规则都不匹配时，将根据链的默认策略来处理数据包。

默认允许的策略：

首先默认允许接受所有的输入、输出、转发包，然后拒绝某些危险包，没有被拒绝的都被允许。这种方式灵活方便，但安全性不高。

```
# iptables  -P  OUTPUT  ACCEPT
```

默认禁止的策略：

首先拒绝所有的输入、输出、转发包，然后根据需要逐个打开要开放的各项服务，没有明确确允许的都被拒绝。这种方式安全性高，但不灵活。通常采用默认禁止的策略。

```
# iptables  -t  filter  -P  FORWARD  DROP
```

6] 匹配条件的设置

匹配条件是识别不同数据包的依据，它详细描述了数据包的某种特征，以使这个包区别于其他所有的包。

iptables的匹配条件有3类：

1 **通用匹配条件——不依赖于其他匹配条件和扩展模块，可直接使用。包括协议、地址和网络接口匹配。**

① iptables -I FORWARD -s 10.10.1.0/24 -j ACCEPT

允许内网10.10.1.0/24子网里所有的客户机上Internet网

② iptables -I INPUT -p icmp -s 172.16.102.36 -j DROP

③ iptables -I INPUT -i eth0 -p icmp -j DROP

禁止Internet上的计算机通过ICMP协议ping到本服务器的上连接公网的网络接口eth0。

2 隐含条件匹配——以协议匹配为前提

① 端口匹配——以协议匹配为前提，不能单独使用端口匹配

```
# iptables -I FORWARD  -p tcp --dport  20：21 -j DROP
```

禁止内网的10.10.1.0/24子网里所有的客户机使用FTP协议下载

② TCP标记匹配

用于检查数据包的TCP标记位（--tcp-flags），以便有选择的过滤不同类型的TCP数据包，使用格式为：

```
--tcp-flags 检查范围 被设置的标记
```

- 检查范围：用于指定要检查哪些标记（可识别的标记有：SYN，ACK，FIN，RST，URG，PSH）。
- 被设置的标记：指定在“检查范围”中出现过且被设为1（即状态是打开的）的标记。

③ #iptables -I INPUT -p tcp --tcp-flags ！SYN，FIN，ACK SYN -j DROP

禁止那些FIN和ACK标记被设置而SYN标记未设置的数据包。

④ ICMP类型匹配——以“-p icmp”协议匹配为前提

用于检查ICMP数据包的类型（--icmp-type），以便有选择地过滤不同类型的ICMP数据包。使用格式为：

--icmp-type ICMP类型

ICMP类型可为：“Echo-Request” “Echo-Reply” “Destination-Unreachable”，分别对应ICMP协议的请求、回显、目标不可达数据。

- # iptables -A INPUT -p icmp --icmp-type Echo-Request -j DROP
- # iptables -A INPUT -p icmp --icmp-type Echo-Reply -j ACCEPT
- # iptables -A INPUT -p icmp --icmp-type Destination-Unreachable -j ACCEPT

禁止其他主机ping到本服务器，但允许从本服务器上ping其他主机（允许接收ICMP回应数据包）。

3 显式条件匹配

这种匹配的功能需要由额外的内核模块提供，因此需要在iptables命令中使用“-m模块关键字”的形式调用相应功能的内核模块。

常见的显式条件匹配有：MAC地址匹配、非连续的多端口匹配、多IP地址匹配、状态检测匹配。

```
# iptables  -I FORWARD  -m mac --mac-source  00-19-21-F1-83-C7  -j DROP
```

禁止转发来自MAC地址为00-19-21-F1-83-C7的主机的数据包。

```
# iptables  -I INPUT -p tcp  -m multiport  --dport 20,21,53  -j ACCEPT
```

允许开放本机的20、21、53等TCP端口。

7 规则的保存与恢复

使用iptables命令手工进行设置，在系统中是即时生效的，但如果不进行保存将在系统下次启动时丢失。

（1）保存防火墙规则

命令1：service iptables save

将当前正在运行的防火墙规则，保存到“/etc/sysconfig/iptables”文件中，文件原有的内容将被覆盖。iptables每次启动或重启时都使用/etc/sysconfig/iptables文件中所提供的规则进行规则恢复。

在保存防火墙当前配置前应先将原有配置进行备份

```
cp /etc/sysconfig/iptables  iptables.raw
```

命令2：iptables-save

将配置信息显示到标准输出（屏幕）中

命令3：iptables-save >路径/文件名

将显示到标准输出（屏幕）中的当前正在运行的防火墙规则配置信息重定向保存到指定目录的指定文件中。

service iptables save命令等效于iptables-save > /etc/sysconfig/iptables命令；使用iptables-save命令可以将多个版本的配置保存到不同的文件中。

```
# iptables-save  >/etc/sysconfig/ipt.v1.0
# service  iptables  save
```

将当前运行的防火墙规则先后保存到用户指定的配置文件和系统默认的配置文件。

（2）恢复防火墙规则

命令：iptables-restore <路径/文件名

功能：将使用iptables-save保存的规则文件中的规则恢复到当前系统中。

iptables-restore命令可恢复不同版本的防火墙配置文件。

```
# iptables-save  > /etc/sysconfig/iptables
# service  iptables  restart
```

实例1
管理icmp

禁止某机（物理机）ping防火墙所在主机（虚拟机）

```
service  iptables  start   启动防火墙
iptables   -F                清空所有规则
iptables  -A INPUT -p  icmp  -s  172.16.102.X  -j  DROP
```

禁止除某机以外的其他主机ping防火墙所在的主机

```
iptables  -D   INPUT  1
iptables  -A INPUT -p  icmp  -s  ！172.16.102.X  -j  DROP
```

禁止本网段以外的主机ping本机

```
iptables  -D   INPUT  1
iptables  -A INPUT -s  ！ 172.16.102.0/24  -p icmp -j DROP
```

禁止所有人ping本机

```
iptables  -D   INPUT  1
iptables   -A   INPUT  -p  icmp  -j   DROP
```

实例2
设置远程登录限制

初始化：

```
iptables  -P  INPUT  DROP
iptables  -P  OUTPUT  DROP
iptables  -P  FORWARD  DROP
iptables  –F
iptables  –X
```

仅允许某机172.16.102.36使用ssh连接防火墙

```
iptables  -A  INPUT  -s  172.16.102.X  -p  tcp  --dport 22  -j  ACCEPT
iptables  -A  OUTPUT  -d  172.16.102.X  -p  tcp  --sport 22  -j  DROP
```

