网络安全——

防火墙

北京邮电大学

郑康锋
kfzheng@bupt.edu.cn

网络安全——

访问控制

什么是访问控制?

- **访问控制（Access Control）指系统对用户身份及其所属的预先定义的策略组限制其使用数据资源能力的手段。**通常用于系统管理员控制用户对服务器、目录、文件等网络资源的访问。

- 访问控制是系统保密性、完整性、可用性和合法使用性的重要基础，是网络安全防范和资源保护的关键策略之一，也是主体依据某些控制策略或权限对客体本身或其资源进行的不同授权访问。

- 访问控制的主要目的是限制访问主体对客体的访问，从而保障数据资源在合法范围内得以有效使用和管理。为了达到上述目的，**访问控制需要完成两个任务：识别和确认访问系统的用户、决定该用户可以对某一系统资源进行何种类型的访问。**

什么是访问控制?

- 访问控制包括三个要素：主体、客体和控制策略。

  - （1）主体S（Subject）。是指提出访问资源具体请求。是某一操作动作的发起者，但不一定是动作的执行者，可能是某一用户，也可以是用户启动的进程、服务和设备等。

  - （2）客体O（Object）。是指被访问资源的实体。所有可以被操作的信息、资源、对象都可以是客体。客体可以是信息、文件、记录等集合体，也可以是网络上硬件设施、无限通信中的终端，甚至可以包含另外一个客体。

  - （3）控制策略A（Attribution）。是主体对客体的相关访问规则集合，即属性集合。访问策略体现了一种授权行为，也是客体对主体某些操作行为的默认。

访问控制策略

- 安全策略的实施原则：安全策略的制定实施也是围绕主体、客体和安全控制规则集三者之间的关系展开的。

（1） **最小特权原则：最小特权原则是指主体执行操作时，按照主体所需权利的最小化原则分配给主体权力。**最小特权原则的优点是最大限度地限制了主体实施授权行为，可以避免来自突发事件、错误和未授权用主体的危险。

（2） **最小泄漏原则：最小泄漏原则是指主体执行任务时，按照主体所需要知道的信息最小化的原则分配给主体权力。**

（3） **多级安全策略：多级安全策略是指主体和客体间的数据流向和权限控制按照安全级别**的绝密（TS）、秘密（S）、机密（C）、限制（RS）和无级别（U）五级**来划分**。多级安全策略的优点是避免敏感信息的扩散。具有安全级别的信息资源，只有安全级别比他高的主体才能够访问。

访问控制——

访问控制模型

自主访问控制模型DAC

- **自主访问控制模型**（DAC Model，Discretionary Access Control Model）是根据自主访问控制策略建立的一种模型，**允许合法用户以用户或用户组的身份访问策略规定的客体**，同时阻止非授权用户访问客体，某些**用户还可以自主地把自己所拥有的客体的访问权限授予其它用户**。
- **自主访问控制又称为任意访问控制**。LINUX，UNIX、WindowsNT或是SERVER版本的操作系统都提供自主访问控制的功能。
- 在实现上，首先要对用户的身份进行鉴别，然后就可以按照访问控制列表所赋予用户的权限允许和限制用户使用客体的资源。主体控制权限的修改通常由特权用户或是特权用户（管理员）组实现。

自主访问控制模型DAC

- 任意访问控制对用户提供的这种灵活的数据访问方式，使得DAC广泛应用在商业和工业环境中；
- 由于用户可以任意传递权限，那么，没有访问文件File1权限的用户A就能够从有访问权限的用户B那里得到访问权限或是直接获得文件File1；因此，DAC模型提供的安全防护还是相对比较低的，不能给系统提供充分的数据保护。
- 自主访问控制模型的特点是授权的实施主体（1、可以授权的主体；2、管理授权的客体；3、授权组）自主负责赋予和回收其他主体对客体资源的访问权限。
- DAC模型一般采用访问控制矩阵和访问控制列表来存放不同主体的访问控制信息，从而达到对主体访问权限的限制目的。

强制访问控制模型（MAC）

- 强制访问控制模型（MAC Model：Mandatory Access Control Model）一种多级访问控制策略。
- 系统对访问主体和受控对象实行强制访问控制，系统事先给访问主体和受控对象分配不同的安全级别属性，实施访问控制时，系统先对访问主体和受控对象的安全级别属性进行比较，再决定访问主体能否访问该受控对象。
- MAC对访问主体和受控对象标识两个安全标记：
  - 一个是具有偏序关系的安全等级标记；
  - 另一个是非等级分类标记。
- 当主体s的安全类别为TS，而客体o的安全类别为S时，用偏序关系可以表述为SC(s)≥SC(o)。

强制访问控制模型（MAC）

- **考虑到偏序关系，主体对客体的访问主要有四种方式：**

（1）**向下读**（rd，read down）：主体安全级别高于客体信息资源的安全级别时允许查阅的读操作；

（2）**向上读**（ru，read up）：主体安全级别低于客体信息资源的安全级别时允许的读操作；

（3）**向下写**（wd，write down）：主体安全级别高于客体信息资源的安全级别时允许执行的动作或是写操作；

（4）**向上写**（wu，write up）：主体安全级别低于客体信息资源的安全级别时允许执行的动作或是写操作。

强制访问控制模型（MAC）

- MAC模型中的几种主要模型：Lattice模型，Bell-LaPadula模型（BLP Model）和Biba模型（Biba Model）。
- Lattice模型
  - 在Lattices模型中，每个资源和用户都服从于一个安全类别。这些安全类别我们称为安全级别，也就是我们在本章开始所描述的五个安全级别，TS，S，C，R，U。
  - 在整个安全模型中，信息资源对应一个安全类别，用户所对应的安全级别必须比可以使用的客体资源高才能进行访问。
  - Lattices模型是实现安全分级的系统，这种方案非常适用于需要对信息资源进行明显分类的系统。

强制访问控制模型（MAC）

- Bell-LaPadula模型

  BLP[Bell and LaPadula，1976]模型是典型的信息保密性多级安全模型，主要应用于军事系统。

- 无上读、无下写

- Bell-LaPadula模型可以有效防止低级用户和进程访问安全级别比他们高的信息资源。此外，安全级别高的用户和进程也不能向比他安全级别低的用户和进程写入数据。

- BLP模型的安全策略包括强制访问控制和自主访问控制两部分：强制访问控制中的安全特性要求对给定安全级别的主体，仅被允许对同一安全级别和较低安全级别上的客体进行"读"；对给定安全级别上的主体，仅被允许向相同安全级别或较高安全级别上的客体进行"写"；任意访问控制允许用户自行定义是否让个人或组织存取数据。

- **BLP模型的出发点是维护系统的保密性，有效地防止信息泄露，忽略了完整性指标，使非法、越权篡改成为可能。**

强制访问控制模型（MAC）

- **Biba模型**

  Biba模型[Biba,1977]在研究BLP模型的特性时发现，BLP模型只解决了信息的保密问题，其在完整性定义存在方面有一定缺陷。

- 禁止向上写，没有向下读

- Biba模型模仿BLP模型的信息保密性级别，定义了信息完整性级别，在信息流向的定义方面不允许从级别低的进程到级别高的进程，也就是说用户只能向比自己安全级别低的客体写入信息，从而防止非法用户创建安全级别高的客体信息，避免越权、篡改等行为的产生。

- Biba模型是和BLP模型相对立的模型，Biba模型改正了被BLP模型所忽略的信息完整性问题，但在一定程度上却忽视了保密性。

基于角色的访问控制模型

- 基于角色的访问控制模型（RBAC Model，Role-based Access Model）的基本思想是将访问许可权分配给一定的角色，用户通过饰演不同的角色获得角色所拥有的访问许可权。

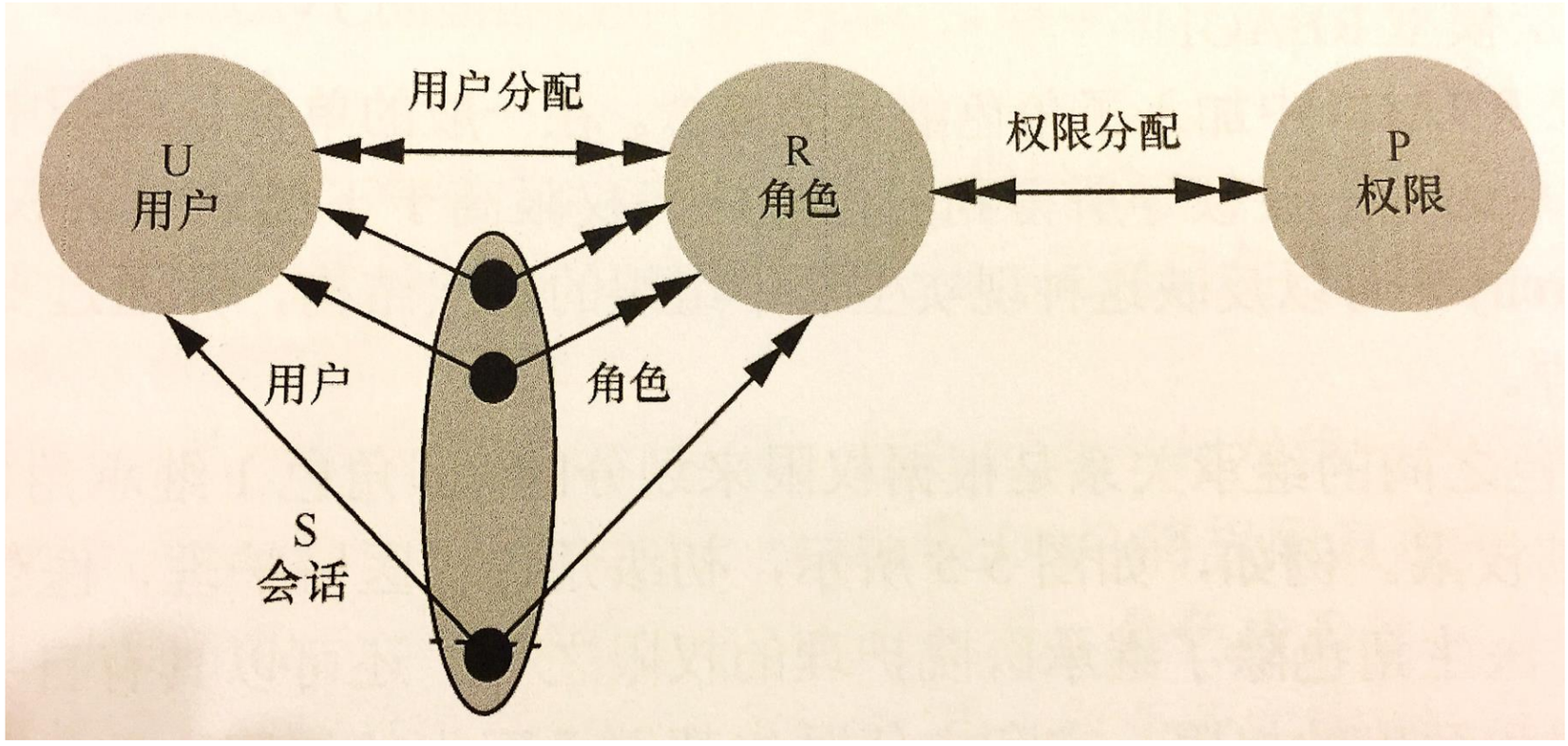
- RBAC从控制主体的角度出发，根据管理中相对稳定的职权和责任来划分角色，将访问权限与角色相联系，这点与传统的MAC和DAC将权限直接授予用户的方式不同；

- 通过给用户分配合适的角色，让用户与访问权限相联系。角色成为访问控制中访问主体和受控对象之间的一座桥梁。

基于角色的访问控制模型

- 相比较而言， RBAC是实施面向企业的安全策略的一种有效的访问控制方式，其具有灵活性、方便性和安全性的特点，目前在大型数据库系统的权限管理中得到普遍应用。
- 角色由系统管理员定义，角色成员的增减也只能由系统管理员来执行，即只有系统管理员有权定义和分配角色。用户与客体无直接联系，他只有通过角色才享有该角色所对应的权限，从而访问相应的客体。
- 用户不能自主地将访问权限授给别的用户，这是RBAC与DAC的根本区别所在。
- RBAC与MAC的区别在于：MAC是基于多级安全需求的，而RBAC则不是。

基本模型RBAC0



- 表达了RBAC系统的最小需求，由User,Role,Session（一个用户和一组激活的角色）,Permission构成。
- 用户分配（User Assignment)，用户集合到角色集合的多对多映射。
- 权限分配（Permission Assignment）权限集合到角色集合的多对多映射。

