第3讲 网络防火墙开发与实验

姚立红
yaolh@sjtu.edu.cn

主要内容

1、网络防火墙开发解析

2、Linux的netfilter机制

3、内核模块包过滤防火墙原型

4、内核模块包过滤防火墙原型的扩展开发

5、应用层包过滤防火墙原型

6、应用层包过滤防火墙原型的扩展开发

7、应用代理防火墙原型

8、应用代理防火墙原型的扩展开发

9、透明代理防火墙原型

10、透明代理防火墙原型的扩展开发

1、网络防火墙开发解析

基本概念

- **抽象**：防止网络攻击
- **功能**：按一定的策略管理网络访问，即进行网络访问控制、审计及告警等。

逻辑结构

- **访问决策**：依据访问控制规则，做出当前网络访问是正当的、应该放行的，或不正当的、需阻断的判决。
- **访问实施**：对行程的访问控制判决，对一特定的网络访问进行控制，即放行或阻断该网络访问。

（1）包过滤防火墙

- **包过滤防火墙的安放点**：通常嵌入在连接内外网的路由器（或网关）上实现 。
- **包过滤防火墙的工作原理**：
  - 包过滤操作是在IP层实现。
  - 根据数据包的源IP地址、目标IP地址、协议类型（TCP、UDP、ICMP等）、源端口、目的端口、ICMP消息类型等报头信息及数据包传输方向等信息来，判断是否允许数据包通过。

包过滤防火墙的特点

- **接入方便**：包过滤防火墙工作在IP层，与应用层不相关，用户不需要改变客户端的任何应用程序或设置，也无需对内部网络用户进行相关使用培训，因而很容易接入到现存的网络环境中。
- **速度快**：包过滤防火墙工作在IP层，至多分析所对应的传输层协议（即获得端口等相关的属性信息），协议处理比较简单，所以处理包的速度比应用代理防火墙快。
- **实现简单**：包过滤防火墙实现相对简单，甚至可以集成到原有的路由器中。目前很多网络路由器都有IP包过滤功能，它们在逻辑上可以认为是包过滤防火墙。

（2）应用代理防火墙

- **应用代理防火墙的原理**：应用代理防火墙采取的是一种代理机制，它可以为每一种应用服务建立一个专门的代理，所以内外网之间的通信不是直接的，都需先经过应用代理防火墙的审核，审核通过后再由应用代理防火墙代为连接。

——不给内、外网的计算机任何直接会话的机会，从而避免了外部攻击者入侵内部网络。

应用代理防火墙的优缺点

- **优点**：实现基于网络会话的连接控制并产生相应的日志记录——安全性好。
- **缺点**
  - 效率低
  - 需要客户端设置

（3）透明代理防火墙

透明代理防火墙的目标

- 通过一定的技术手段，实现包过滤防火墙和应用代理型防火墙二者的优点。
- 既能够像应用代理防火墙一样，在应用层实现基于网络会话的连接控制并产生相应的日志记录，又能够像包过滤防火墙一样对局域网内部的网络用户透明，无需对客户端软件（如WEB或FTP的客户端软件）进行设置就能完成内外部网络之间的通信。

（4）应用层包过滤防火墙

工作原理

- 对绝大部分习惯了应用程序开发的程序者而言，Linux内核模块开发具有很大的难度 。
- Netfilter框架提供了队列功能（即IPqueue功能），它可以将Netfilter所截获的IP数据包不经过传输层（TCP、UDP等），而通过Netfilter通道（即IPqueue）以队列方式直接传递到应用层 。
- 借助于Netfilter机制支持的队列功能，可以在应用层实现对IP数据包的过滤和控制，基于这种方式实现的包过滤防火墙就是常说的应用层包过滤防火墙。