实例3——作为专门Web服务器终端的配置

要让局域网内的计算机访问内部网的FTP、Web等服务器，实现如图14-5的应用场景，应在防火墙上按如下步骤配置：

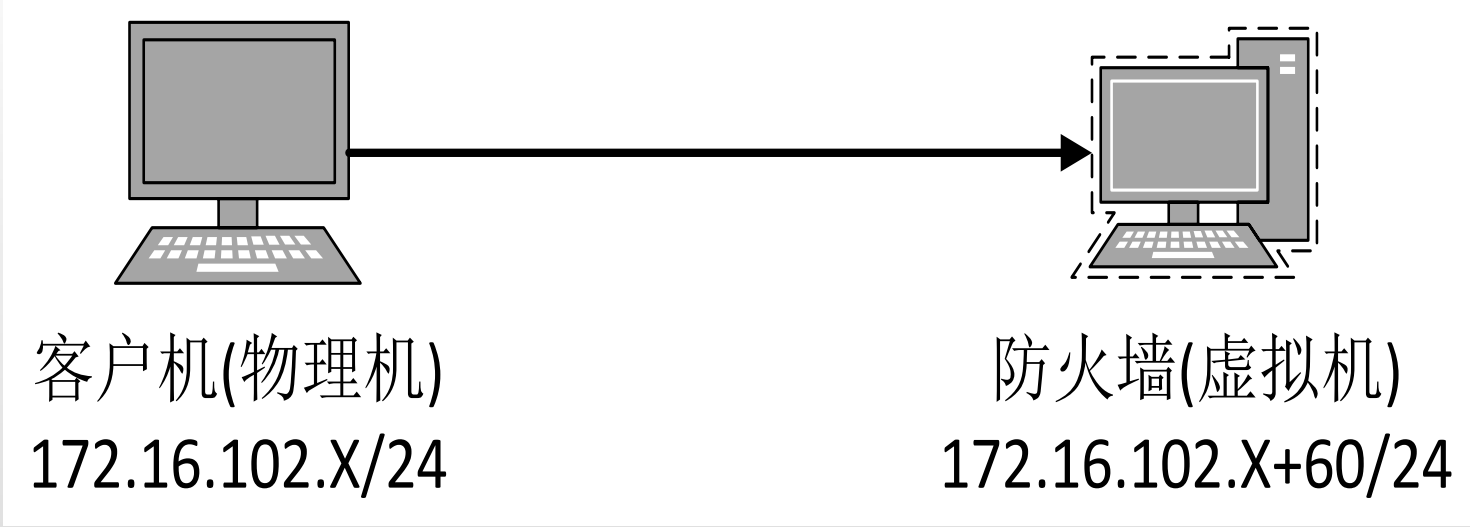


图14-5 作为专门Web服务器终端的配置

实例3——作为专门Web服务器终端的配置

1．初始化防火墙，清除任何以前配置的规则

```
iptables  -F       //清除filter规则表中的所有规则
iptables  -X       //清除filter规则表中的自定义规则链
iptables  -Z       //将指定表中的数据包计数器和流量计数器归零
```

2．使客户机能远程登录访问服务器

```
iptables -A INPUT -p tcp -d 172.16.102.X+60 --dport 22  -j ACCEPT
iptables -A OUTPUT -p tcp - s 172.16.102.X+60 --sport 22  -j ACCEPT
```

3．把所有默认策略设置为DROP

```
iptables  -P  INPUT  DROP
iptables  -P  OUTPUT  DROP
iptables  -P  FORWARD  DROP
```

实例3——作为专门Web服务器终端的配置

4．让本机的回环设备可以使用

```
iptables  -I  INPUT  1  -i lo -p all  -j ACCEPT
iptables  -I  OUTPUT 1  -o lo   -j ACCEPT
```

没有上述两条规则时，系统启动会卡住

5．客户机通过80端口访问WEB服务器

```
iptables -A INPUT -p tcp --sport 80 -j ACCEPT
iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
```

6．使防火墙能够解析进出来的包

```
iptables -A INPUT -p udp --sport 53 -j ACCEPT
iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
```

实例3——作为专门Web服务器终端的配置

7．保存设置

```
service  iptables  save
```

请在启动Linux系统之前添加第二块网卡。

第1步：启动虚拟机软件。

第2步：添加第二块网卡。

在虚拟机软件菜单上单击“虚拟机”→“设置”→“添加”，接着单击“网络适配器”→“下一步”，选择“网桥”，单击“完成”→“确定”。

第3步：启动RHEL7.2。

第4步：配置IP地址。

第一块网卡eth0：172.16.102.X+60

第二块网卡eth1：10.10. 1.X/24

注意

其中：X——为物理机网卡IP地址的第4段。

目录

本章要点

IP地址的分配与管理由ICANN管理机构负责，公网地址必须经申请后才能合法使用。为解决IP地址资源紧缺问题，IANA机构将IP地址划分了一部分出来，将其规定为私网地址，只能在局域网内使用，不同局域网可重复使用。