角色分层模型RBAC1

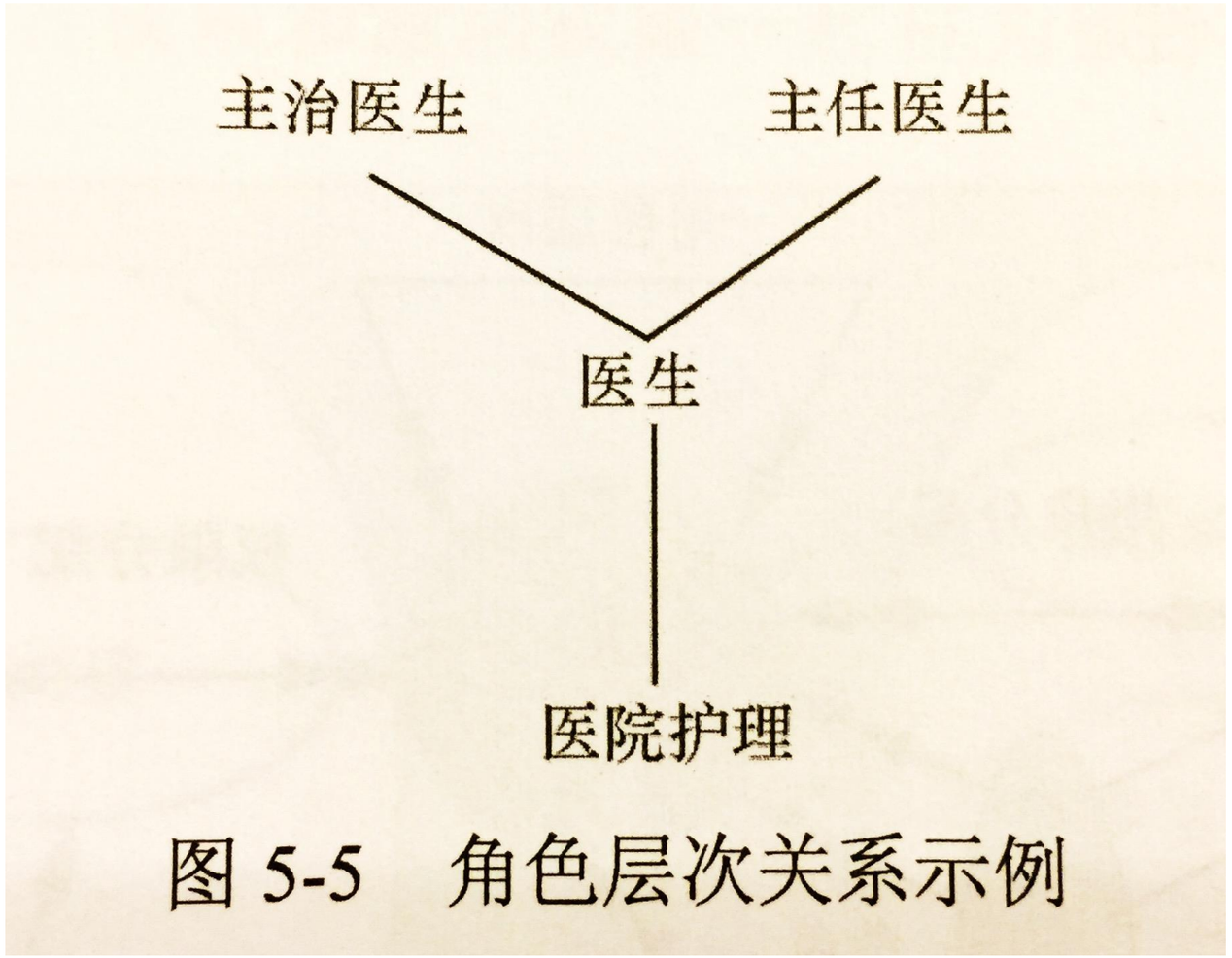


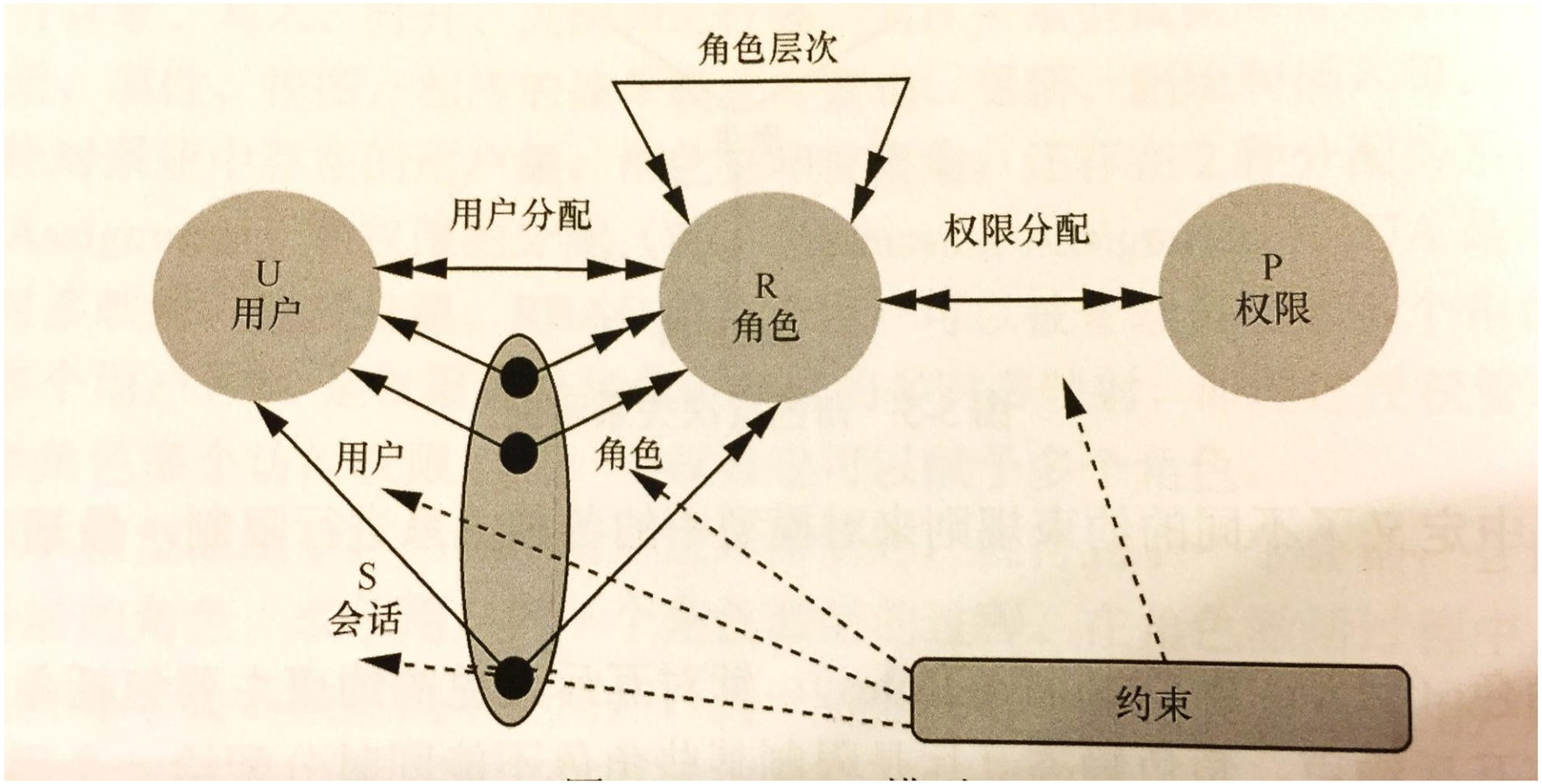
图 5-5　角色层次关系示例

- 在RBAC0基础上加入了角色继承的概念。
- 角色层次（RH，Role Hierarchy）反应层次结构，支持成员和权限继承以方便权限管理。

角色约束模型RBAC2

- 在RBAC0基础上引入了约束（Constraints）的概念。
- 约束是角色之间及角色与权限之间的一种限制关系。
- RBAC2定义了不同的约束规则来对各种关系进行限制：
  - 互斥角色：角色静态互斥是限制某些角色不能同时分配给一个用户，角色动态互斥是一个用户开始会话，不能同时激活某些角色。
  - 基数约束：可以限制一个角色可以分配的最大和最小用户数。
  - 先决角色：用户为获得某些高级角色必须先拥有低等级角色。
  - 会话约束：限制仅在特定会话中才允许激活某个角色，还可以限制一个用户在同一时间可以激活的会话数量。
  - 等级约束：限制角色的层次不能超过多少层。

统一模型RBAC3



- 在RBAC0基础上加入了角色继承和约束。
- 实际上是把RBAC1和RBAC2结合在了一起，既提供了角色的继承关系，又提供角色之间以及角色与权限之间的限制关系。

基于任务的访问控制模型（TBAC）

- 基于任务的访问控制模型（TBAC Model，Task-based Access Control Model）是从应用和企业层角度来解决安全问题，以面向任务的观点，从任务（活动）的角度来建立安全模型和实现安全机制，在任务处理的过程中提供动态实时的安全管理。
- 在TBAC中，对象的访问权限控制并不是静止不变的，而是随着执行任务的上下文环境发生变化。TBAC首要考虑的是在工作流的环境中对信息的保护问题，TBAC是一种上下文相关的访问控制模型。其次，TBAC不仅能对不同工作流实行不同的访问控制策略，而且还能对同一工作流的不同任务实例实行不同的访问控制策略。
- TBAC是基于任务的，是一种基于实例（instance-based）的访问控制模型。
- TBAC模型由工作流、授权结构体、受托人集、许可集四部分组成。

访问控制——

访问控制实现

访问控制表

- 访问控制表（ACLs：Access Control Lists）是以文件为中心建立的访问权限表，简记为ACLs。目前，大多数PC、服务器和主机都使用ACLs作为访问控制的实现机制。

- 访问控制表的优点在于实现简单，任何得到授权的主体都可以有一个访问表，例如授权用户A1的访问控制规则存储在文件File1中，A1的访问规则可以由A1下面的权限表ACLsA1来确定，权限表限定了用户UserA1的访问权限。

访问控制矩阵

- **访问控制矩阵（ACM：Access Control Matrix）是通过矩阵形式表示访问控制规则和授权用户权限的方法；**也就是说，对每个主体而言，都拥有对哪些客体的哪些访问权限；而对客体而言，又有哪些主体对他可以实施访问；将这种关连关系加以阐述，就形成了控制矩阵。其中，特权用户或特权用户组可以修改主体的访问控制权限。
- 访问控制矩阵的实现很易于理解，但是查找和实现起来有一定的难度，而且，如果用户和文件系统要管理的文件很多，那么控制矩阵将会成几何级数增长，这样对于增长的矩阵而言，会有大量的空余空间。

访问控制能力列表

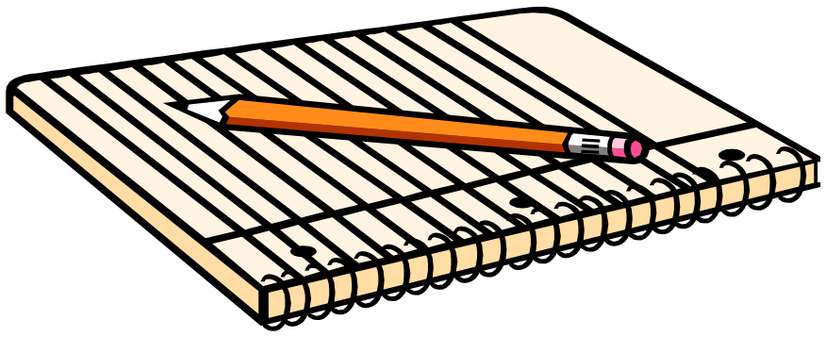
- 能力是访问控制中的一个重要概念，它是指请求访问的发起者所拥有的一个有效标签（ticket），它授权标签表明的持有者可以按照何种访问方式访问特定的客体。
- 访问控制能力表（ACCLs：Access Control Capabilitis Lists）是以用户为中心建立访问权限表。
- 例如，访问控制权限表ACCLsF1表明了授权用户UserA对文件File1的访问权限，UserAF表明了UserA对文件系统的访问控制规则集。因此，ACCLs的实现与ACLs正好相反。
- 定义能力的重要作用在于能力的特殊性，如果赋予哪个主体具有一种能力，事实上是说明了这个主体具有了一定对应的权限。能力的实现有两种方式，传递的和不可传递的。一些能力可以由主体传递给其他主体使用，另一些则不能。能力的传递牵扯到了授权的实现，我们在后面会具体阐述访问控制的授权管理。

访问控制安全标签列表

- 安全标签是限制和附属在主体或客体上的一组安全属性信息。安全标签的含义比能力更为广泛和严格，因为它实际上还建立了一个严格的安全等级集合。
- 访问控制标签列表（ACSLLs： Access Control Security Labels Lists）是限定一个用户对一个客体目标访问的安全属性集合。
- 假设请求访问的用户UserA的安全级别为S，那么UserA请求访问文件File2时，由于S<TS，访问会被拒绝；当UserA请求访问文件FileN时，因为S>C，所以允许访问。

本次课程内容（防火墙）

- 防火墙概述
- 防火墙技术
- 防火墙体系架构
- 网络隔离技术



防火墙发展史

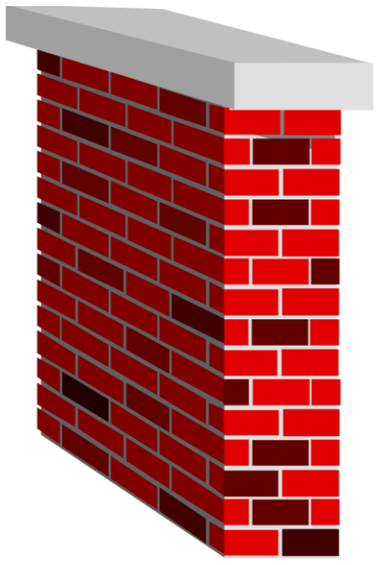
- 第一代：包过滤防火墙。1985年Cisco的IOS软件公司研制；
- 第二代：电路级网关防火墙。1989 - 1990，AT&T贝尔实验室提出的基于电路中继的结构；
- 第三代：应用层网关防火墙。20世纪90年代初，贝尔实验室；
- 第四代：动态包过滤防火墙。1991 - 1994；
- 第五代：尚未有统一的说法。观点一：1996年的内核代理结构；另一观点：1998年NAI公司的自适应代理技术。