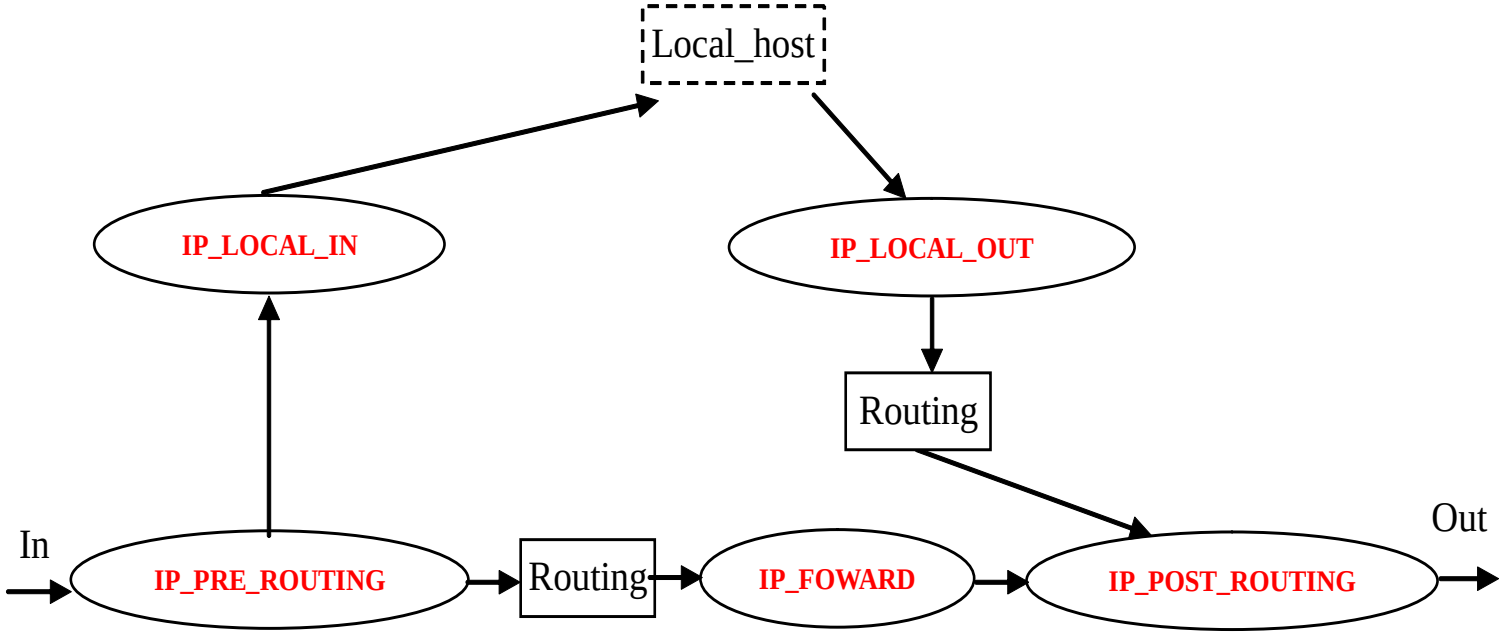
2、 Linux的Netfilter机制

Netfilter的基本概念

- Netfilter机制的核心是一个开放式的IP数据包处理框架，该框架对外提供了操纵和处理IP数据的统一接口，编程人员可以利用该接口实现对IP包的控制以及其它新的处理方式。
- 一方面Linux系统自身借助该机制实现一些常见的IP包处理方式，包括重新实现了其内核包过滤防火墙（称之为Linux内置防火墙），及其它相关的处理功能，如网络地址转换NAT等。
- 另一方面，第三方的软件开发者可以基于Netfilter提供的IP报文处理接口，开发相应的网络工具，包括网络防火墙、NAT、网络审计等。

Netfilter的核心思想

- 在网络IP协议层IP数据包的处理流程中，总结出几个关键点（即钩子点），这些关键点提供了多种可能的IP数据包处理方式和开放接口。
- 安全管理员不但可以配置Netfilter以不同的方式处理IP数据包，也可利用Netfilter所提供的开放接口在IP数据包的协议处理流程中实现新的IP包处理方式。