可使用的私网地址有：

一个A类地址： 10.0.0.0/8

16个B类地址： 172.16.0.0/16～172.31.0.0/16

256个C类地址： 192.168.0.0/16

1 NAT服务的概念及分类

为了解决使用私网地址的局域网用户访问因特网的问题，从而诞生了**网络地址转换（NAT）技术。**

NAT（Network AddressTranslation，网络地址转换）是一种用另一个地址来替换IP数据包头部中的源地址或目的地址的技术。

通过网络地址转换操作，局域网用户就能透明地访问因特网，通过配置静态地址转换，位于因特网中的主机还能实现对局域网内特定主机的访问。

目前几乎所有防火墙的软硬件产品都集成了NAT功能，iptables也不例外。

根据NAT替换数据包头部中地址的不同，NAT分为源地址转换SNAT（IP伪装）和目的地址转换DNAT两大类。

SNAT策略的原理如图14-6所示。

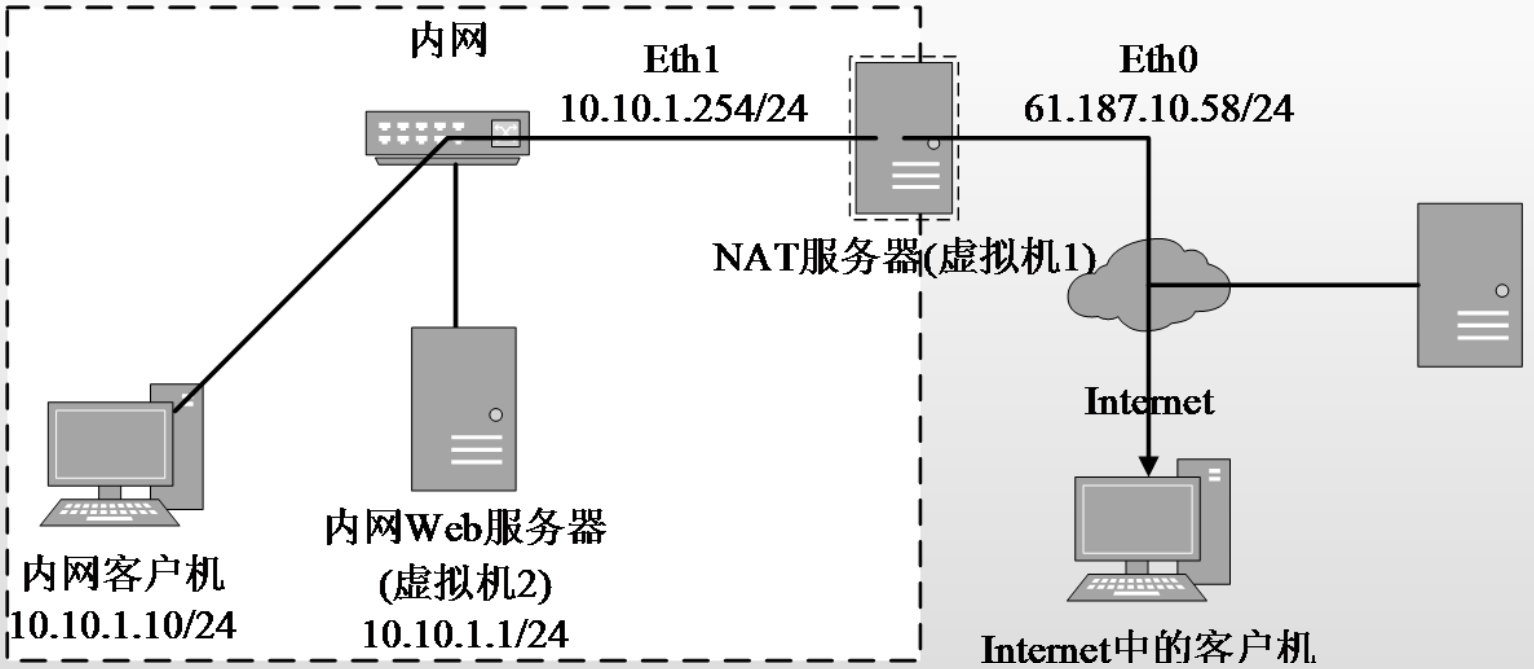


图14-6 SNAT策略的原理

未使用SNAT策略时的情况，如图14-7所示。

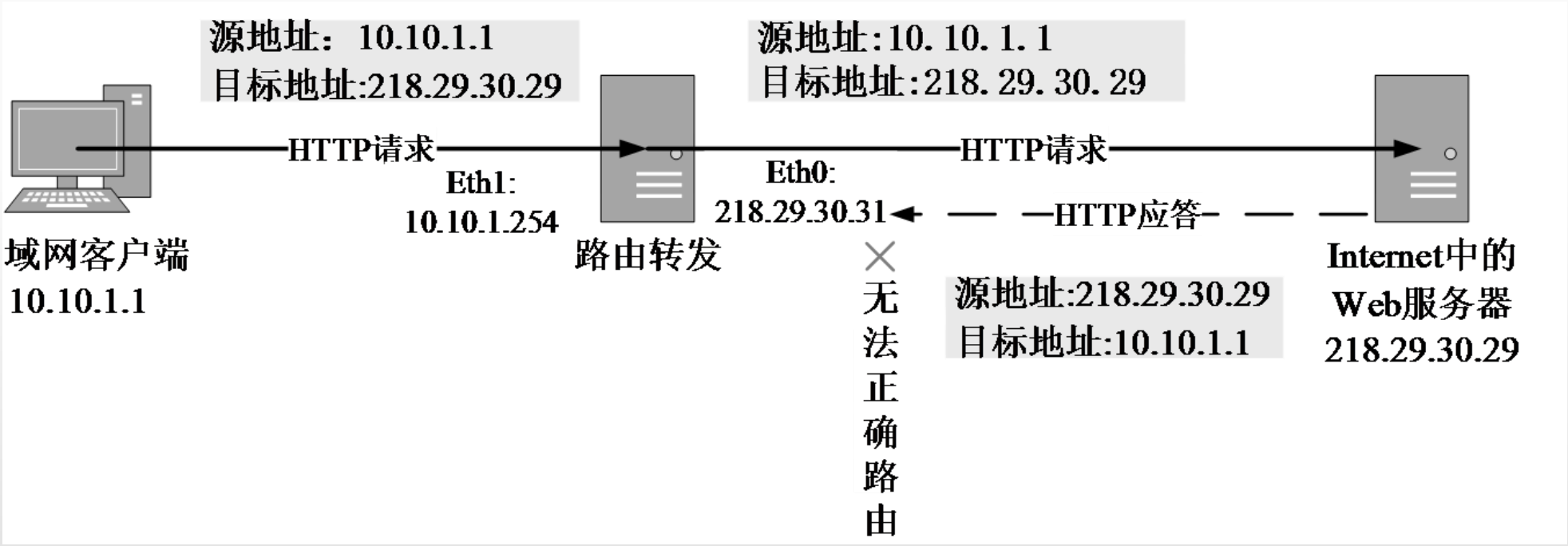


图14-7 Linux网关服务器

在网关中使用SNAT策略以后，如图14-8所示。

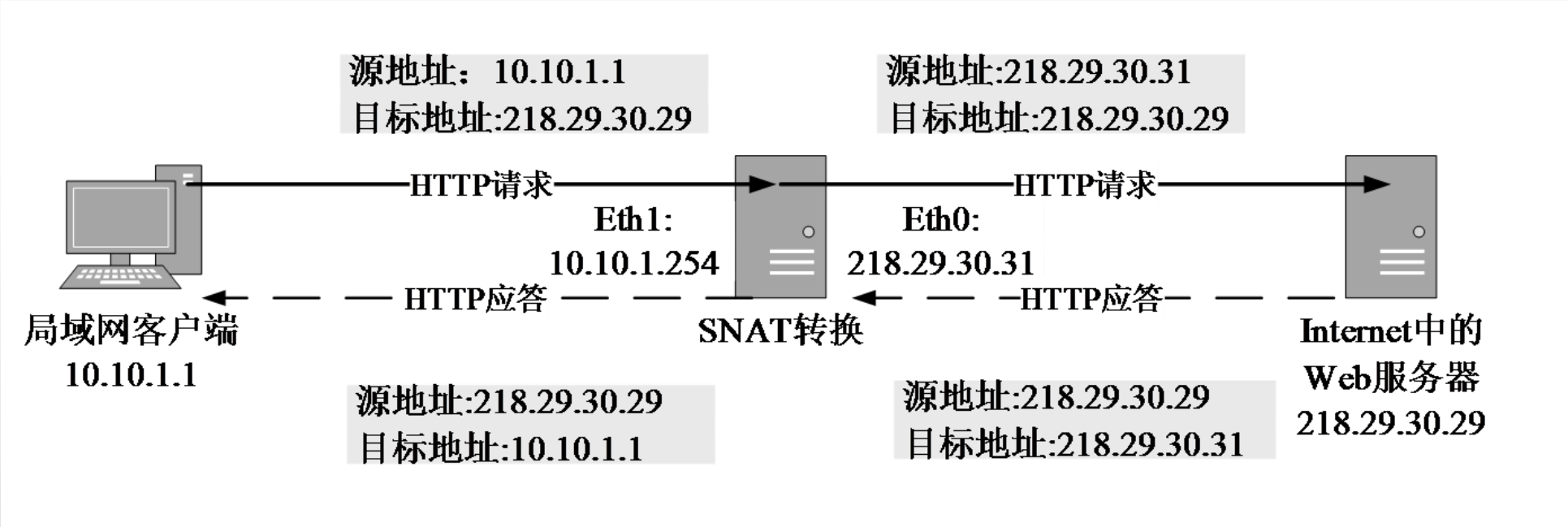
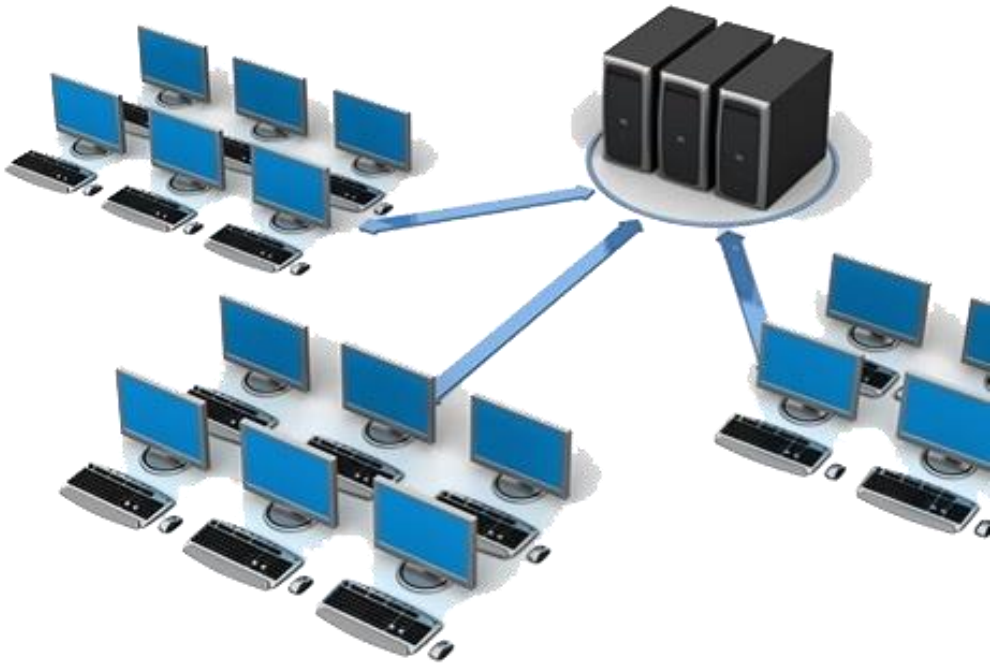


图14-8 Linux网关服务器



局域网用户的访问请求报文中的源地址是**私网地址**，报文在进入因特网后，将被因特网中的路由器丢弃。

利用网络地址转换技术，在报文离开局域网的边界路由器进入因特网之前，对报文中的源地址进行替换修改，将其替换修改为某一个合法的公网地址，这样报文就能在因特网中被正常路由和转发了，访问就会获得成功。