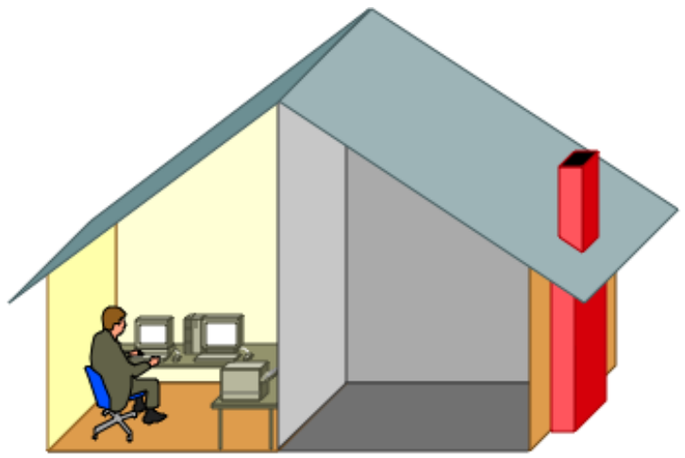
防火墙——

防火墙——概念

什么是防火墙？

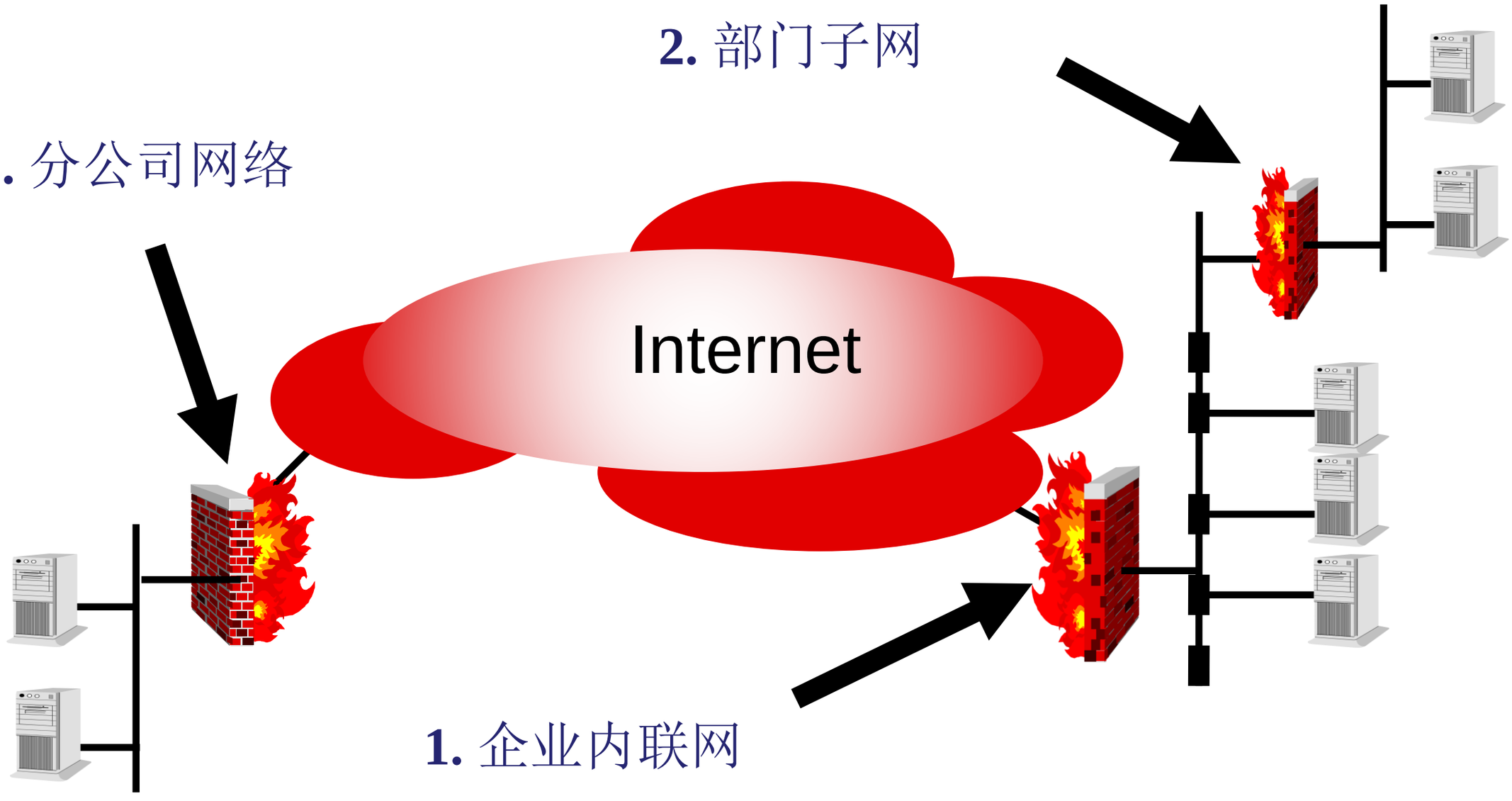
防火墙





- 传统的防火墙用来防止火从大厦的一部分传播到另一部分。

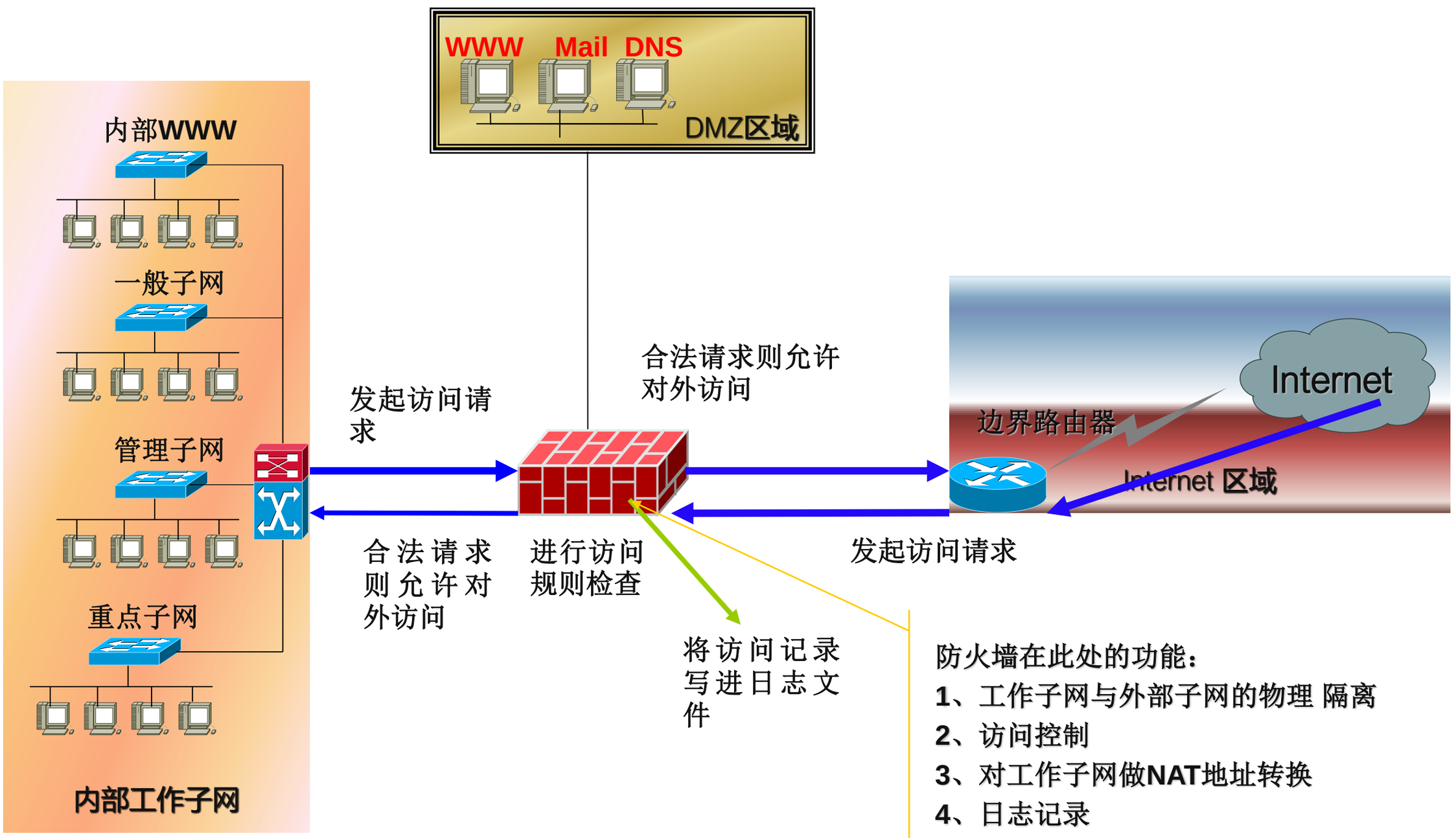
防火墙示意图



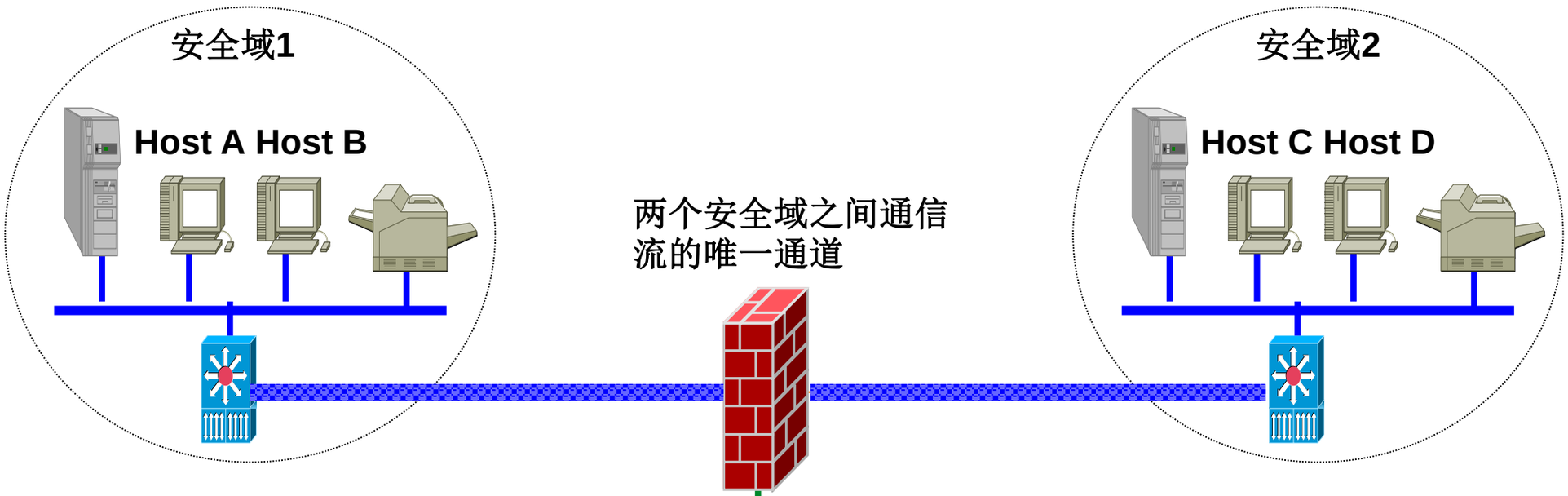
防火墙定义

- **防火墙是位于两个(或多个)网络间实施网间访问控制的一组组件的集合。**
- 它满足以下条件
  - **所有进出被保护网络的通信必须通过防火墙**
  - **所有通过防火墙的通信必须经过安全策略的过滤或者防火墙的授权**
  - **防火墙自身应对渗透(peneration)免疫**

一个典型的防火墙使用形态



IT 领域使用的防火墙概念



Source	Destination	Permit	Protocol
Host A	Host C	Pass	TCP
Host B	Host C	Block	UDP

一种高级访问控制设备，置于不同网络安全域之间的一系列部件的组合。

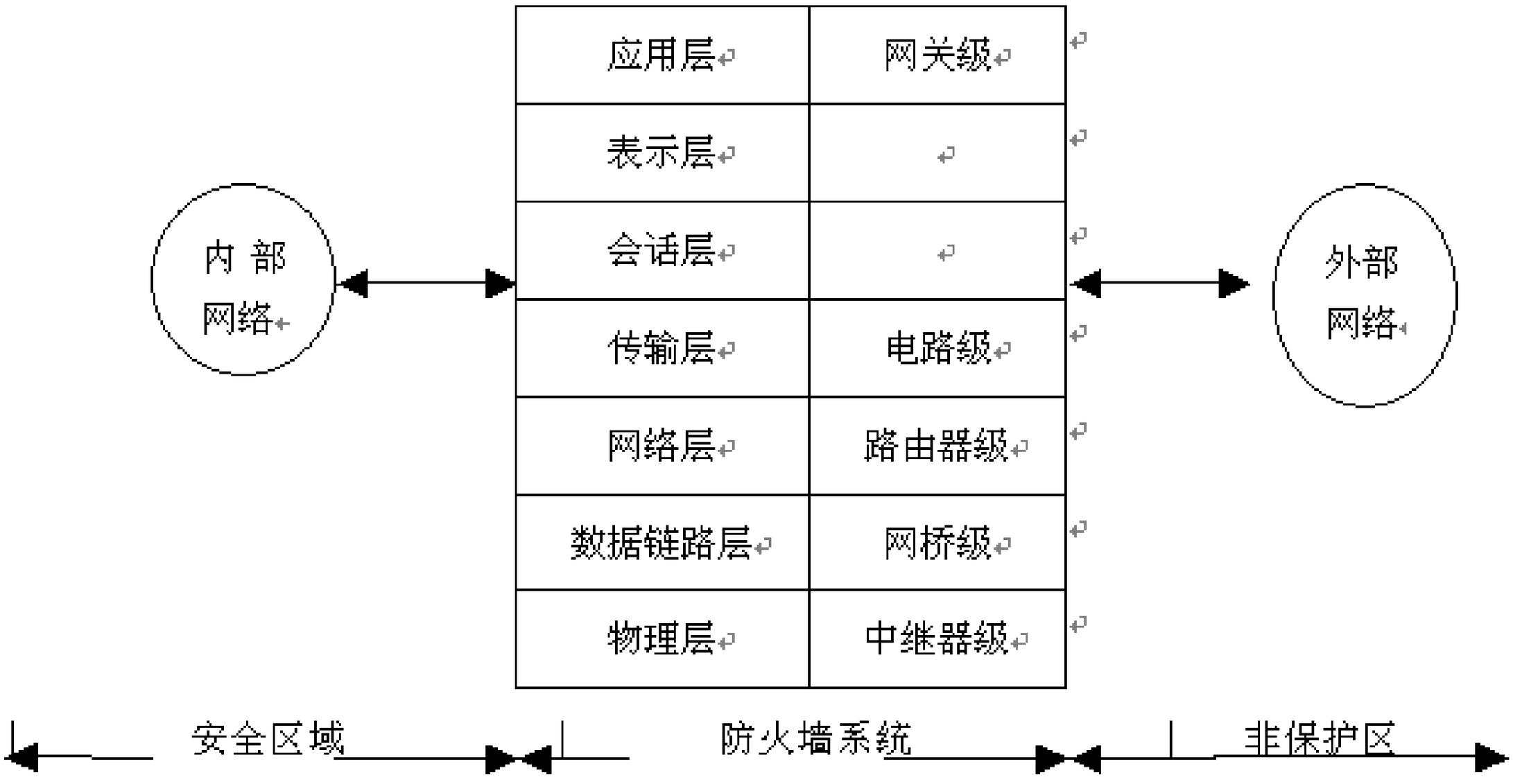
根据访问控制规则决定进出网络的行为

是不同网络安全域间通信流的唯一通道，能根据企业有关的安全政策控制（进出网络的访问行为。

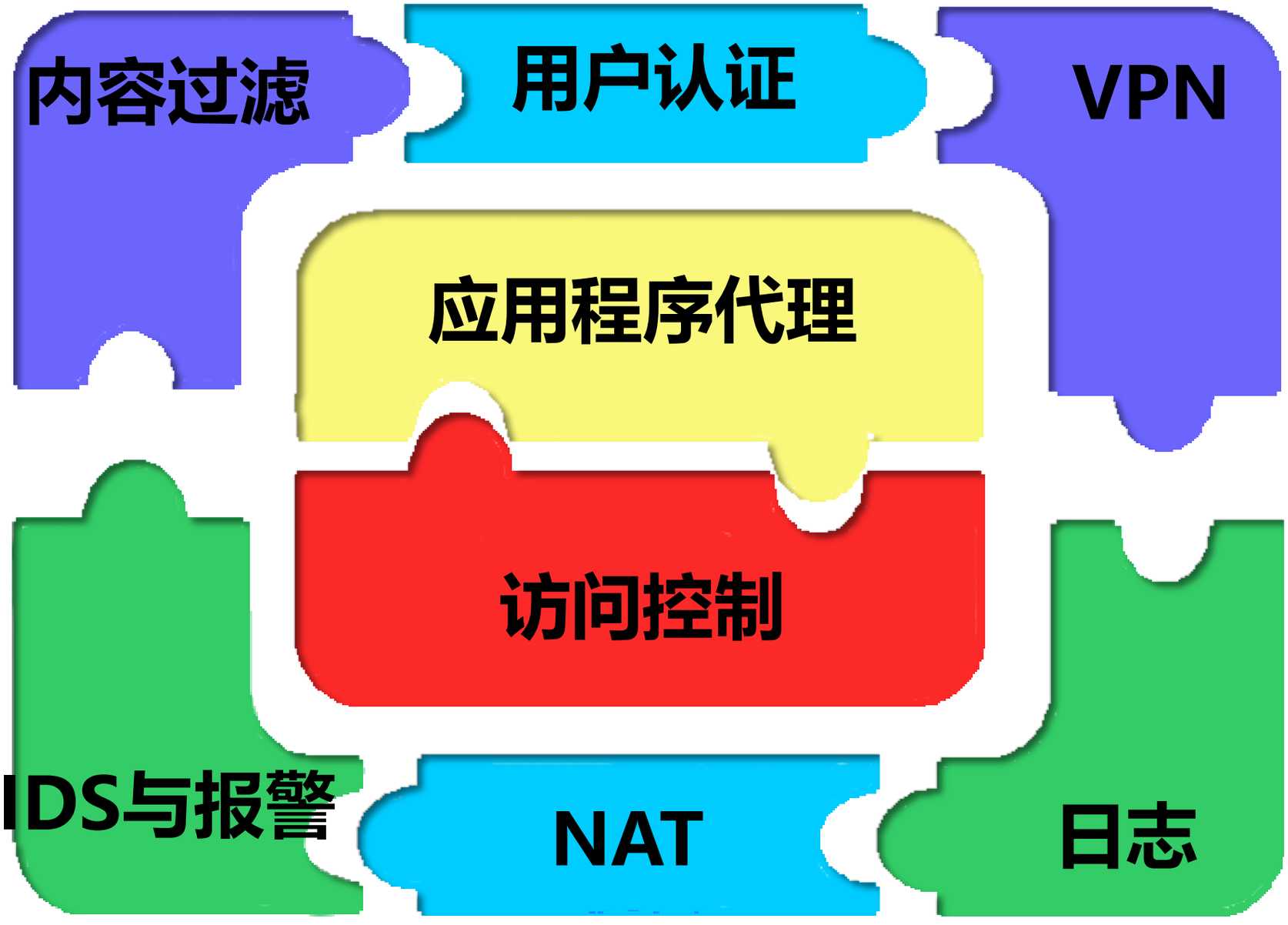
为什么需要防火墙

- 保护内部不受来自Internet的攻击。内部网特点：
  - 组成结构复杂
  - 各节点通常自主管理
  - 信任边界复杂，缺乏有效管理
  - 有显著的内外区别
  - 机构有整体的安全需求
  - 最薄弱环节原则
- 为了创建安全域
- 为了增强机构安全策略

防火墙实现层次


应用层
网关级
表示层
会话层
传输层
电路级
网络层
路由器级
数据链路层
网桥级
物理层
中继器级
内 部
网络
外部
网络
安全区域
防火墙系统
非保护区

防火墙的功能



防火墙的功能

防火墙的功能

- 过滤进出网络的数据
- 管理进出网络的访问行为
- 封堵某些禁止的业务
- 记录进出网络的信息和活动
- 对网络攻击进行检测和告警

防火墙访问控制方法

- **服务控制**：确定可以访问的服务类型；
- **方向控制**：确定特定的服务请求可以发起并允许通过防火墙；
- **用户控制**：不同的用户具有不同服务访问的权限；
- **行为控制**：控制怎样使用特定服务。如过滤垃圾邮件

防火墙技术带来的好处

- 强化安全策略
- 有效地记录Internet上的活动
- 隔离不同网络限制安全问题扩散
- 是一个安全策略的检查站

防火墙的局限性

- 使用不便认为防火墙给人虚假的安全感
- 对用户不完全透明可能带来传输延迟瓶颈及单点失效
- 无法做到绝对的安全
  - **不能防范恶意的内部人员侵入**
  - **不能防范不通过它的连接**
  - **不能防范全新的威胁**
  - **不能有效地防范数据驱动式的攻击**
  - **当使用端-端加密时其作用会受到很大的限制**

防火墙分类

从形态上分类

- 软件防火墙
- 硬件防火墙

从实现技术分类

- 包过滤防火墙
- 应用网关防火墙
- 代理防火墙
- 状态检测防火墙
- 电路级网关

从部署位置分类

- 主机防火墙
- 网络防火墙

防火墙——

防火墙——技术

防火墙技术

- 网络地址翻译
- 静态包过滤
- 动态包过滤
- 电路级网关
- 应用层网关(代理服务器)
- 状态检查包过滤

包过滤防火墙

- 包过滤防火墙对所接收的每个数据包做允许、拒绝的决定。
- 防火墙审查每个数据报以便确定其是否与某一条包过滤规则匹配。
- 过滤规则基于可以提供给IP转发过程的包头信息。
- 分为静态包过滤与动态包过滤两类。
  - 动态包过滤对外出数据包的身份做一个标记，对相同连接的进入的数据包也被允许通过，也就是说，它捕获了一个“连接”，而不是单个数据包头中的信息。

包过滤防火墙

静态包过滤
动态包过滤

- 对所接收的每个数据包做允许、拒绝的决定。
- 审查每个数据报以便确定其是否与某一条包过滤规则匹配。

判断依据

- 基本信息
  - 地址信息：源、目的IP地址
  - 协议信息：数据包协议类型TCP、 UDP、 ICMP 、IGMP等
  - 源、目的端口FTP、 HTTP 、DNS等
- 协议具体信息
  - IP选项源路由、记录路由等
  - TCP选项SYN 、ACK、 FIN 、RST等
  - 其它协议选项ICMP、 ECHO、ICMP、 ECHO 、REPLY等
- 流向及接口信息
  - 数据包流向in或out
  - 数据包流经网络接口eth0 eth1