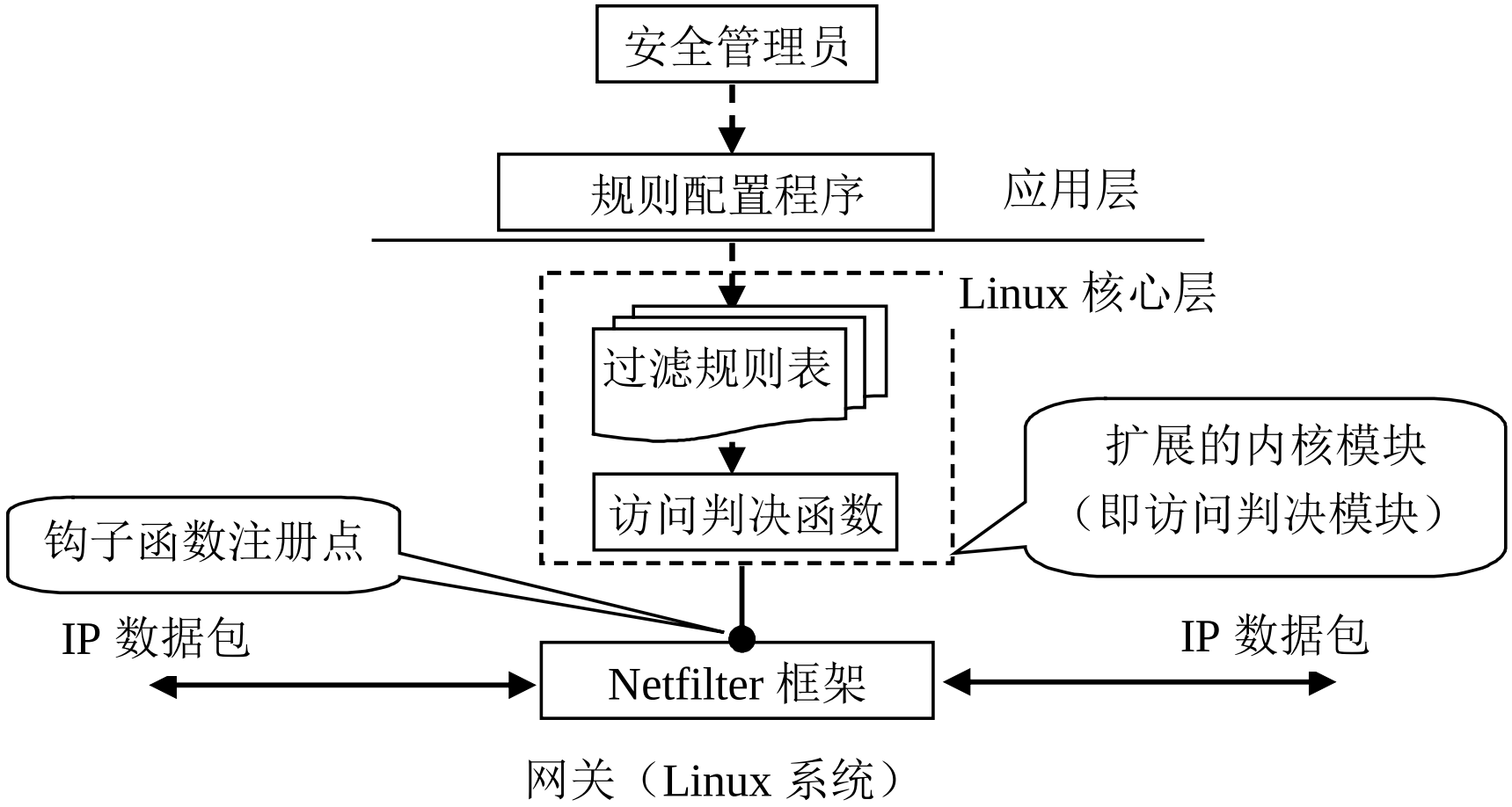
IP数据包的处理流程中的五个钩子点

开放处理方式

- 意味着可在Netfilter机制基础上自己开发新的报文处理方式，Netfilter将IP报文的处理权交给新开发的报文处理方式。
- 开放处理方式包括两种具体方式：
  - **钩子函数方式**：Netfilter机制为每种网络协议(IPv4、IPv6等)定义一套钩子，在数据报流过协议栈的某钩子点，挂接在该钩子上的钩子函数将会被调用。
  - **队列输出方式**： Netfilter提供队列输出功能，在这些关键点将所经过的IP数据包通过一定的方式直接交给应用层，程序员可以在应用层开发应用软件对这些数据包进行完全自主的处理，如丢弃、修改等。