这种对报文中的源地址或目的地址进行替换修改的操作，就称为网络地址转换。

2 使用SNAT实现使用私网IP的多台主机共享上网

为落实上述SNAT技术的结果，要在NAT服务器上完成以下两个操作，如图14-9所示。

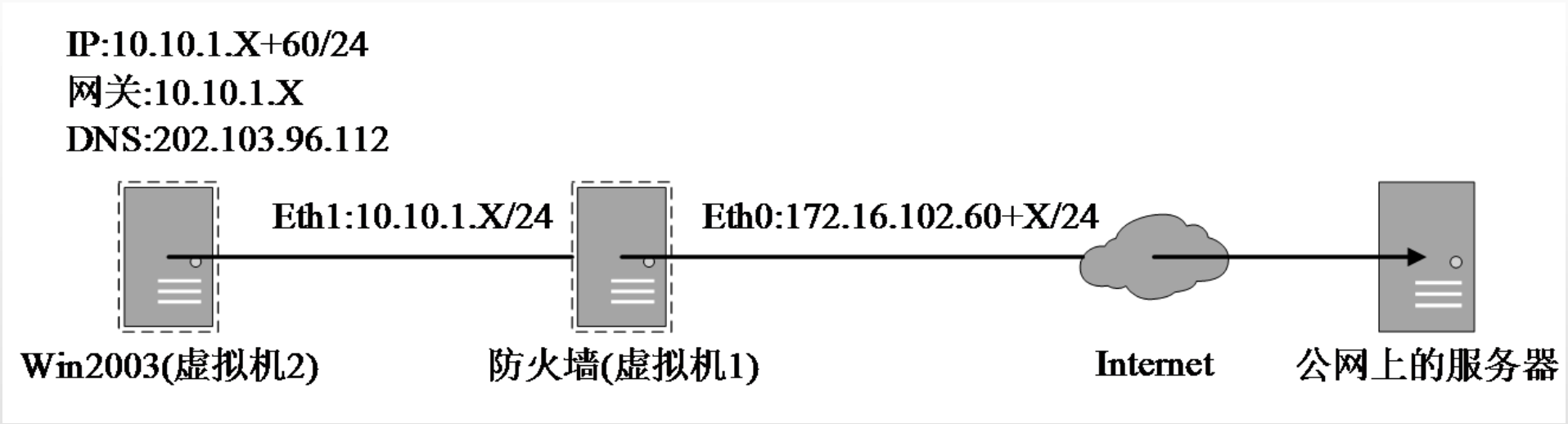


图14-9 NAT服务器操作

步骤 1 ▶

开启Linux内核IP报文转发功能。允许NAT服务器上的eth0和eth1两块网卡之间能相互转发数据包。有两种开启方法：

方法1

编辑/etc/sysctl.conf配置文件，将“net.ipv4.ip_forward=0”配置项修改为：

```
net.ipv4.ip_forward=1
# sysctl  -p  /etc/sysctl.conf
```

方法2

执行命令，使sysctl.conf的修改立即生效。

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
# sysctl  -p  /etc/sysctl.conf
```

步骤 2 ▶

添加使用SNAT策略的防火墙规则。

当NAT服务器的外网接口配置的是固定的公网IP地址

```
# iptables -t nat -A POSTROUTING -s 10.10.1.0/24 -o eth0 -j SNAT
--to-source 172.16.102.60+X
```

当NAT服务器通过ADSL拨号方式连接Internet，即外网接口获取的是动态公网IP地址：

```
# iptables -t nat -A POSTROUTING -s
10.10. 1.0/24 -o  ppp0  -j  MASQUERADE
```

DNAT策略的原理如图14-10所示。

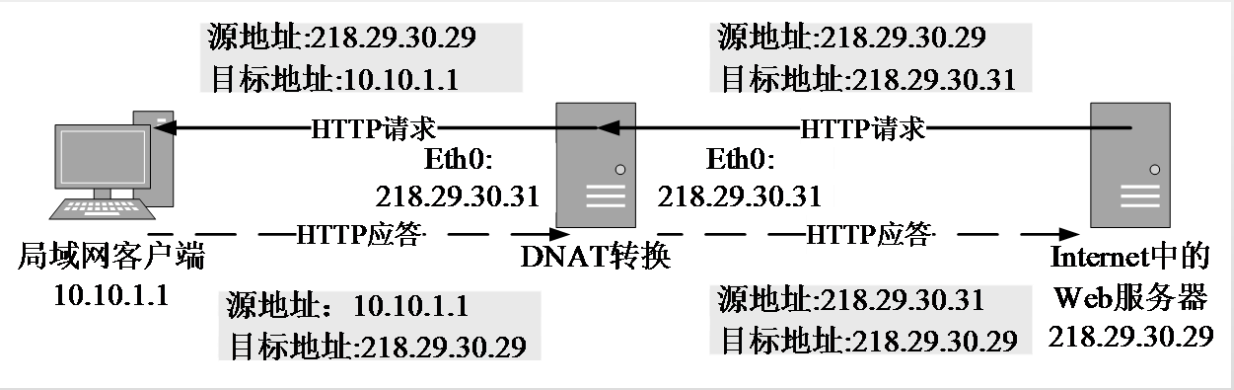


图14-10 在网关中使用DNAT策略发布内网服务器

3 使用DNAT实现向公网发布私网的应用服务器

步骤 1 ▶

确认已开启网关的路由转发功能（方法同上）

步骤 2 ▶

添加使用DNAT策略的防火墙规则。若发布的是Web服务器，其命令如下：

```
# iptables -t nat -A PREROUTING  -p tcp -i eth0 -d 172.16.102.60+X --dport 80
-jDNAT --to-destination 10.10.1. X
```

下面我们来看一个NAT配置案例，企业环境如图14-11所示。

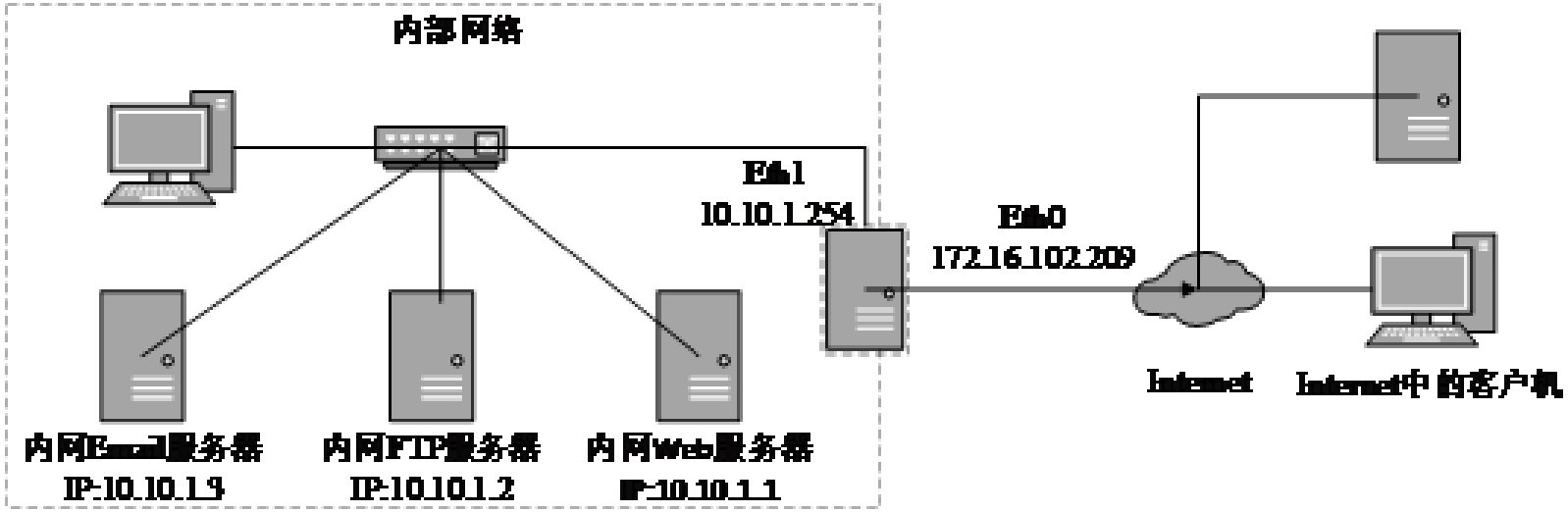


图14-11 企业环境

需求分析：

所有内网计算机要访问互联网，Mail和FTP服务器对内部员工开放，对外发布Web站点，管理员通过外网对Web站点进行远程管理。

首先，删除所有规则设置，将默认规则设置为DROP；然后开启防火墙对于客户端的访问限制，打开Web、MSN、QQ及MAIL的相应端口，允许外部客户端登录Web服务器的80、22端口。

解决方案：

1. 安装iptables

2. 查看本机关于iptables的设置情况