包过滤防火墙

优点

- 逻辑简单，价格便宜，成本低；
- 对网络性能的影响较小，有较强的透明性。
- 易于匹配绝大多数网络层、传输层数据包，定制策略灵活。
- 并且它的工作与应用层无关，无须改动任何客户机和主机上的应用程序，易于安装和使用。

缺点

- 需要对IP、TCP、UDP、ICMP等协议有深入了解，否则容易出现因配置不当带来的问题；
- 据以过滤判别的只有网络层和传输层的有限信息，因而各种安全要求不能得到充分满足；
- 由于数据包的地址及端口号都在数据包的头部，不能彻底防止IP地址欺骗；
- 允许外部客户和内部主机的直接连接；
- 不提供用户的鉴别机制；
- 仅工作在网络层，提供较低水平的安全性。

包过滤防火墙原理

Source	Destination	Permit	Protocol
Host A	Host C	Pass	TCP
Host B	Host C	Block	UDP

控制策略

查找对应的
控制策略

根据策略决定如
何处理该数据包

拆开数据包

数据包

数据包

数据包

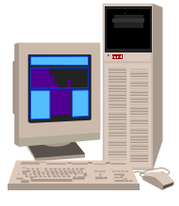
数据包

安全网域

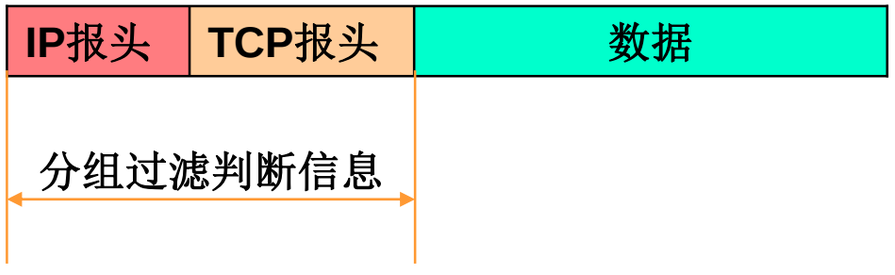
Host C Host D

数据包

数据包



过滤依据主要是TCP/IP报头里面的
信息，不能对应用层数据进行处理



包过滤防火墙原理

Source	Destination	Permit	Protocol
Host A	Host C	Pass	TCP
Host B	Host C	Block	UDP

控制策略

过滤依据主要是TCP/IP报头里面的信息，不能对应用层数据进行处理

查找对应的控制策略

根据策略决定如何处理该数据包

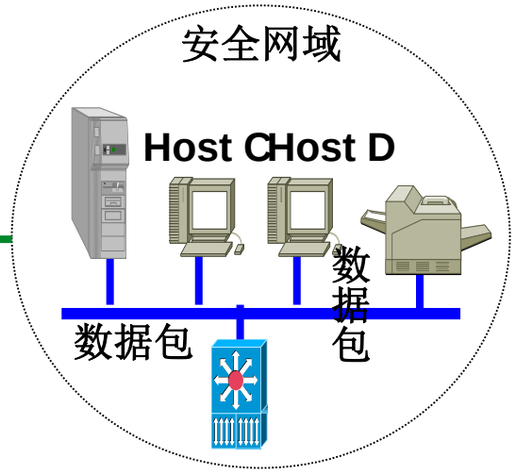
拆开数据包

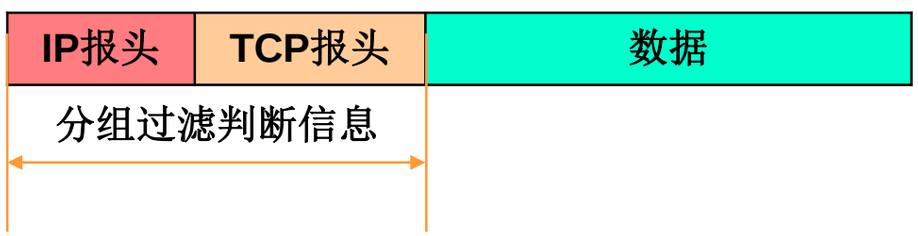
数据包

数据包

数据包

数据包





灵活的访问控制

控制策略

- 基于源IP地址
- 基于目的IP地址
- 基于源端口
- 基于目的端口
- 基于时间
- 基于IP旗标
- 基于用户
- 基于流量

可以灵活的制定的控制策略

查找对应的控制策略

根据策略决定如何处理该数据包

拆开数据包进行分析

数据包

数据包

数据包

数据包

数据包

数据包

Host C Host D

电路级网关

- 是一个通用代理服务器，它工作于OSI互联模型的会话层或是TCP/IP协议的TCP层。

- 它适用于多个协议，但它不能识别在同一个协议栈上运行的不同的应用，当然也就不需要对不同的应用设置不同的代理模块，但这种代理需要对客户端作适当修改。

- 它接受客户端的连接请求，代理客户端完成网络连接，建立起一个回路，对数据包起转发作用，数据包被提交给用户的应用层来处理。

- 通过电路级网关传递的数据似乎起源于防火墙，隐藏了被保护网络的信息。

电路级网关实现方式

- 拓扑结构同应用程序网关相同
  - 接收客户端连接请求代理客户端完成网络连接
  - 在客户和服务器间中转数据
  - 通用性强
- 简单重定向
  - 根据客户的地址及所请求端口将该连接重定向到指定的服务器地址及端口上
  - 对客户端应用完全透明
- 在转发前同客户端交换连接信息
  - 需对客户端应用作适当修改

电路级网关

优点

- 对网络性能有低度到适中程度的影响：工作的层次比包过滤防火墙高，因此过滤性能稍差，但比应用代理防火墙性能好；
- 切断了外部网络到防火墙后面的服务器的直接连接；
- 比静态、动态包过滤防火墙具有更高的安全性。

缺点

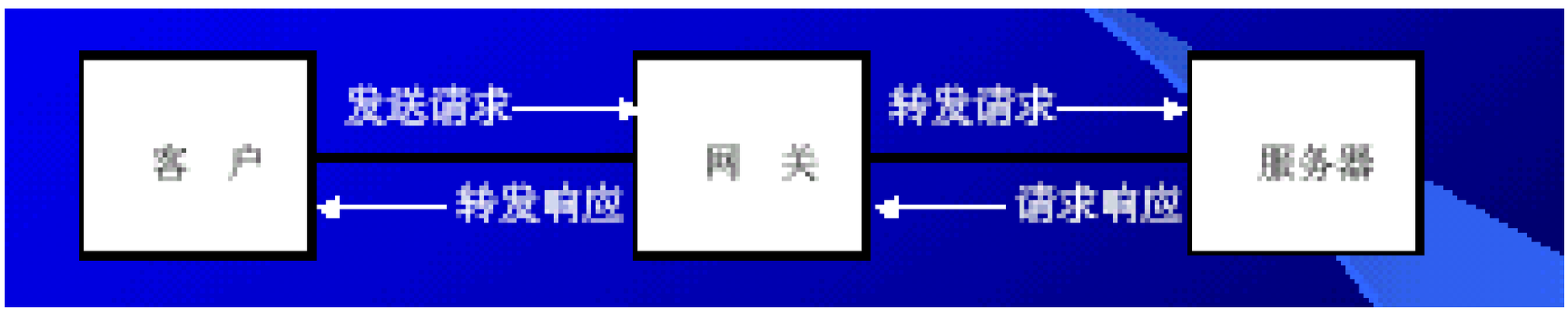
- 具有一些包过滤防火墙固有的缺陷：比如无法对数据内容进行检测，以抵御应用层的攻击等；
- 仅提供低度到中等程度的安全性。

应用网关

- 应用网关防火墙（AGF,Application Gateway Firewall），又称代理防火墙或简称应用网关。
- 应用网关在应用层处理信息
- AGF可以支持多个应用，如E-mail，Web，DNS，Telnet，FTP等

应用网关

- 应用网关代理服务器的工作过程为：
  - 首先，它对该用户的身份进行验证。
  - 若为合法用户，则把请求转发给真正的某个内部网络的主机，同时监控用户的操作，拒绝不合法的访问（访问权限）。
  - 当内部网络向外部网络申请服务时，代理服务器的工作过程刚好相反。（认证输入、输出两个方向的连接）



应用网关

优点

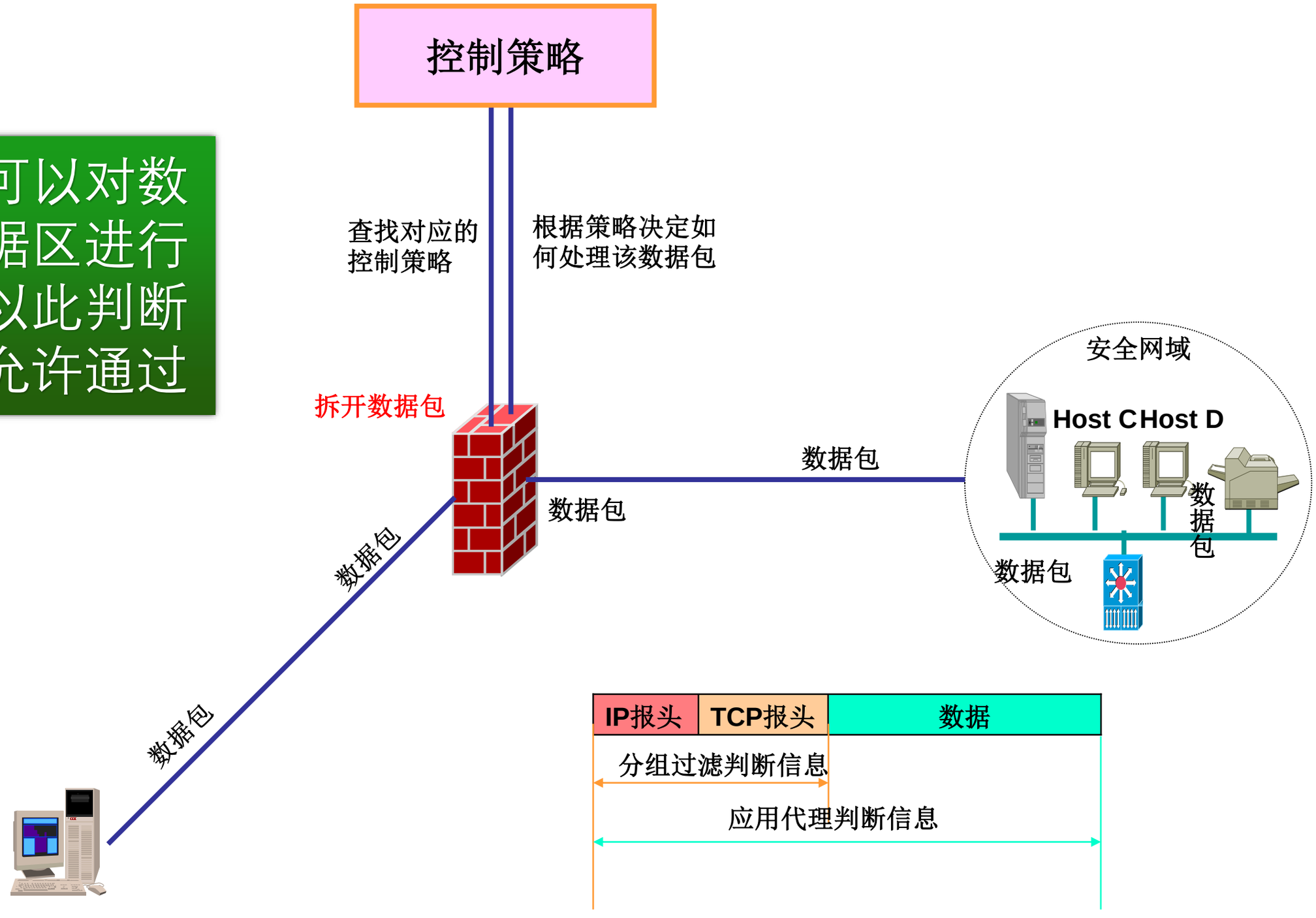
- 不允许内外网主机的直接连接；
- 可以提供比包过滤更详细的日志记录（应用层信息）；
- 可以隐藏内部IP地址；
- 认证用户而非设备；
- 可以为用户提供透明的加密机制；
- 可以与认证、授权等安全手段方便的集成；
- 监控、过滤应用层信息；

缺点

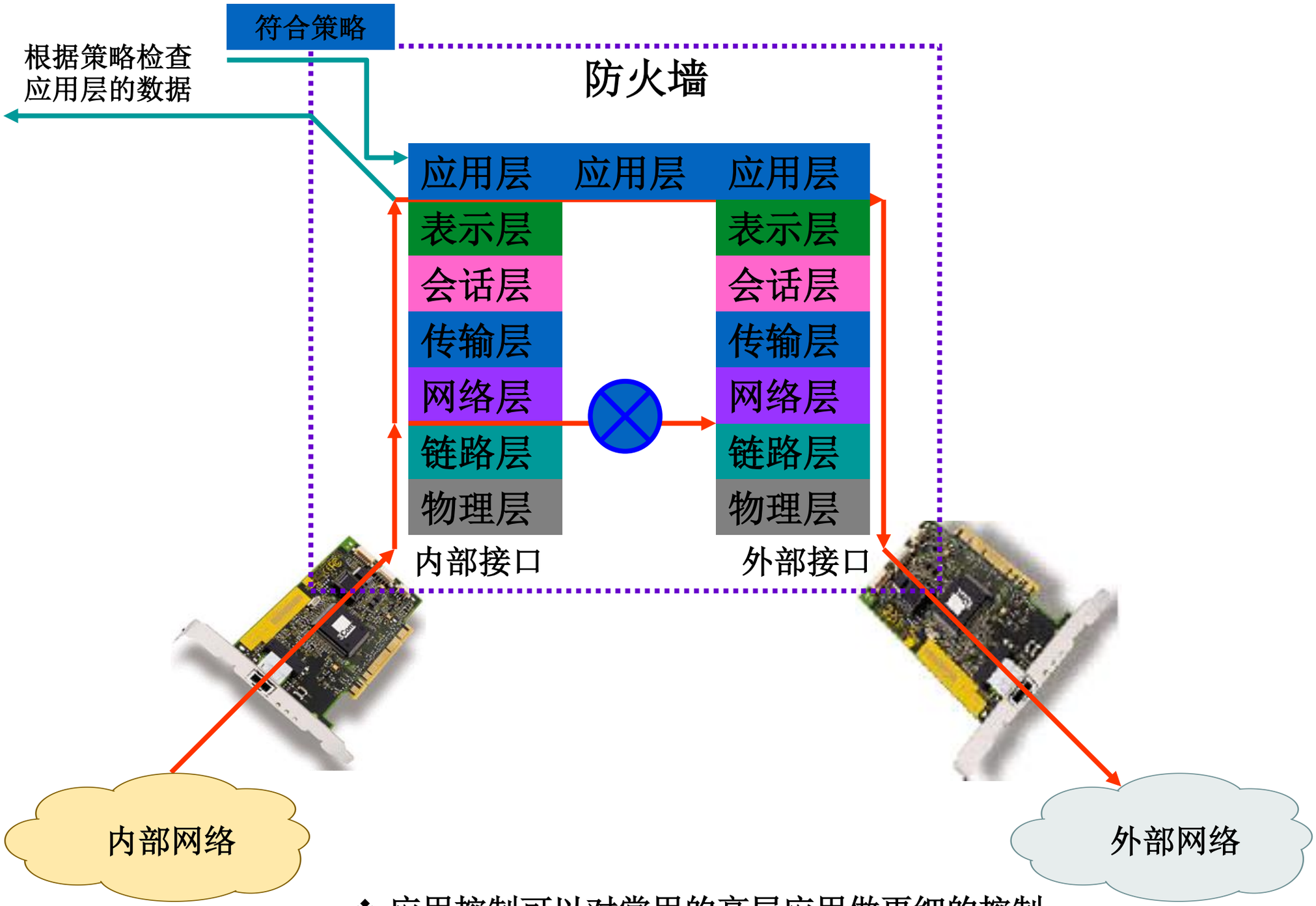
- 代理速度比包过滤慢；
- 代理对用户不透明，给用户的使用带来不便，灵活性不够；
- 这种代理技术需要针对每种协议设置一个不同的代理服务器；
- 有时要求特定的客户端软件。