3、内核模块包过滤防火墙原型

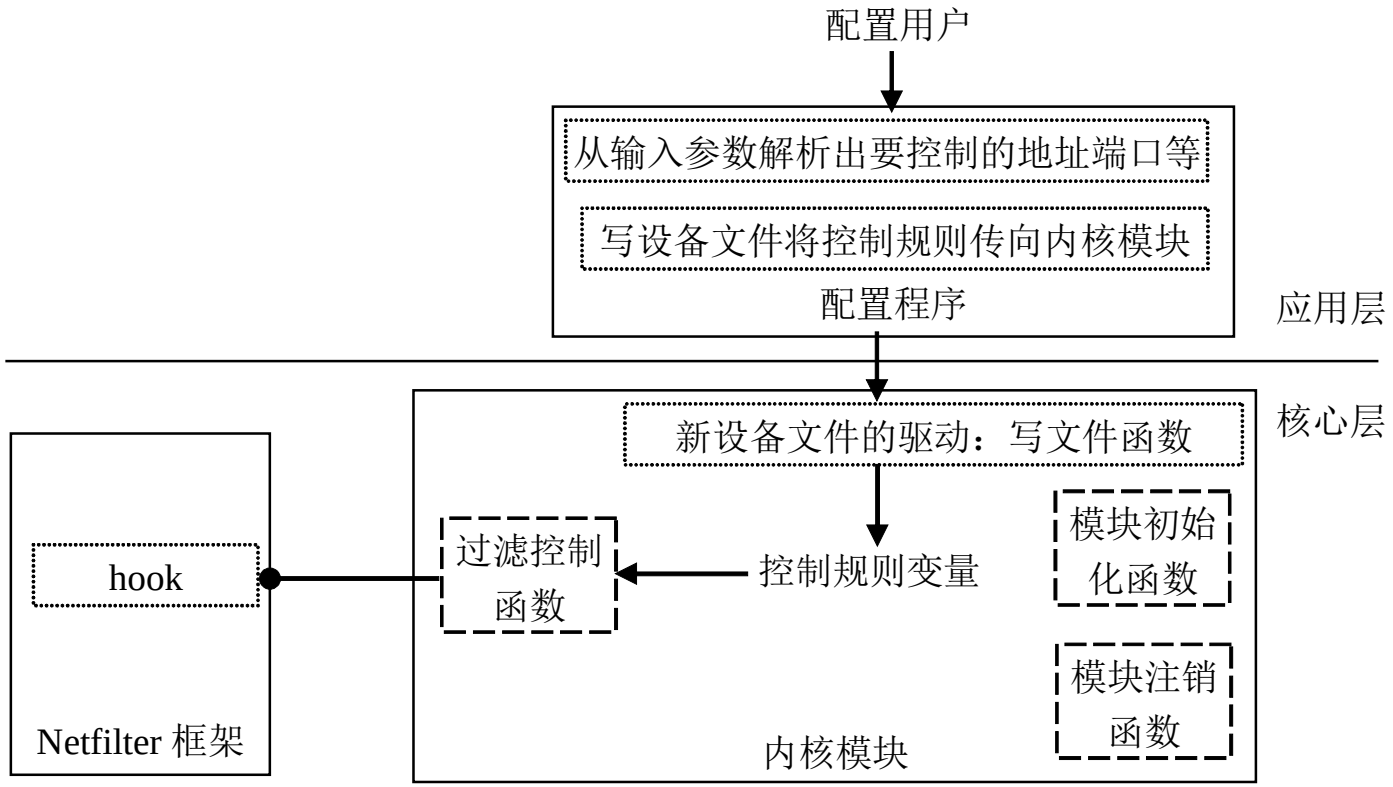
运行原理



实现结构

原型系统分为两个部分独立实现

- **配置程序**：运行在应用层，用来设置启用哪一条包过滤策略和配置相应的参数，包括要控制的IP地址和端口等；
- **Linux内核模块**：运行在内核层，完成包过滤防火墙的功能，该模块借助注册Netfilter 钩子函数的方式来实现对数据包的过滤和控制。



运行方式

- 插入内核模块
- 通过命令行参数，输入要被丢弃的数据包的协议类型，源、目的IP地址以及源、目的端口号。

./configure –p protocol –x source_ip –y source_port –m dst_ip –n dst_port

各选项的具体含义如下：

- -p protocol 指明要控制的协议（或网络应用）类型，具体为 tcp、udp、ping三种之一；
- –x source_ip 指明要控制报文的源IP地址；
- –y dst_ip 指明要控制报文的目标IP地址；
- –m source_port 指明要控制报文的源端口地址；
- –n dst_port 指明要控制报文的目标端口；

——若某个选项省略则取其默认值为0，0表示对任意值相匹配，即表示控制的报文覆盖该选项的所有值域。

4、内核包过滤防火墙扩展开发

（1）内核模块包过滤防火墙的控制功能扩展

（2）基于Netfilter内核模块的网络加密通信系统

（1）内核模块包过滤防火墙的控制功能扩展

背景：原型系统的控制功能简单

- 控制规则简单，只是依据通信双方的IP地址和端口进行检查和控制，实际上包过滤防火墙还可以依据其它要素进行控制。比如，基于时间段进行控制，比如在休息日时间不准从外网访问内网等。
- 以几个简单变量存储相应的控制信息，相当于只能支持一条包过滤规则。一个实用的包过滤防火墙需要同时支持多条包过滤规则。

开发目标：

- **检查和控制要素的扩展**。除实现基于IP地址、端口的检查和控制外，还能使基于时间段，网络接口，ICMP报文的子类型等进行报文的检查和控制。
- **多包过滤规则的扩展**。以表的形式存储包过滤规则，能够支持用户配置多条包过滤规则，使内核模块防火墙能够同时按多条包过滤规则进行报文检查和过滤控制。
- **友好的包过滤规则的配置和管理界面**。支持包过滤的规则导入、导出，添加、编辑、删除等基本功能。

（2）基于Netfilter内核模块的网络加密通信系统

技术基础：

- IP报文在处理流程中经过Netfilter框架某钩子点时，Netfilter不仅会调用所注册的钩子函数，还会将该IP报文的具体内容以参数的形式交给钩子函数处理。
- 在钩子函数中，可以按照一定的目的对报文内容进行变换（或修改），然后将经过内容变换的IP报文再返回给Netfilter框架，Netfilter框架会像处理原始的IP报文一样继续处理被钩子函数变换过内容的IP报文。