```
# iptables -L -n
```

3 清除原有规则

不管你在安装Linux时是否启动了防火墙，如果你想配置属于自己的防火墙，那就清除现在filter的所有规则。

iptables –F

清空所选链中的规则，如果没有指定链则清空指定表中所有链的规则。

iptables –X

清除预设表filter中使用者自定链中的规则。

iptables –Z

清除预设表filter中使用者自定链中的规则。

4 设置默认策略

```
iptables  –P   INPPUT  DROP
iptables  –P   FORWARD  DROP
iptables  –P   OUTPUT  ACCEPT
iptables  –t  nat  –P  PREROUTING ACCEPT
iptables  –t  nat  –P  OUTPUT ACCEPT
iptables  –t  nat  –P  POSTROUTING  ACCE
```

注意

默认全部链都是开启的，所以有些命令可以不操作，另外mangle表本文没用到，所以不做处理，mangle主要用在数据包的特殊变更处理上，比如修改TOS等特性。

设置默认策略为关闭filter表的INPPUT及FORWARD链开启OUTPUT链，nat表的三个链PREROUTING、OUTPUT、POSTROUTING全部开启。

5 设置回环地址

```
iptables -A INPUT -i lo  -j  ACCEPT
```

有些服务的测试需要使用回环地址，为了保证各个服务的正常工作，需要允许回环地址的通信。

6 连接状态设置

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED  -j ACCEPT
```

为了简化防火墙的配置操作，并提高检查的效率，需要添加连接状态设置。

连接跟踪存在四种数据包状态

① **NEW**——想要新建连接的数据包

② **INVALID**——无效的数据包，例如损坏或者不完整的数据包

③ **ESTABLISHED**——已经建立连接的数据包

④ **RELATED**——与已经发送的数据包有关的数据包

7 设置80端口转发

```
iptables  -A  FORWARD  -p  tcp  --dport 80  -j  ACCEPT
```

公司网站需要对外开放，需要开放80端口。

8 DNS相关设置

```
iptables  -A FORWARD  -p tcp  --dport 53  -j ACCEPT
iptables  -A FORWARD  -p udp  --dport 53  -j ACCEPT
```

为了客户端能够正常使用域名访问互联网，需要允许内网计算机与外部DNS服务器的数据转发。

注意

开启DNS使用UDP、TCP的53端口。

9 允许管理员通过外网进行远程管理，开启22端口

```
iptables  -A  INPUT     -p tcp   --dport 22  -j ACCEPT
iptables  -A  OUTPUT -p tcp   --sport 22  -j ACCEPT
```

10 允许内网主机登录MSN和QQ相关设置

```
iptables -A FORWARD -p tcp --dport 1863 -j ACCEPT
iptables -A FORWARD -p tcp --dport 443 -j ACCEPT
iptables -A FORWARD -p tcp --dport 8000 -j ACCEPT
iptables -A FORWARD -p udp --dport 8000 -j ACCEPT
iptables -A FORWARD -p udp --dport 4000 -j ACCEPT
```

QQ能够使用TCP80、8000、443及UDP8000、4000登录，而MSN通过TCP1863、443验证。因此只需要允许这些端口的FORWARD转发即可以正常登录。

11 允许内网主机登录MSN和QQ相关设置

客户端发送邮件时访问邮件服务器的TCP25端口。接收邮件时访问，可能使用的端口较多。UDP协议以及TCP协议的端口：110、143、993和995。

```
smtp:
# iptables -A FORWARD -p tcp --dport 25 -j ACCEPT
pop3:
# iptables -A FORWARD -p tcp --dport 110 -j ACCEPT
# iptables -A FORWARD -p udp --dport 110 -j ACCEPT
imap:
# iptables -A FORWARD -p tcp --dport 143 -j ACCEPT
# iptables -A FORWARD -p udp --dport 143 -j ACCEPT
imaps:
# iptables -A FORWARD -p tcp --dport 993 -j ACCEPT
# iptables -A FORWARD -p udp --dport 993 -j ACCEPT
pop3s:
# iptables -A FORWARD -p tcp --dport 995 -j ACCEPT
# iptables -A FORWARD -p udp --dport 995 -j ACCEPT
```

12 NAT端口映射设置

```
iptables -t nat -A POSTROUTING -o eth0 -s
10.10.1.0/24 -j SNAT --to-source 172.16.102.209
```

由于局域网的地址为私网地址，在公网上不合法，所以必须将私网地址转为服务器的外部地址进行地址映射，连接外网接口为eth0。

13 内网机器对外发布Web网站

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT
--to-destination 10.10.1.1
```

内网Web服务器IP地址为10.10.1.1，我们需要进行如下配置，当公网客户端访问服务器时，防火墙将请求映射到内网的10.10.1.1的80端口。

14 禁止访问具体域名和IP地址

禁止访问QQ主页

```
# iptables -A FORWARD -d [url]www.qq.com[/url] -j DROP
```

禁止访问指定IP地址

```
# iptables -A FORWARD -d 119.147.15.17 -j DROP
```

15 禁止INTERNET上计算机通过ICMP协议PING到代理服务器的eth0接口

```
#iptables –A  INPUT -i eth0 -p icmp -j DROP
```

此时，局域网中的计算机还是可以ping通Internet上的计算机的，因为从局域网到Internet的数据包使用NAT方式传输，仅经过PREROUTING链——FORWARD链——POSTROUTING链——链这条通道，并没有经过INPUT和OUPUT链。

16 保存与恢复iptables配置