应用网关

应用代理可以对数据包的数据区进行分析，并以此判断数据是否允许通过

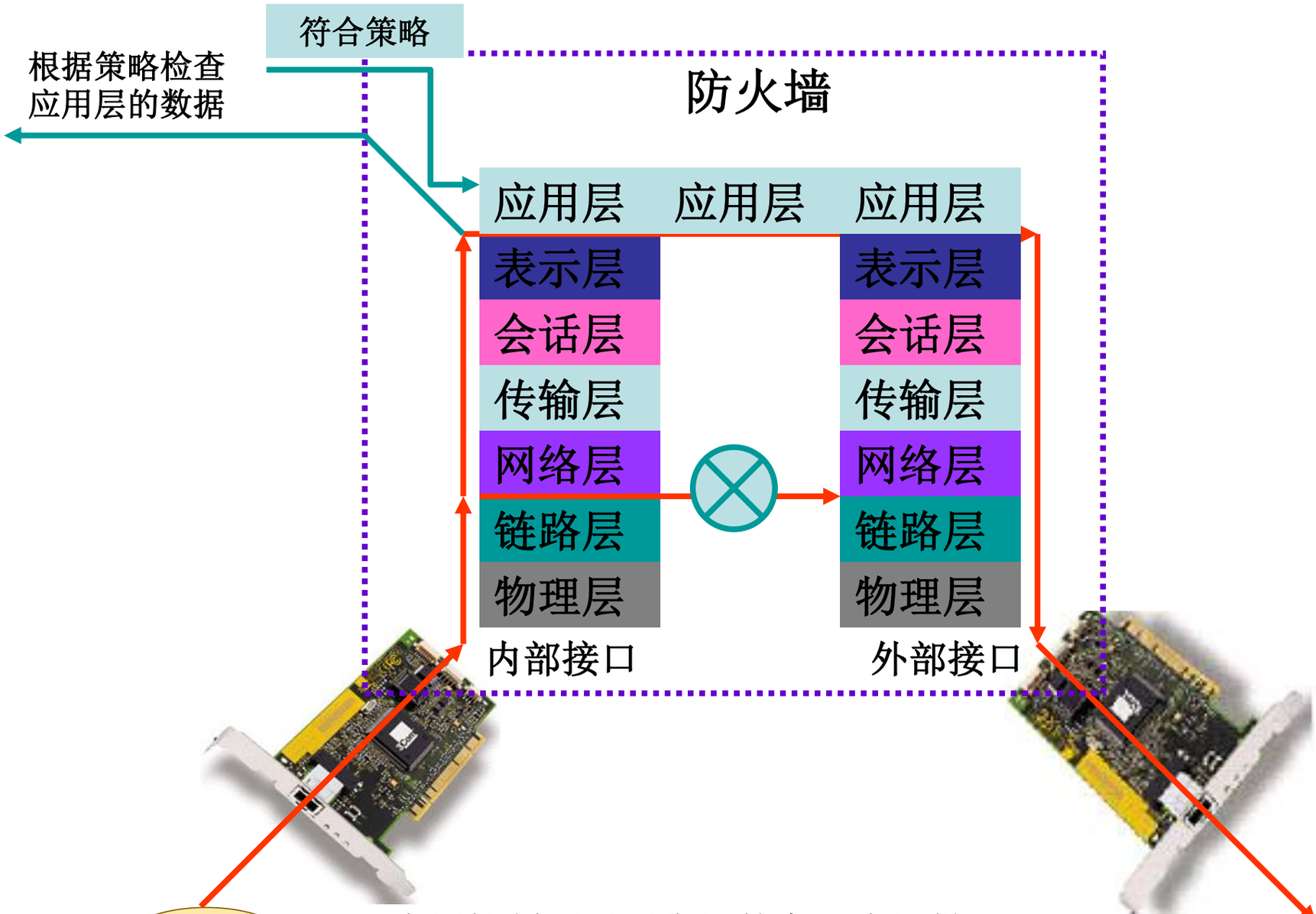


内容安全



- 应用控制可以对常用的高层应用做更细的控制
- 如HTTP的GET、POST、HEAD
- 如FTP的GET、PUT等

内容安全



- 应用控制可以对常用的高层应用做更细的控制
  - 如HTTP的GET、POST、HEAD
  - 如FTP的GET、PUT等

状态检测防火墙

- 状态检测防火墙是在动态包过滤防火墙基础上，增加了状态检测机制而形成的。
- 具有连接的跟踪能力。

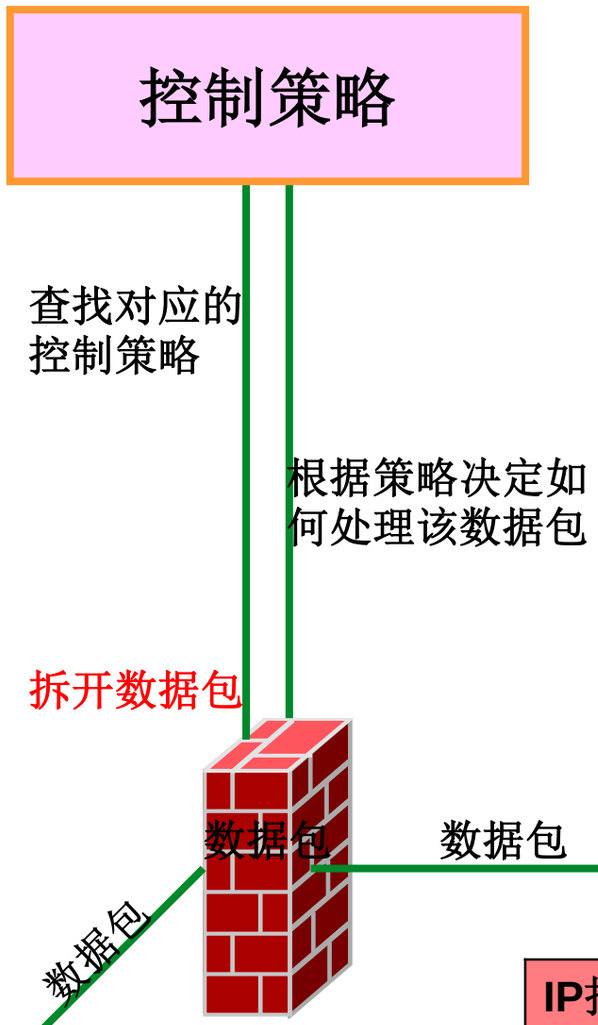
示例

- 以TCP协议为例：所谓的状态检测机制关注的主要问题不再仅是SYN和ACK标志位，或者是来源端口和目标端口，还包括了序号、窗口大小等其它TCP协议信息。

优缺点

- 优点：
  - 具备动态包过滤的所有优点，同时具有更高的安全性。
- 缺点：
  - 检测的层次仅限于网络层与传输层，无法对应用层内容进行检测，从而无法抵抗应用层的攻击；
  - 性能比动态包过滤稍差：因为检测更多的内容。

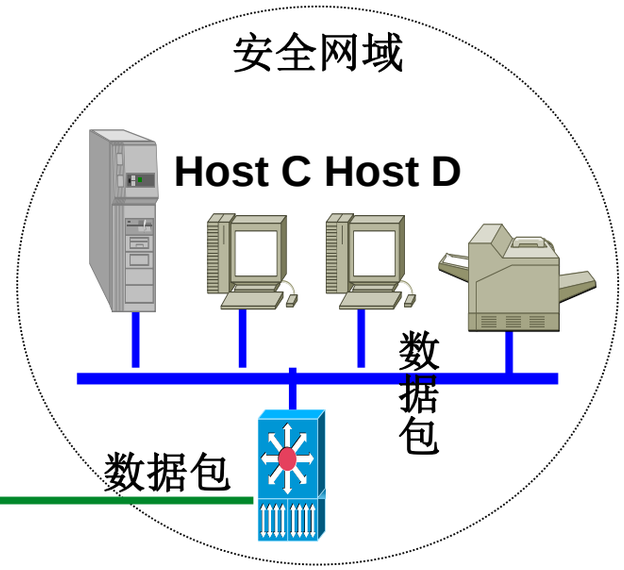
状态检测原理



查找对应的
控制策略

根据策略决定如
何处理该数据包

拆开数据包

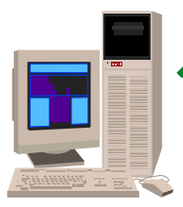


数据包

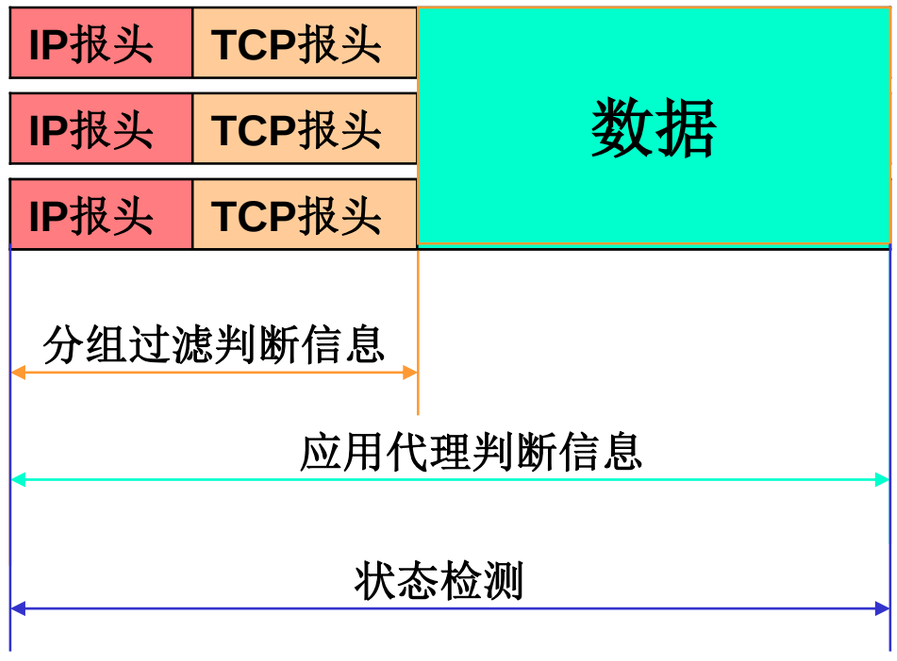
数据包

数据包

数据包

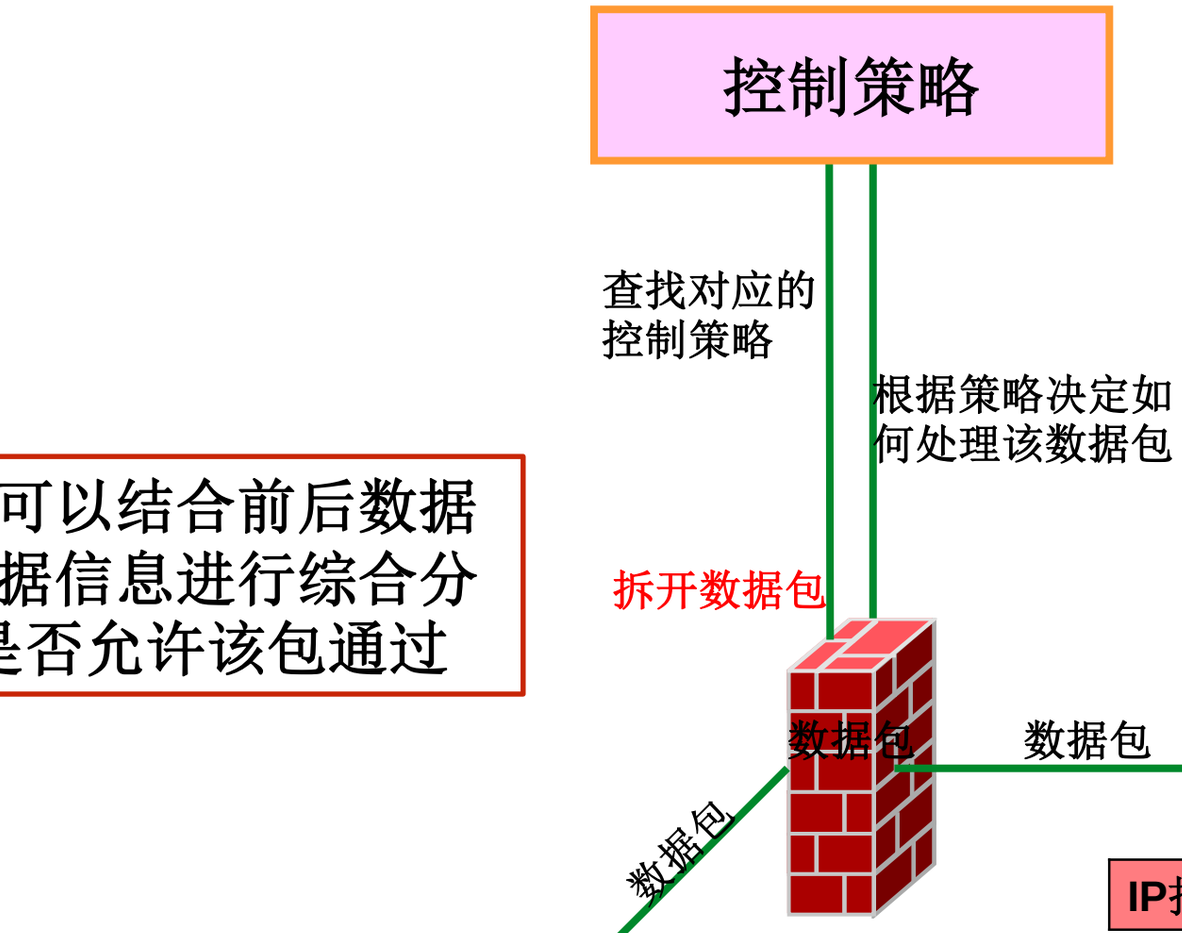


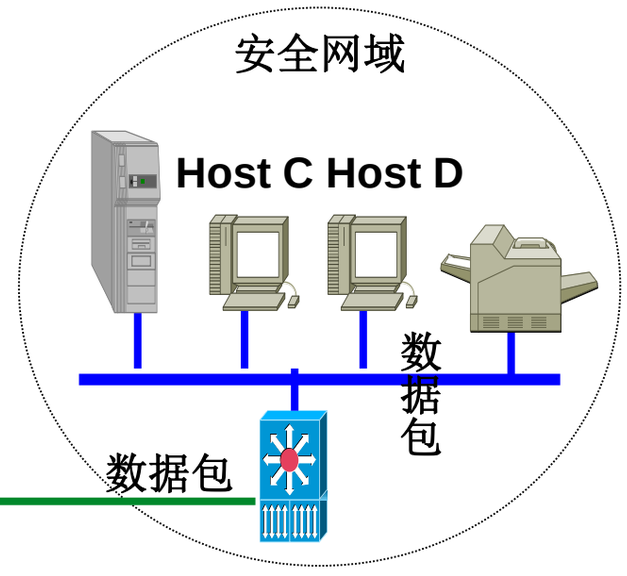
状态检测可以结合前后数据包里的数据信息进行综合分析决定是否允许该包通过

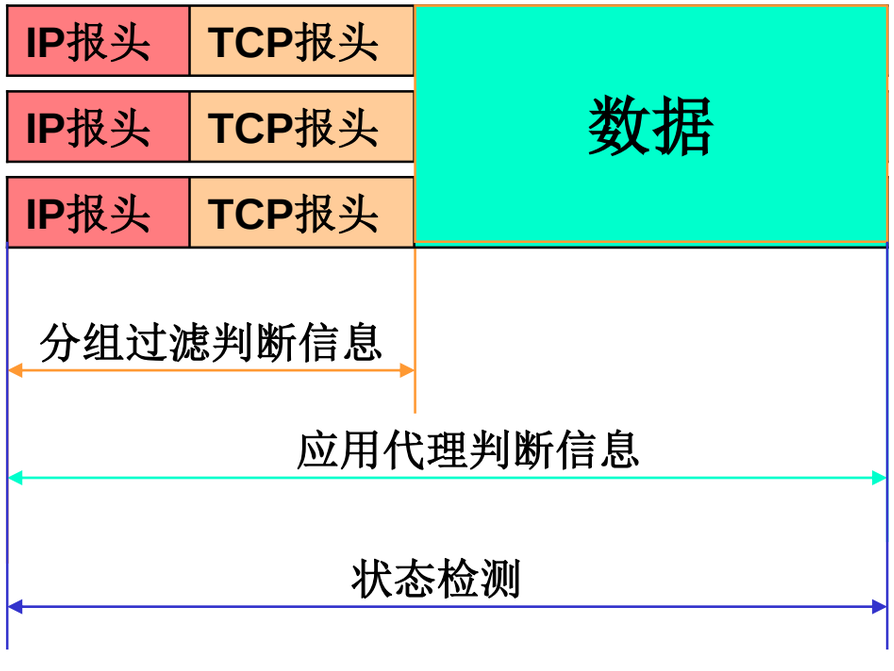


状态检测原理

状态检测可以结合前后数据包里的数据信息进行综合分析决定是否允许该包通过







防火墙——

网络地址翻译(NAT)