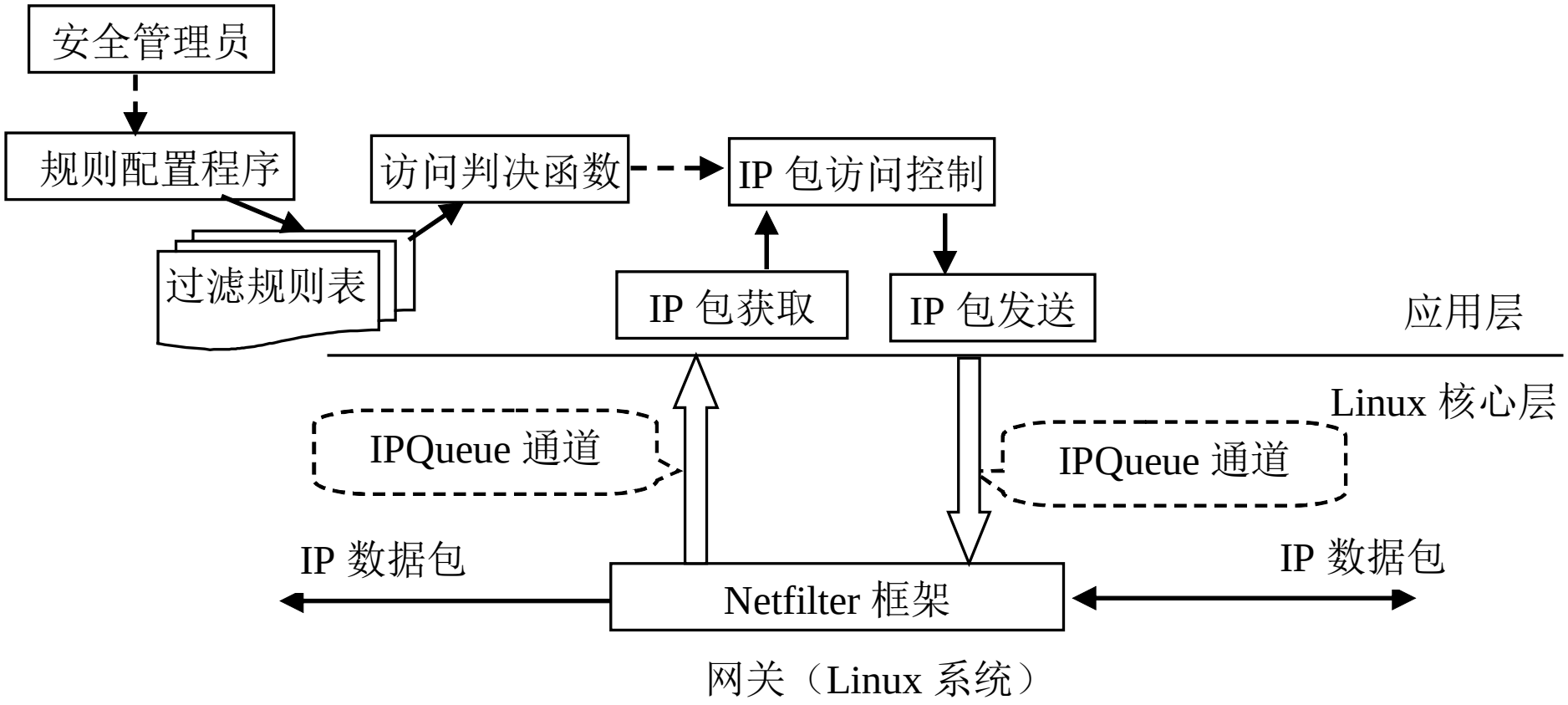
实现思路：

- **在发送端的网关或端系统**：在Netfilter框架中注册相应的钩子函数，在钩子函数中，对IP报文的内容进行加密处理，这样由本主机或本网关发出的IP报文，其内容都是密文。
- **在接收端的网关或端系统**：在Netfilter框架中注册相应的钩子函数，在钩子函数实现对IP报文内容的解密。

——实现应用层透明的网络加密数据通信，无论是客户机还是服务器都感受不到该加解密过程的存在，而在中间的网络链路上传输的数据是经过加密处理的，安全性得到很大的提高。

5、应用层包过滤防火墙原型

运行原理



实现结构

- 应用层包过滤防火墙不实现单独的包过滤规则配置程序。
- 包过滤规则配置和基于包过滤规则进行IP报文过滤实现在一个程序中 。

运行方式

- **配置netfilter队列机制**

  iptables –A OUTPUT –j QUEUE

- 通过命令行参数，输入要被丢弃的数据包的协议类型，源、目的IP地址以及源、目的端口号

./queue_fw –p protocol –x source_ip –y source_port –m dst_ip –n dst_port

各选项的具体含义如下：

- -p protocol 指明要控制的协议（或网络应用）类型，具体为 tcp、udp、ping三种之一；
- –x source_ip 指明要控制报文的源IP地址；
- –y dst_ip 指明要控制报文的目标IP地址；
- –m source_port 指明要控制报文的源端口地址；
- –n dst_port 指明要控制报文的目标端口；

——若某个选项省略则取其默认值为0，0表示对任意值相匹配，即表示控制的报文覆盖该选项的所有值域。

6、应用层包过滤防火墙扩展开发

（1）应用层包过滤防火墙的控制功能扩展

（2）基于队列机制的网络加密通信系统

（1）应用层包过滤防火墙的控制功能扩展

背景：原型系统的控制功能简单

- **控制规则简单**，只是依据通信双方的IP地址和端口进行检查和控制，实际上包过滤防火墙还可以依据其它要素进行控制。如基于时间段进行控制：在休息日时间不准从外网访问内网等。
- 以几个简单变量存储相应的控制信息，相当于只能支持一条包过滤规则。一个实用的包过滤防火墙需要同时支持多条包过滤规则。