```
iptables-save > /etc/iptables-save
```

保存

```
iptables-restore < /etc/iptables-save
```

恢复

```
service iptables save
```

如果要在服务或系统重启后依然生效。

17 重启服务

```
# service iptables restart
```

现在iptables配置表里什么配置都没有了。

目录

本章要点

1 检查是否安装了Squid服务器

```
rpm  -qa | grep  squid
```

2 安装Squid软件包

```
mount  /dev/cdrom  /mnt
rpm -ivh /mnt/Server/squid-2.6.STABLE21-3.el5.i386.rpm
```

目录

本章要点

1．设置监听的端口和IP地址

http_port 3128

2．设置内存缓冲大小

cache_mem 512MB

3．设置保存到缓存的最大文件的大小

maximum_object_size 4096 KB

4．设置用户下载的最大文件的大小

reply_body_max_size 10240000 allow all

5．设置硬盘缓存的大小

cache_dir ufs /var/spool/squid 4096 16 256

6．设置DNS服务器的地址

dns_nameservers 61.144.56.101

7．设置运行Squid主机的名称

visible_hostname 10.10.1.254

8．设置访问控制

acl 列表名称列表类型 [-i] 列表值1 列表值2...

9．设置日志文件

① 用户访问因特网的日志——cache_access_log /var/log/squid/access.log

access.log文件中包含了对Squid发起的每个终端客户请求，每个请求有一行记录。假如因为某些原因，不想让Squid记录终端客户请求日志，则可以设定日志文件的路径为“/dev/null”，或用“cache_access_log none”语句取消。

② 缓存日志文件——cache_log /var/log/squid/cache.log

cache.log包含了状态性的和调试性的消息。

③ 缓存中网站传输情况的日志文件——cache_store_log /var/log/squid/store.log

store.log文件包含了进入和离开缓存的每个目标的记录，平均记录大小典型的为175～200字节。

Squid最重要的日志文件是“/var/log/squid/access.log”，该日志文件记录了客户使用代理服务器的许多有用信息，共包含10个字段，每个字段的含义如表14-4所示。

表14-4 日志文件中的字段信息描述

字段	描述
time	记录客户访问代理服务器的时间，从1970年1月1日到访问时所经历的秒数，精确到毫秒
eclapsed	记录处理缓存所花费的时间，以毫秒计数
remotehost	记录访问客户端的IP地址或者域名
code/status	结果信息编码/状态信息编码，如TCP_MISS/205
bytes	缓存字节数
method	HTTP请求方法：GET或者POST
URL	访问的目的地址的URL，如www.sina.com.cn
rfc931	默认的，暂未使用
peerstatus/peerhost	缓存级别/目的IP地址，如DIRECT/211.163.21.19
type	缓存对象类型，如text/html

10．初始化Squid缓存目录

成功安装并配置好Squid服务器后，为了能够使Squid在硬盘中缓存用户访问目标服务器的内容，在初次运行Squid之前，或者修改了cache_dir设置后，都必须对Squid初始化。初始化的实质就是按配置项cache_dir ufs /var/spool/squid 4096 16 256的要求，在指定目录下自动建立指定数量的一级和二级子目录。

Squi化的命令格式为