网络地址翻译(NAT)

- 目的
  - 解决IP地址空间不足问题
  - 向外界隐藏内部网结构
- 方式
  - M-1 多个内部网地址翻译到1个IP地址
  - 1-1 简单的地址翻译
  - M-N 多个内部网地址翻译到N个IP地址池

网络地址翻译(NAT)

- 目的
  - 解决IP地址空间不足问题
  - 向外界隐藏内部网结构
- 方式
  - M-1 多个内部网地址翻译到1个IP地址
  - 1-1 简单的地址翻译
  - M-N 多个内部网地址翻译到N个IP地址池

NAT技术

- 地址翻译NAT（Network Address Translation）就是将一个IP地址用另一个IP地址代替。地址翻译主要用在两个方面：
  - 网络管理员希望隐藏内部网络的IP地址。这样互联网上的主机无法判断内部网络的情况。
  - 内部网络的IP地址是无效的IP地址。 这种情况主要是因为现在的IP地址不够用，要申请到足够多的合法IP地址很难办到，因此需要翻译IP地址。

NAT技术

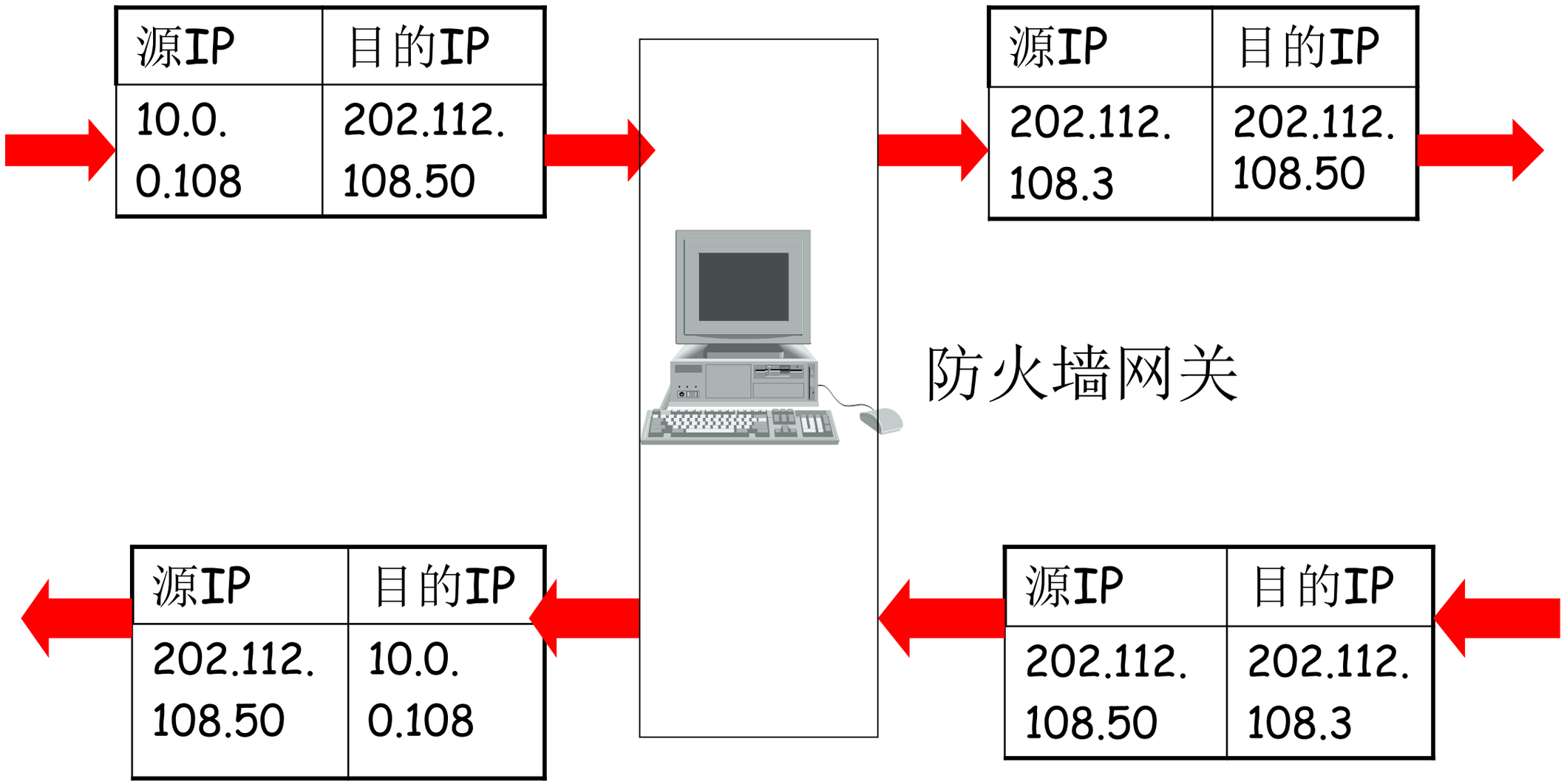
- 网络地址转就是在防火墙上装一个合法IP地址集，然后
  - 当内部某一用户要访问Internet时，防火墙动态地从地址集中选一个未分配的地址分配给该用户；
  - 同时，对于内部的某些服务器如Web服务器，网络地址转换器允许为其分配一个固定的合法地址。

NAT技术

- NAT的三种类型
  - 静态NAT
    - 内部网络每个主机都永久映射成外部合法的地址
  - NAT池
    - 在外部网络中定义了一系列合法地址，采用动态分配的方法映射到内部网络
  - 端口NAT（PNAT)
    - 把内部地址映射到外部网络的一个IP地址的不同端口上

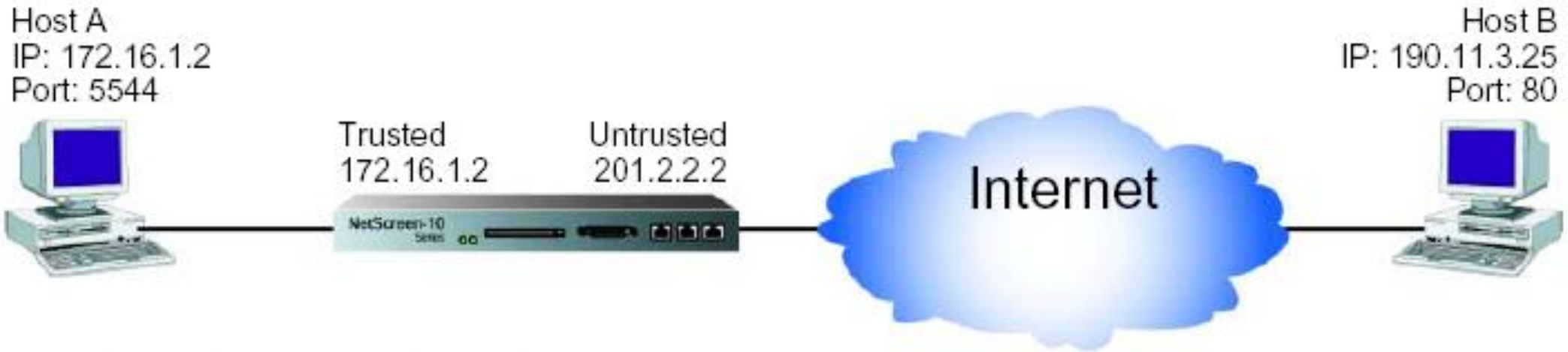
NAT技术

- **基本原理**


源IP
目的IP
10.0.0.108
202.112.108.50
源IP
目的IP
202.112.108.3
202.112.108.50
防火墙网关
源IP
目的IP
202.112.108.50
10.0.0.108
源IP
目的IP
202.112.108.50
202.112.108.3

NAT技术

- PNAT模式



DST IP	SRC IP	DST Port	SRC Port	. . .	DATA
190.11.3.25	172.16.1.2	80	5544		

Outbound (Before NAT)

DST IP	SRC IP	DST Port	SRC Port	. . .	DATA
190.11.3.25	201.2.2.2	80	7112		

Outbound (After NAT)

DST IP	SRC IP	DST Port	SRC Port	. . .	DATA
201.2.2.2	190.11.3.25	7112	80		

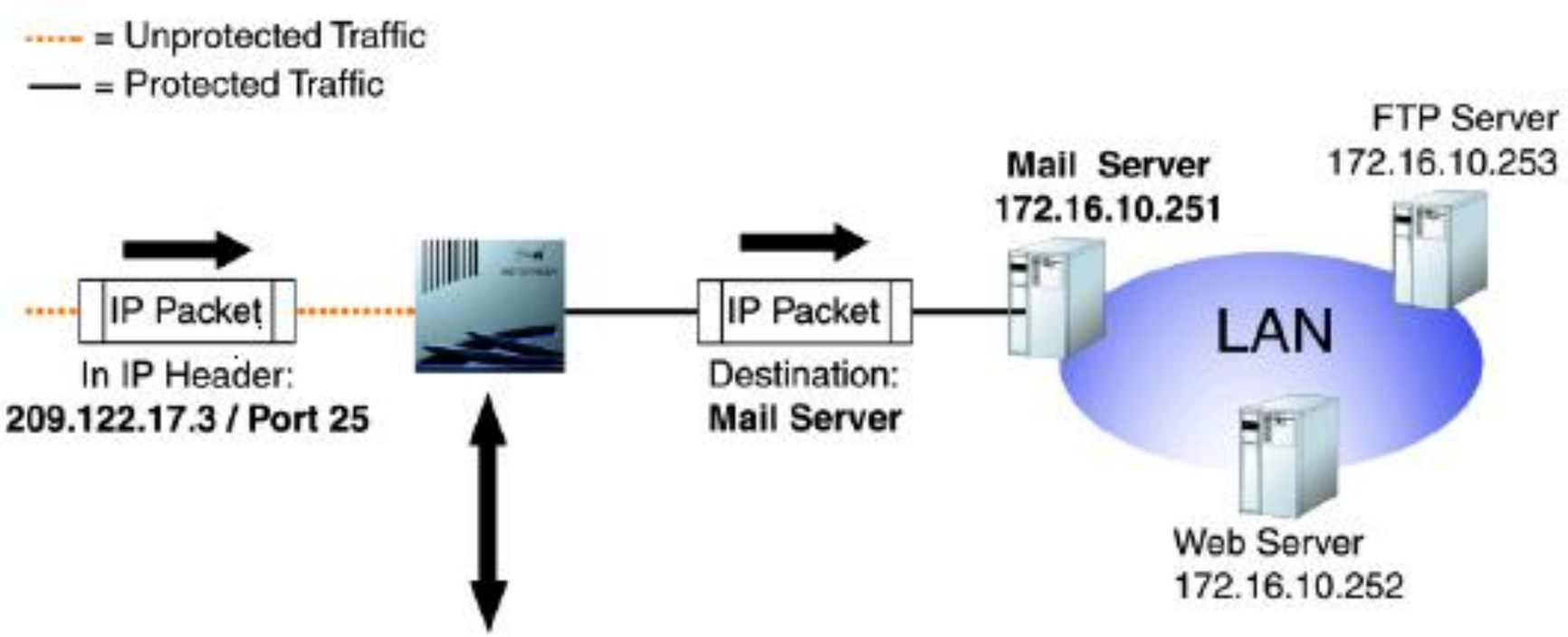
Inbound (Before NAT)

DST IP	SRC IP	DST Port	SRC Port	. . .	DATA
190.11.3.25	190.11.3.25	5544	80		

Inbound (After NAT)

NAT技术

- 虚拟IP模式

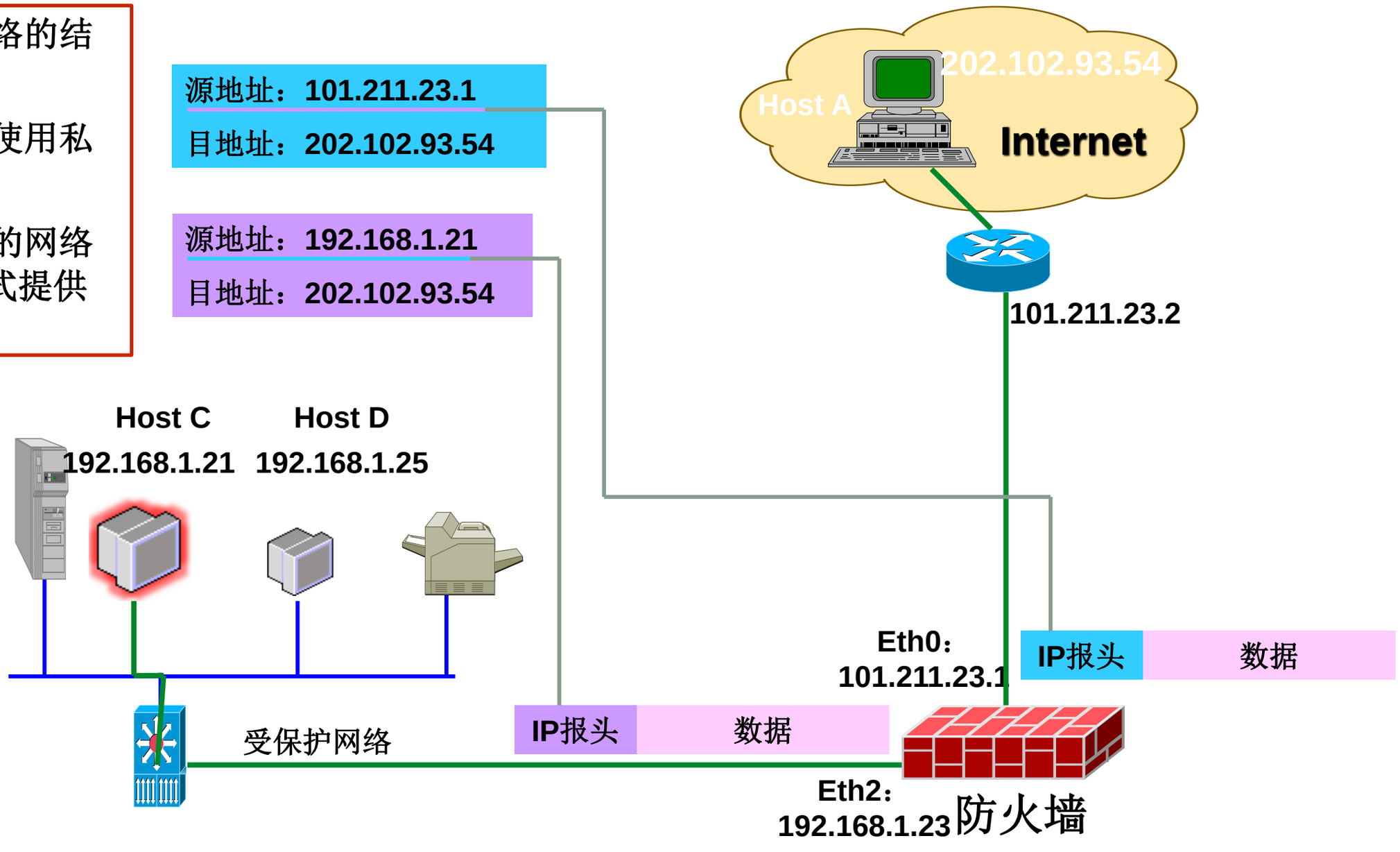


Untrusted IP	Port		Trusted IP
209.122.17.3	**25 (SMTP)**	→	**172.16.10.251**
209.122.17.3	80 (WWW)	→	172.16.10.252
209.122.17.3	21 (FTP)	→	172.16.10.253
.	.	.	.
.	.	.	.
.	.	.	.