开发目标：

- **检查和控制要素的扩展**。除实现基于IP地址、端口的检查和控制外，还能使基于时间段，网络接口，ICMP报文的子类型等进行报文的检查和控制。
- **多包过滤规则的扩展**。以表的形式存储包过滤规则，能够支持用户配置多条包过滤规则，使内核模块防火墙能够同时按多条包过滤规则进行报文检查和过滤控制。
- **友好的包过滤规则的配置和管理界面**。支持包过滤的规则导入、导出，添加、编辑、删除等基本功能。

（2）基于队列机制的网络加密通信系统

技术基础

- Netfilter利用队列机制将IP层的报文发往应用层，同时可以将接收应用层发回的报文继续处理，而不仅是向应用询问每个IP报文的处理方式（丢弃和放行等）。

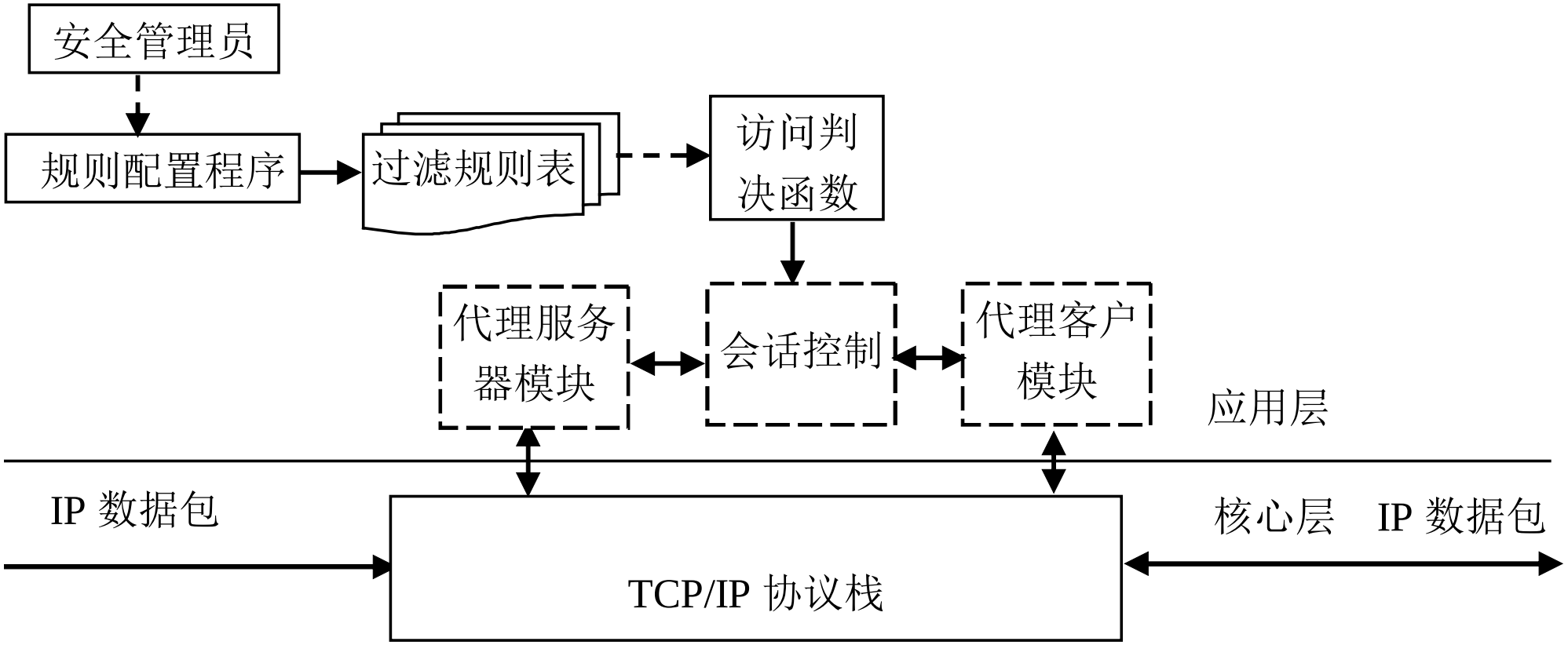
实现思路

- **发送端的网关或端系统**：对IP报文的内容进行加密处理，这样由本机或本网关发出的IP报文其内容都是密文。
- **接收端的网关或端系统**：对IP报文的内容进行解密处理。

——实现应用层透明的网络加密数据通信

7、应用代理防火墙原型

运行原理



实现结构

- 应用代理防火墙不实现单独的控制规则配置程序，控制规则配置和基于控制规则进行连接控制实现在一个程序中 。
- 原型系统采用多线程的方式实现，主线程和若干子线程，主线程每接收到一个代理请求时，都会创建一个子线程，将该子线程负责处理该代理请求。

主线程的功能

主线程的具体功能依次包括：

- 解析出命令行参数，获得代理防火墙的服务端口。
- 创建套接字接口，然后监听（listen）相应的服务端口。
- 循环执行：接收（accept）来自客户端的代理请求，并进行客户端的IP地址检查。若通过检查，创建一个子线程，将该请求交给该子线程处理。