```
squid –zX
```

其中：-X选项的作用是网管员可观察到初始化的过程。

初始化完成后，可以看到在/var/spool/squid/目录下建立了相应的两级子目录。

目录

本章要点

普通代理服务器的配置如图14-12所示。

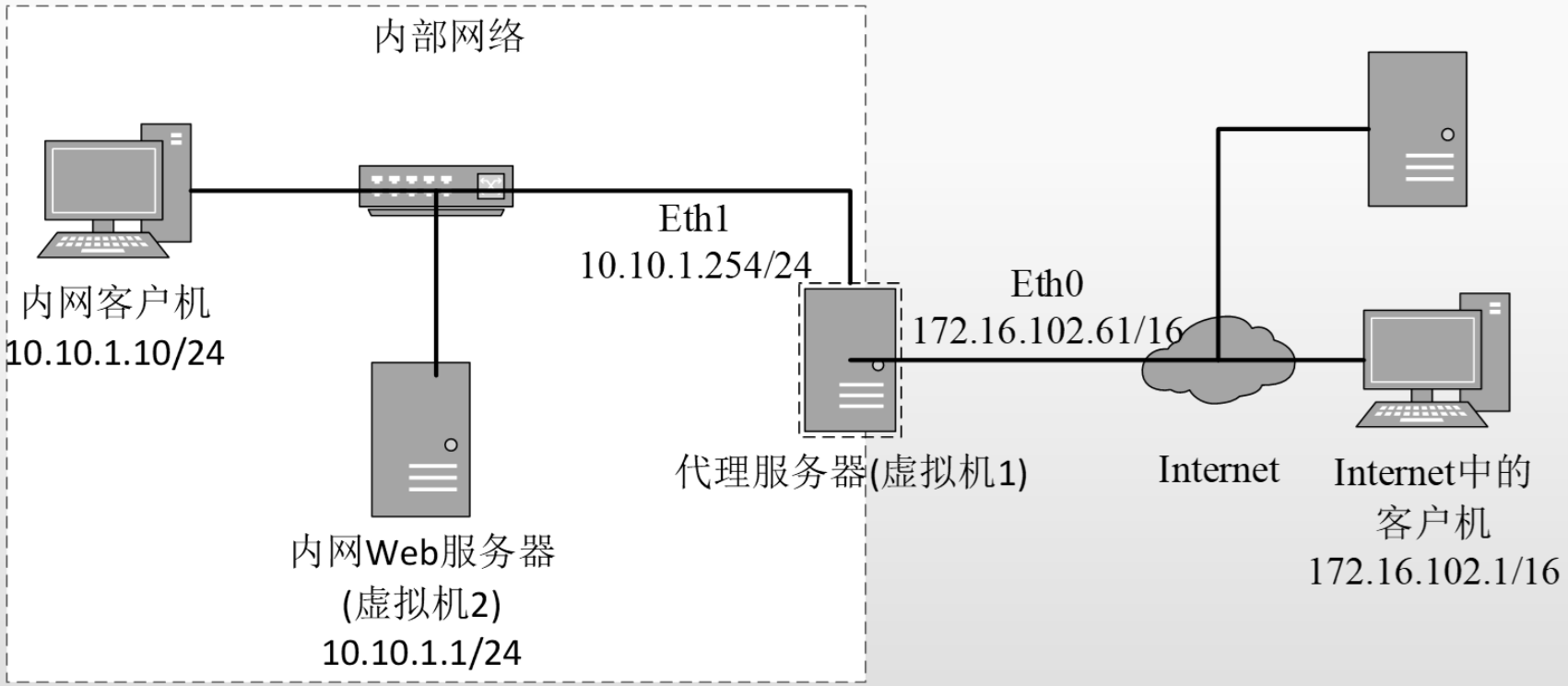


图14-12 普通代理服务器配置

步骤 1 ▶

安装Squid软件包。

步骤 2 ▶

使用setup命令配置网卡eth0和eth1的参数。

步骤 3 ▶

开启内核路由功能。

步骤 4 ▶

修改主配置文件/etc/squid/squid.conf。

步骤 5 ▶

初始化并启动Squid服务。

步骤 6 ▶

设置内网中客户机网卡及IE代理。

步骤 7 ▶

测试能否上网访问，如图14-13所示。

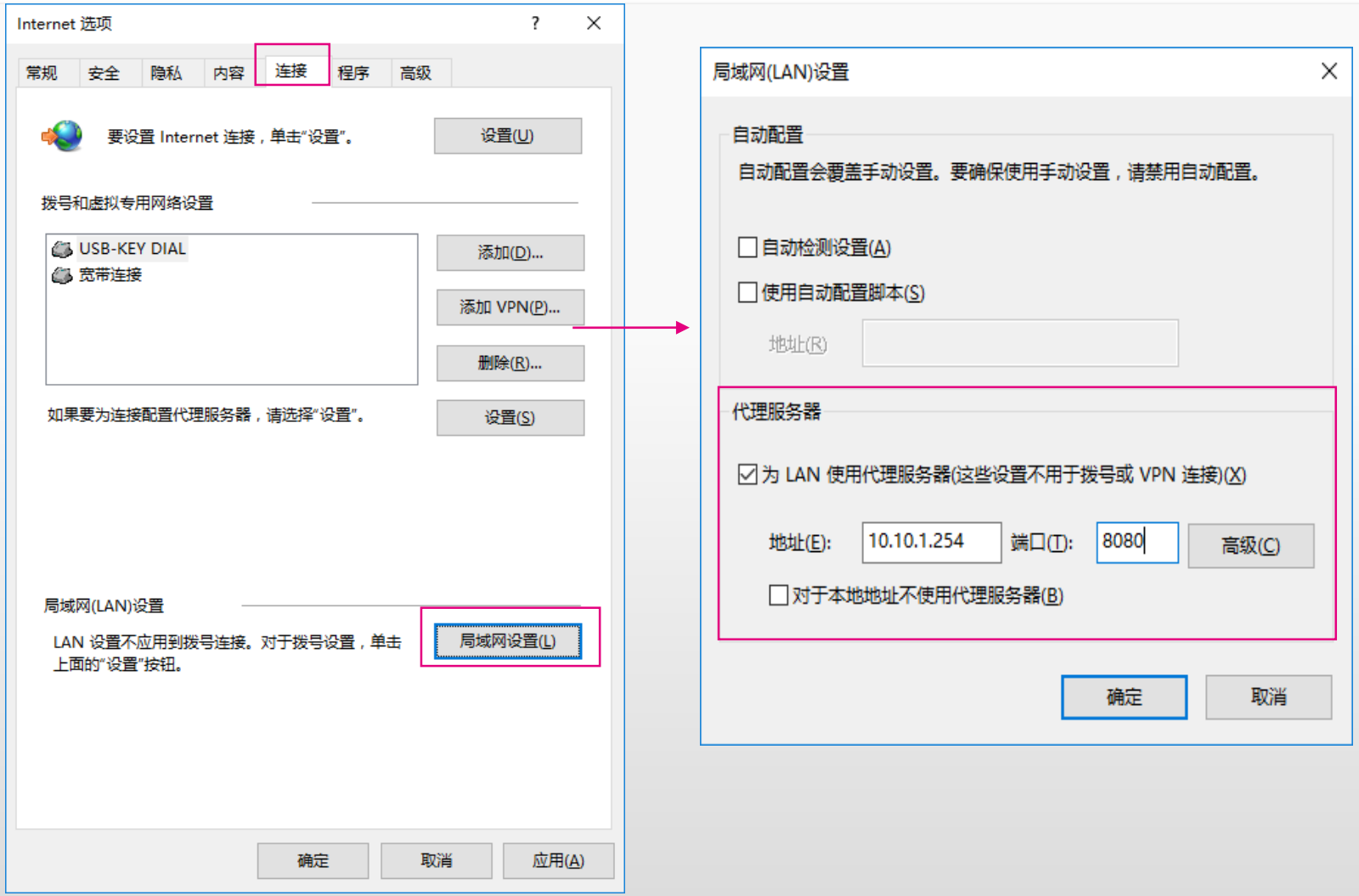


图14-13 测试能否上网

目录

本章要点

步骤 1 ▶

添加iptables的重定向规则。

步骤 2 ▶

修改seLinux设置。

步骤 3 ▶

修改squid.conf配置文件，添加对透明代理的支持（其他配置项略）。

步骤 4 ▶

检查squid.conf配置文件，当更改过配置文件后最好验证一下配置文件的语法正确性。

步骤 5 ▶

Squid服务初始化。

步骤 6 ▶

重新加载Squid服务配置。

步骤 7 ▶

测试。

客户端只要设置IP地址、子网掩码、默认网关及DNS就可以直接上网了（浏览器中不需要设置代理，若已设置则将其取消）。

目录

本章要点

步骤 1 ▶

修改squid.conf配置文件，添加对反向代理的支持（其他配置项略），并指定反向代理后台真实的Web服务器的位置等参数。

步骤 2 ▶

启动squid服务。

步骤 3 ▶

测试。

提示

在Internet中的客户端上，使用浏览器访问反向代理服务器的地址（如：http://172.16.102.61）。

① 实现该案例一个前提条件：若各Web服务器处于局域网内，要能够访问Internet。

② 需要指出的是，透明代理与反向代理不能同时应用，监听端口改为80是为了对应于标准Web端口，便于用户使用。

③ 配置Squid时最好制定内部DNS，或者修改/etc/hosts文件，否则Squid可能会回环访问其自身而出现问题。


谢谢观看