Virtual IP Routing Table

NAT转换 & IP复用

- ❖ 隐藏了内部网络的结构
- ❖ 内部网络可以使用私有IP地址
- ❖ 公开地址不足的网络可以使用这种方式提供IP复用功能



防火墙——

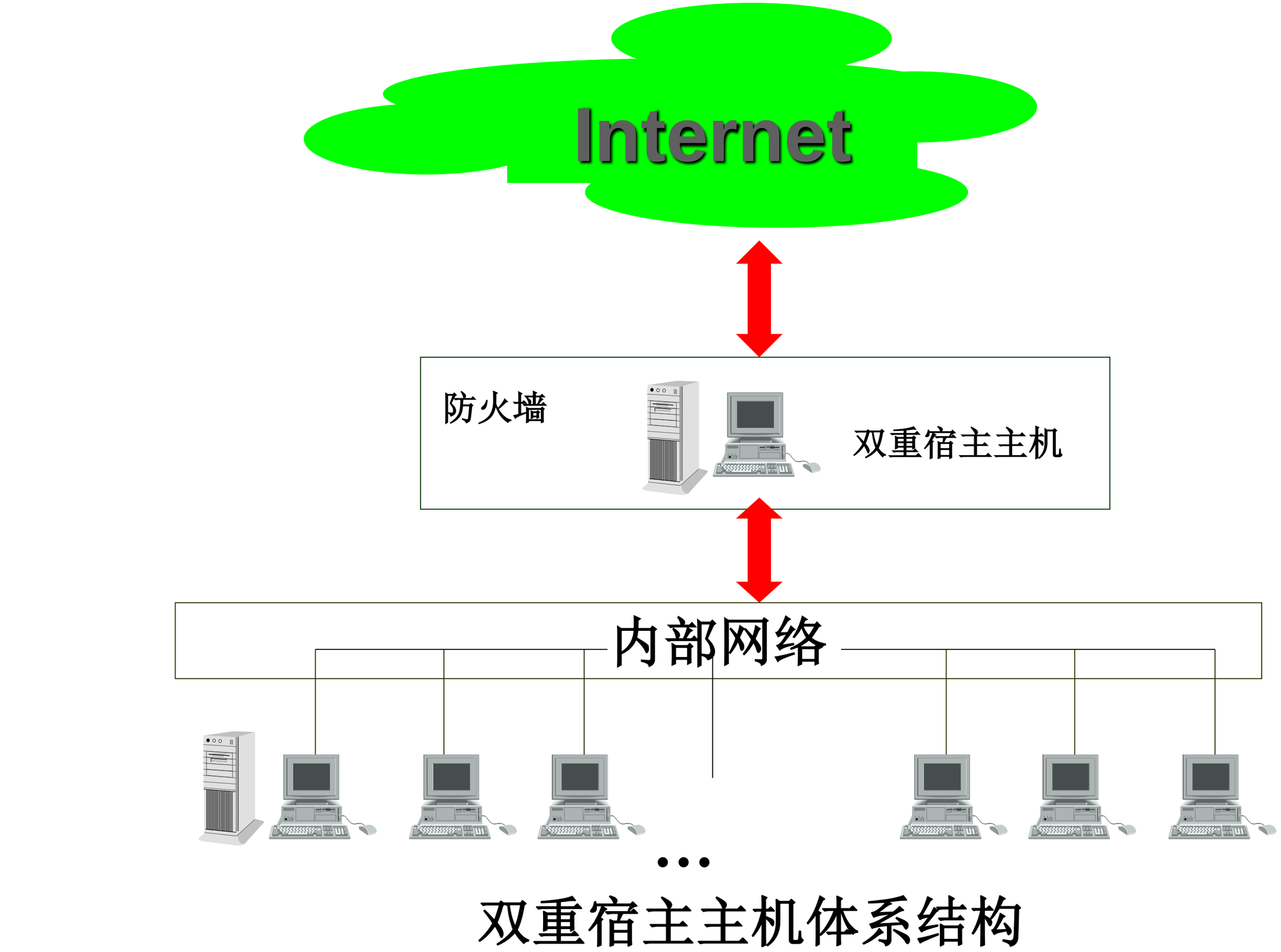
防火墙——体系架构

防火墙体系结构

- 双重宿主主机体系结构
- 屏蔽主机体系结构
- 屏蔽子网体系结构

双重宿主主机体系结构

- 双重宿主主机体系结构是围绕双重宿主主机构筑的。
- 双重宿主主机至少有两个网络接口
  - 它位于内部网络和外部网络之间，这样的主机可以充当与这些接口相连的网络之间的路由器，它能从一个网络接收IP数据包并将之发往另一网络。
  - 双重宿主主机的防火墙体系结构禁止这种发送功能，完全阻止了内外网络之间的IP通信。
- 两个网络之间的通信可通过应用层数据共享和应用层代理服务的方法实现。一般情况下采用代理服务的方法。



双重宿主主机体系结构

双重宿主主机体系结构

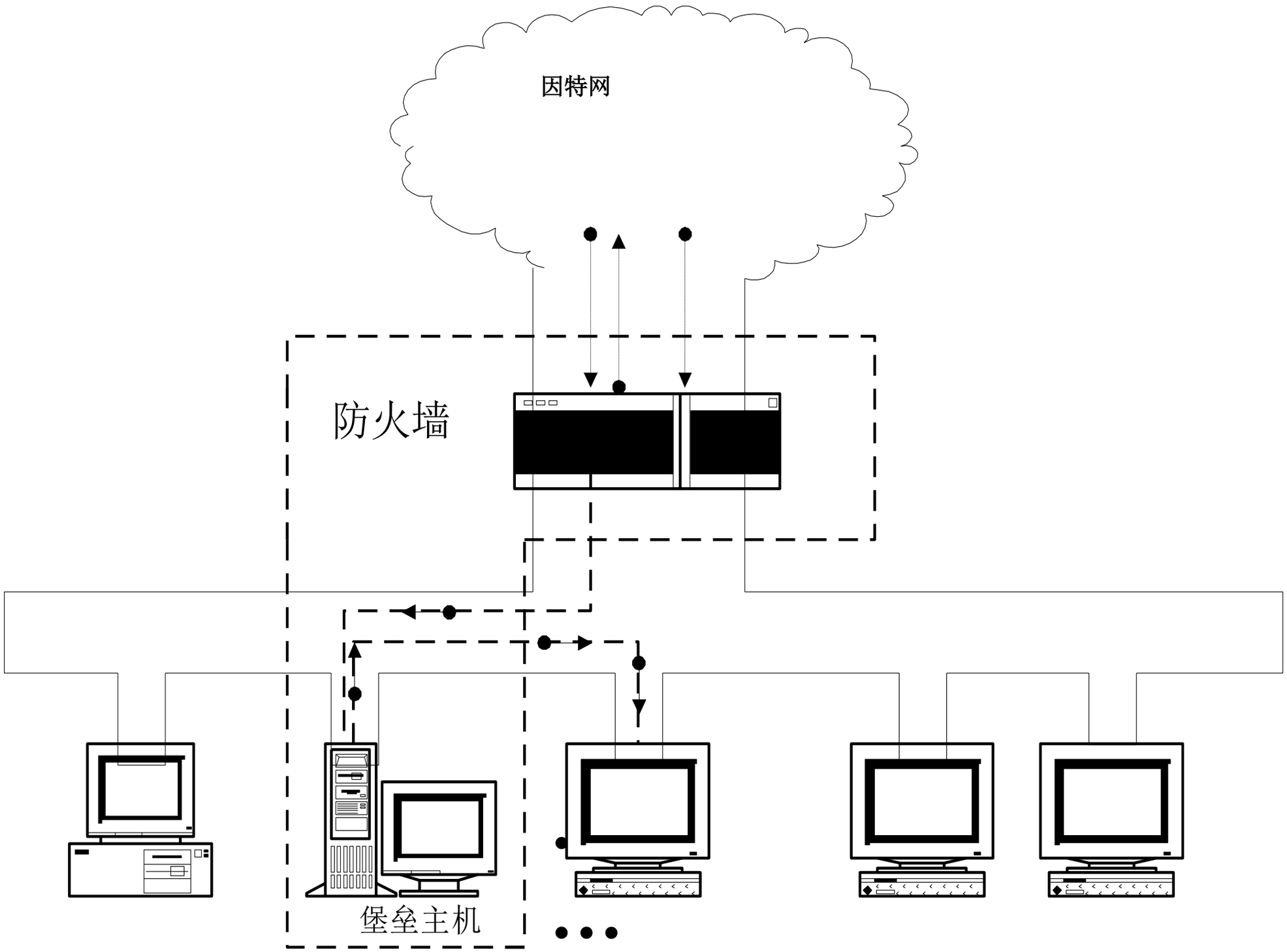
- 双重宿主主机的特性：
  - 安全至关重要（唯一通道），其用户口令控制安全是关键
  - 必须支持很多用户的访问（中转站），其性能非常重要。
- 缺点：双重宿主主机是隔开内外网络的唯一屏障，一旦它被入侵，内部网络便向入侵者敞开大门。

屏蔽主机体系结构

- 屏蔽主机体系结构由防火墙和内部网络的堡垒主机承担安全责任。一般这种防火墙较简单，可能就是简单的路由器。

- 典型构成：**包过滤路由器 + 堡垒主机**。
  - 包过滤路由器配置在内部网和外部网之间，保证外部系统对内部网络的操作只能经过堡垒主机。
  - 堡垒主机配置在内部网络上，是外部网络主机连接到内部网络主机的桥梁，它需要拥有高等级的安全。

屏蔽主机体系结构



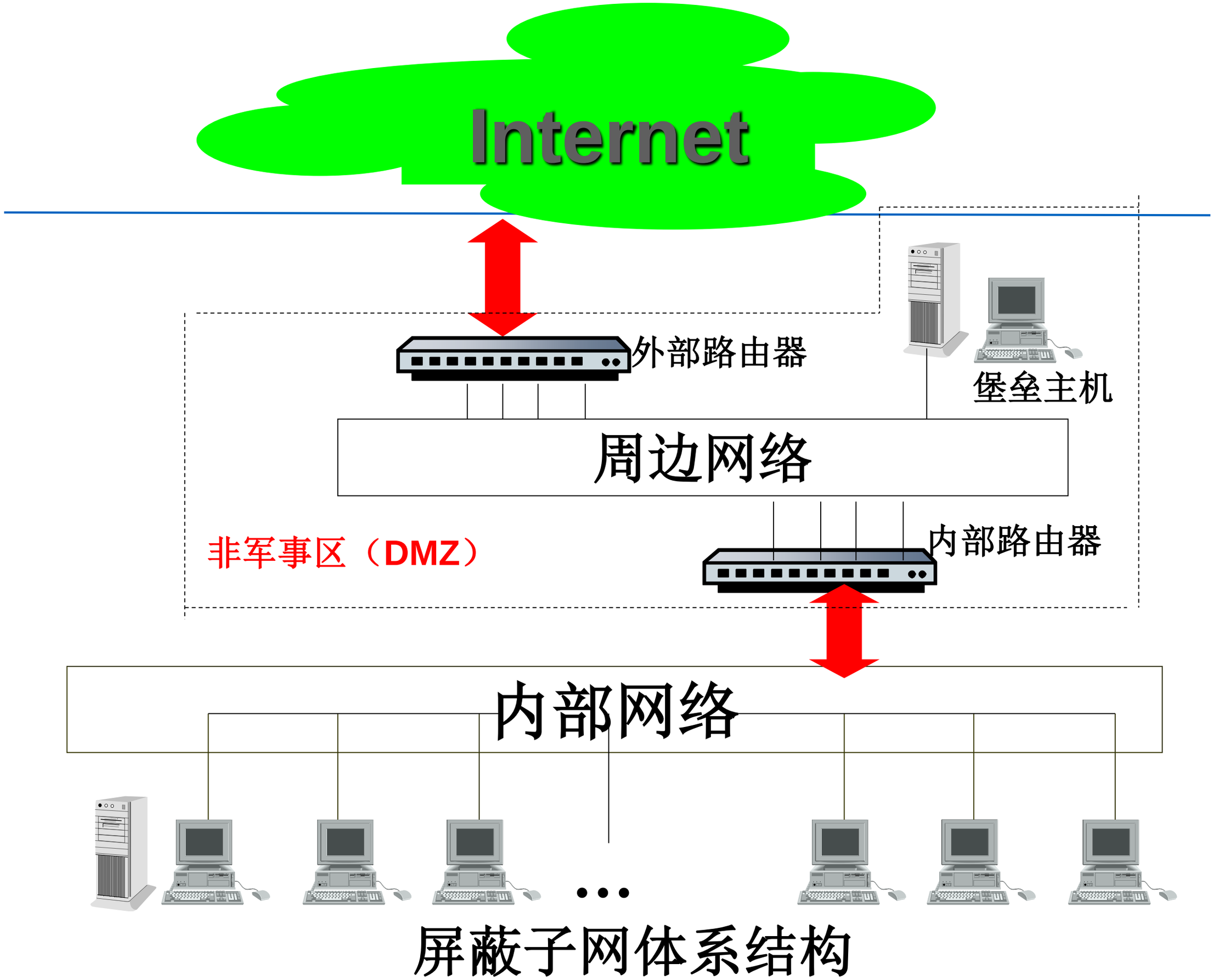
屏蔽主机体系结构

- 屏蔽路由器可按如下规则之一进行配置：
  - 允许内部主机为了某些服务请求与外部网上的主机建立直接连接（即允许那些经过过滤的服务）。
  - 不允许所有来自外部主机的直接连接。
- 安全性更高，双重保护：实现了网络层安全（包过滤）和应用层安全（代理服务）。
- 缺点：过滤路由器能否正确配置是安全与否的关键。如果路由器被损害，堡垒主机将被穿过，整个网络对侵袭者是开放的。

屏蔽子网体系结构

- 屏蔽子网体系结构在本质上与屏蔽主机体系结构一样，但添加了额外的一层保护体系——周边网络。堡垒主机位于周边网络上，周边网络和内部网络被内部路由器分开。

- 原因：堡垒主机是用户网络上最容易受侵袭的机器。通过在周边网络上隔离堡垒主机，能减少在堡垒主机被侵入的影响。



屏蔽子网体系结构

屏蔽子网体系结构

- 周边网络是一个防护层，在其上可放置一些信息服务器，它们是牺牲主机，可能会受到攻击，因此又被称为非军事区（DMZ）。

- 周边网络的作用：即使堡垒主机被入侵者控制，它仍可消除对内部网的侦听。例: netxray等的工作原理。

屏蔽子网体系结构

- 堡垒主机
  - 堡垒主机位于周边网络，是整个防御体系的核心。
  - 堡垒主机可被认为是应用层网关，可以运行各种代理服务程序。
  - 对于出站服务不一定要求所有的服务经过堡垒主机代理，但对于入站服务应要求所有服务都通过堡垒主机。

屏蔽子网体系结构

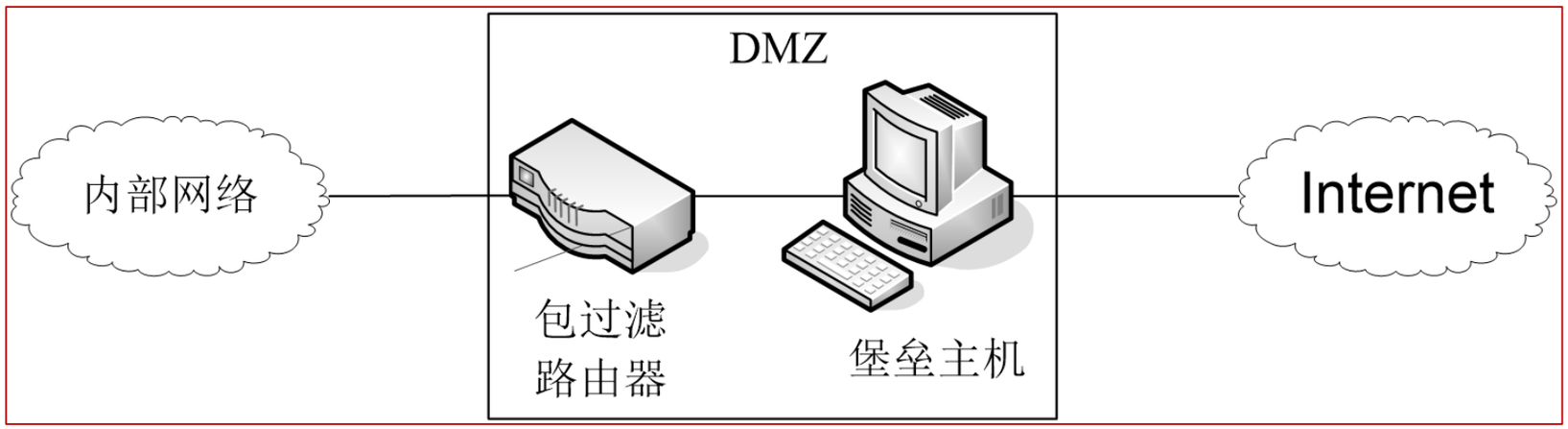
- 外部路由器（访问路由器）
  - 作用：保护周边网络和内部网络不受外部网络的侵犯。
    - 它把入站的数据包路由到堡垒主机。
    - 防止部分IP欺骗，它可分辨出数据包是否真正来自周边网络，而内部路由器不可。