子线程的功能

子线程的具体功能依次包括：

- 以参数的形式从主线程中获得代理请求对应的套接字（下称客户套接字）。
- 从该套接字中读取（read）客户端发来的代理请求的具体内容。
- 从该内容中，解析出客户端所要请求的http服务器（以域名形式表示的）。
- 进行服务器域名检查，如果检查不通过终止该子线程。若检查通过，则继续后面的处理。
- 与远程的http服务器（即客户端请求的http服务器）建立socket连接，并将客户端发来的请求内容，发送到该socket接口（下称服务套接字）。
- 一直从服务socket接口读取远程http服务的响应，直至对方关闭该连接，并将该响应转发至客户socket接口。

运行方式

- 防火墙原型工具以命令行程序的形式运行，可以接收一个参数，即指定防火墙的服务端口。
- 采用标准的unix参数指定方式，运行方式如下：

proxy -p port

➢proxy：为通过源代码编译出来的代理防火墙（即可执行程序名）。

➢port：表明防火墙的服务端口。

8、应用代理防火墙扩展开发

（1）应用代理防火墙的控制功能扩展

（2）应用代理防火墙的FTP支持扩展

（1）应用代理防火墙的控制功能扩展

背景：原型系统的控制功能简单

- **控制规则简单**，只能依据客户端和服务器控制是否允许使用代理服务，不能依据其它要素进行控制。
- 以几个简单变量存储相应的控制信息，相当于**只能支持一条包过滤规则**。一个实用的包过滤防火墙需要同时支持多条包过滤规则。
- **无法实现控制规则的动态设置**。

开发目标：

- **访问控制规则要素的扩展**。支持基于http协议信息的访问控制。具体来讲主要两类：基于用户名和口令的身份认证，基于http消息内容的控制。
- **多控制规则的扩展**。
- **控制信息配置、保存及动态更新扩展，**具体内容包括：实现一个友好的控制信息配置界面（或配置程序），管理员可通过该配置界面实现对防火墙的访问控制信息的配置。

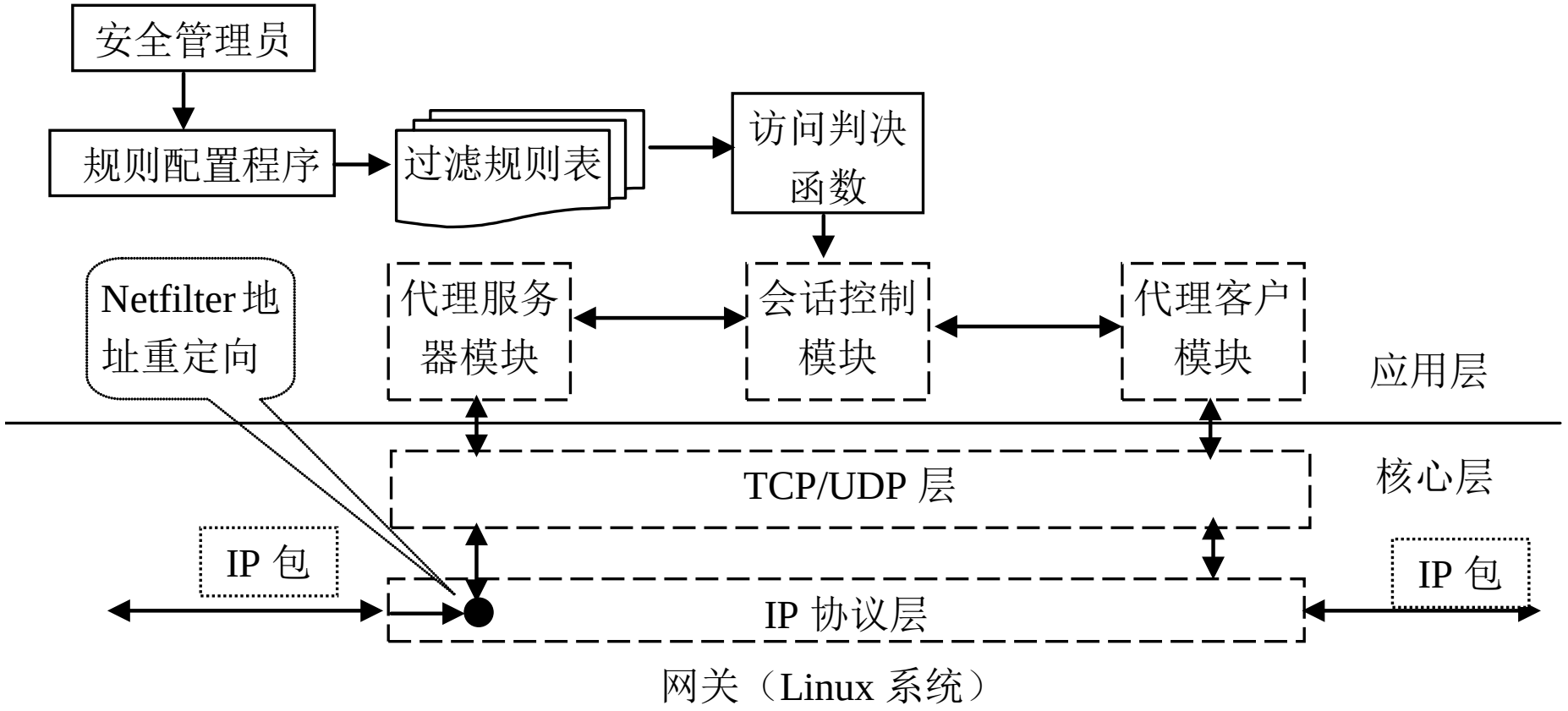
（2）应用代理防火墙的FTP支持扩展

- **背景基础**：代理防火墙原型只有对http应用的代理功能，对其它的网络应用（如ftp，mail等）不能提供代理支持。
- **支持ftp协议代理防火墙的主要任务**：分析ftp服务器和客户端交互的协议数据，并根据分析出的各个要素进行相应的检查和控制。

  常见的安全检查和控制要素包括：身份客户机和服务器的位置（IP地址等），认证信息（用户名和口令等）、传输文件名（或目录名），传输的文件类型、文件大小等。

9、透明代理防火墙原型

运行原理



实现结构

- 透明代理防火墙不实现单独的控制规则配置程序，控制规则配置和基于控制规则进行连接控制实现在一个程序中 。
- 原型系统采用多线程的方式实现：一个主线程和若干子线程。主线程每接收到一个代理请求时，都会创建一个子线程，将该子线程负责处理该代理请求。

- **主线程的功能**

主线程的具体功能依次包括：

- 解析出命令行参数，获得代理防火墙的服务端口。
- 创建套接字接口，然后监听（listen）相应的服务端口。
- 循环执行：接收（accept）来自客户端的代理请求，并进行客户端的IP地址检查。若通过检查，创建一个子线程，将该请求交给该子线程处理。

- **子线程的功能**

子线程的具体功能依次包括：

- 以参数的形式从主线程中获得代理请求对应的套接字。
- 调用getsockopt函数，获得客户套接字重定向前的目标地址和端口，即服务器的目标地址和端口。
- 进行服务器检查，如果检查不通过终止该子线程。若检查通过，则继续后面的处理。
- 与远程的http服务器建立socket连接，并将客户端发来的请求内容，**发送**到该socket接口。
- 一直从服务socket接口读取远程http服务的响应，直至对方关闭该连接，并将该响应转发至客户socket接口。

运行方式

- 设置netfilter的重定向

iptables –t nat –A PREROUTING –p tcp –j REDIRECT –to –ports 8888

- 防火墙原型工具以命令行的形式运行，可以接收一个参数，指定防火墙的服务端口。

该工具采用标准的unix参数指定方式，运行方式如下：

tc-proxy -p port

- tc-proxy：为通过源代码编译出来的代理防火墙（即可执行程序名）。
- port：重定向端口 8888

10、透明代理防火墙扩展开发

（1）透明代理防火墙的多规则支持和动态配置扩展

（2）透明代理防火墙的HTTP协议解析与控制扩展

（3）透明代理防火墙的FTP协议解析与控制扩展

（1）透明代理防火墙的多规则支持和动态配置扩展

背景：原型系统的控制功能简单

- **控制规则简单**，只能依据客户端和服务器控制是否允许使用代理服务，不能依据其它要素进行控制。
- 以几个简单变量存储相应的控制信息，相当于**只能支持一条包过滤规则**。一个实用的包过滤防火墙需要同时支持多条包过滤规则。
- **无法实现控制规则的动态设置**。

开发目标：

- **多控制规则的扩展。**
- **控制信息配置、保存及动态更新扩展：**
  - 实现一个友好的控制信息配置界面（或配置程序），管理员可通过该配置界面实现对防火墙的访问控制信息的配置。

（2）透明代理防火墙的HTTP协议解析与控制扩展

- 该扩展涉及到对http消息内容分析和控制。
- HTTP消息包括：浏览器向服务器的请求消息和服务器向浏览器的响应消息。
  - **请求消息：**请求方法常用的有GET、HEAD、POST，可以依据不同的请求方法进行控制，也可以实现对URL的过滤和控制。
  - **响应消息：**响应消息中对网络安全影响较大的要素是响应的实体类型，可以据此进行访问控制，如禁止下载applet、禁止下载word文件等。

（3）透明代理防火墙的FTP协议解析与控制扩展

- 支持ftp协议代理防火墙的主要任务：分析ftp服务器和客户端交互的协议数据，并根据分析出的各个要素进行相应的检查和控制。

常见的安全检查和控制要素包括：

- ✓ 身份客户机和服务器的位置（IP地址等）
- ✓ 认证信息（用户名和口令等）
- ✓ 传输文件名（或目录名），文件类型、文件大小等 。