- 内部路由器（阻塞路由器）
  - 作用：保护内部网络不受外部网络和周边网络的侵害，它执行大部分过滤工作。
  - 外部路由器一般与内部路由器应用相同的规则。

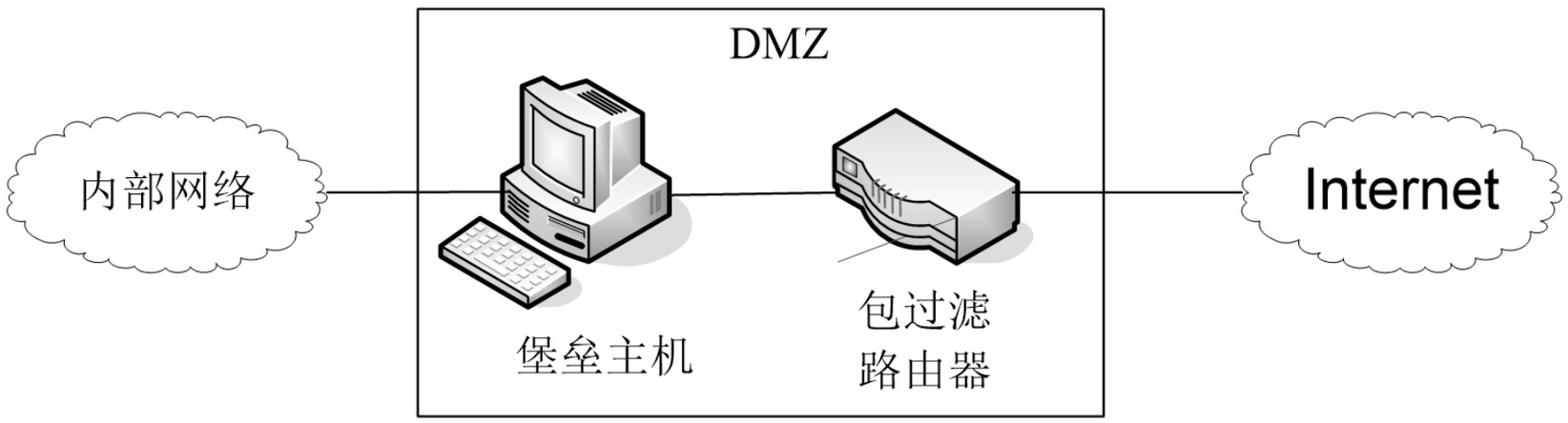
屏蔽子网体系结构优点

- 入侵者需突破3个不同的设备才能入侵内部网络
- 只对外通告DMZ区的网络，保证内部网络不可见
- 内部网络用户通过堡垒主机或代理服务器访问外部网络

其它的防火墙结构

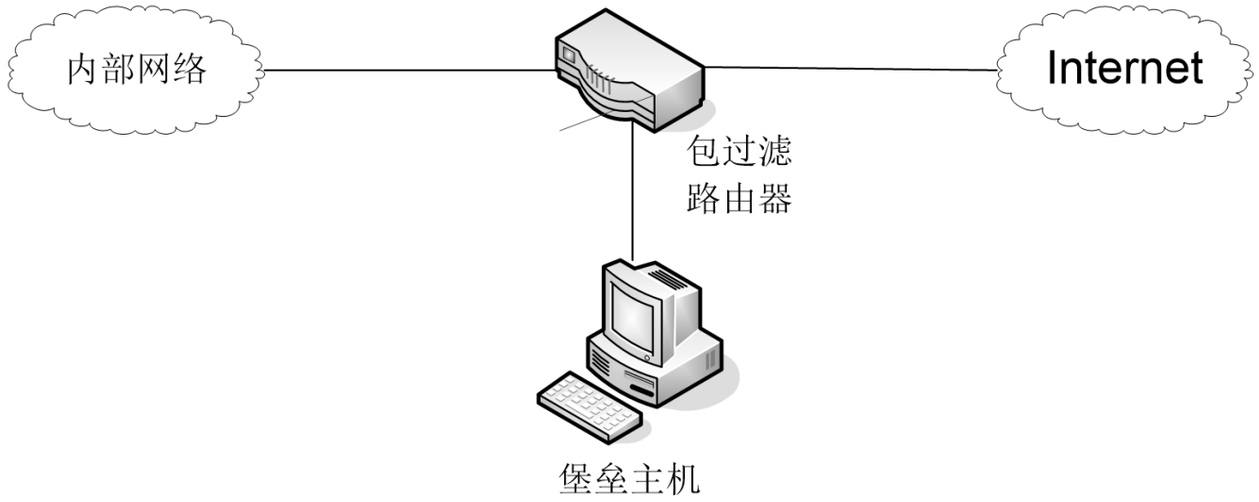


(1)合并“非军事区”的外部路由器和堡垒主机结构

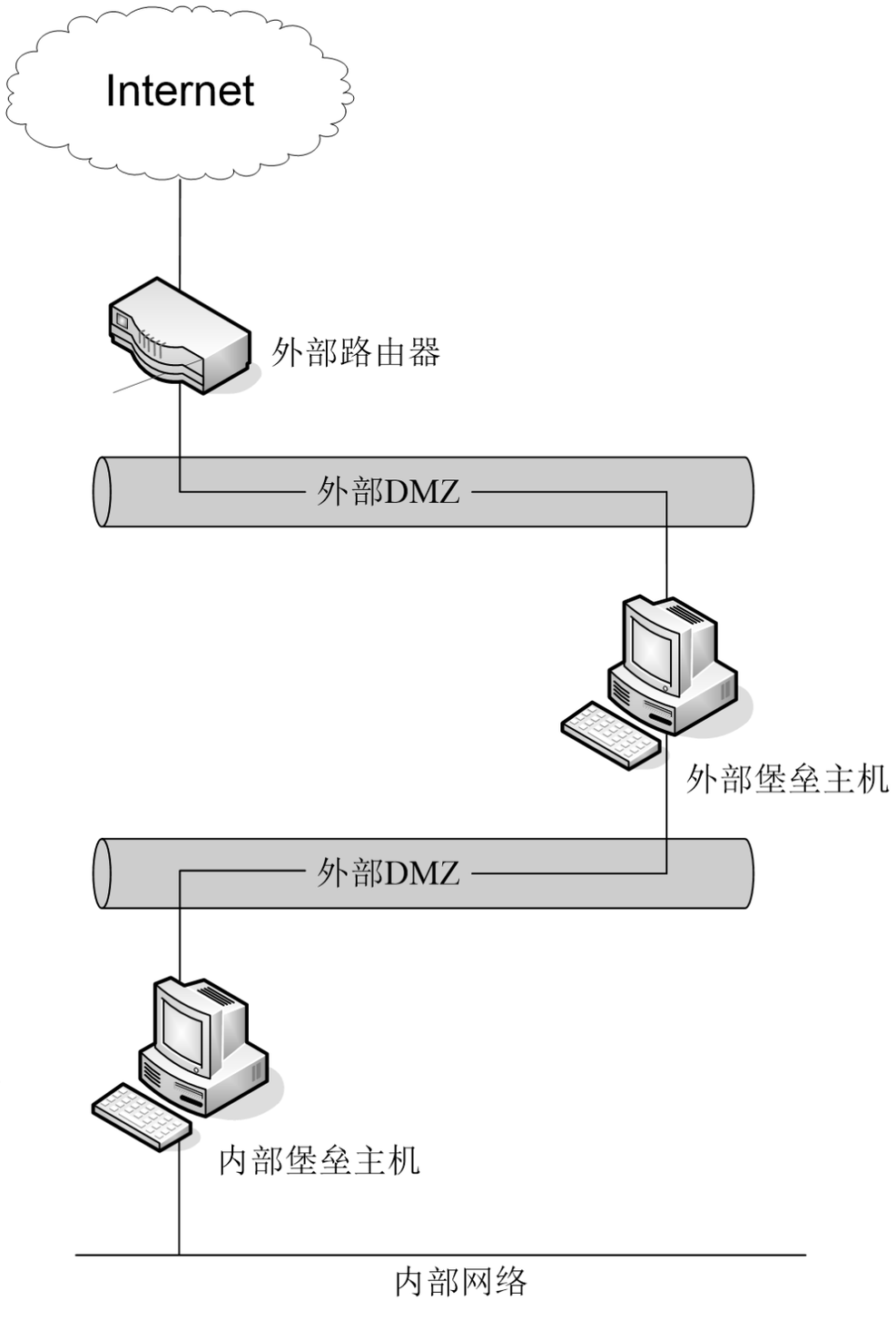


(2)合并内部路由器和堡垒主机结构

其它的防火墙结构

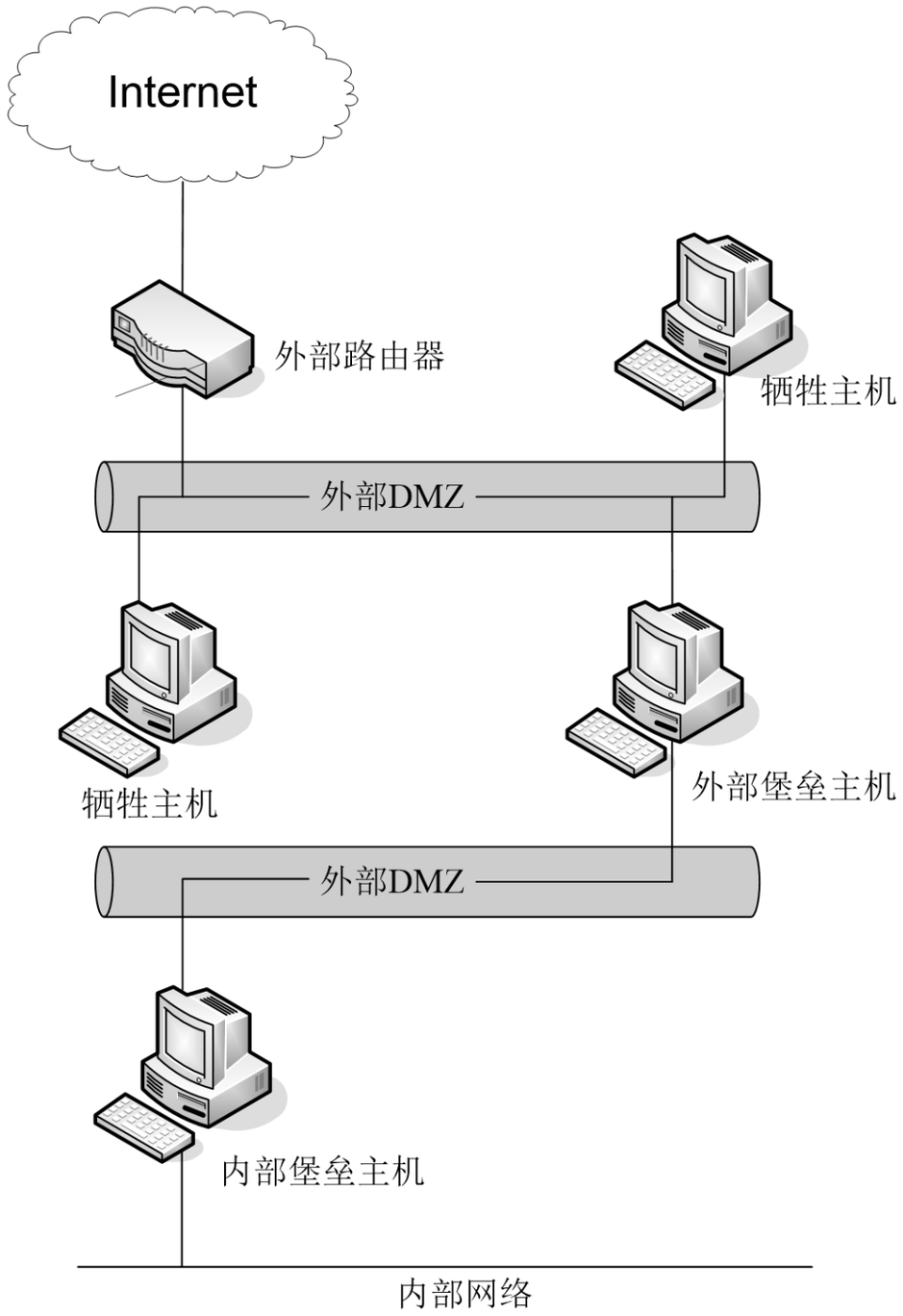


(3) 合并DMZ的内部路由器和外部路由器结构

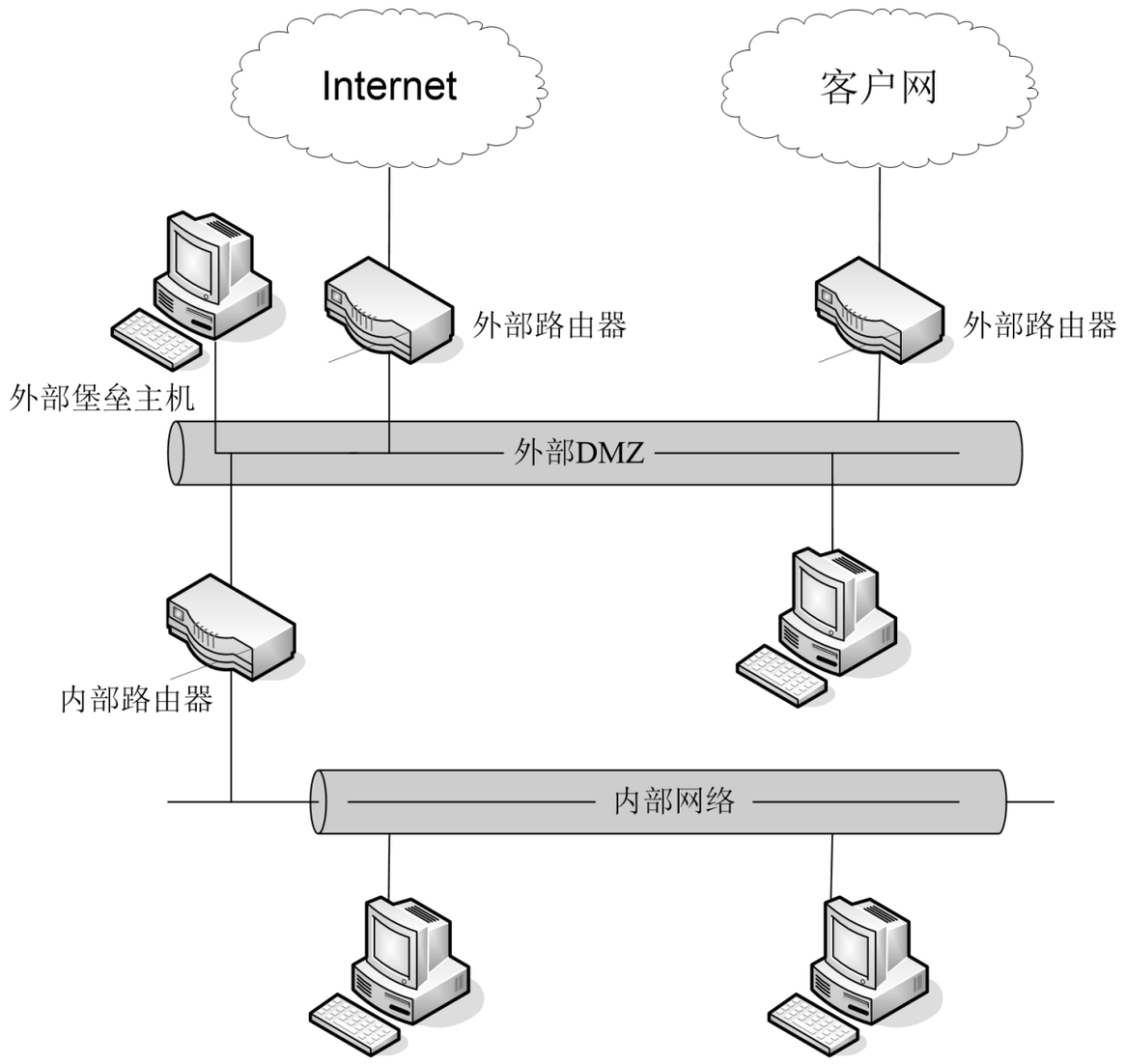


(4) 两个堡垒主机和两个非军事区结构

其它的防火墙结构



(5) 牺牲主机结构



(6)使用多台外部路由器的体系结构

防火墙——

iptables

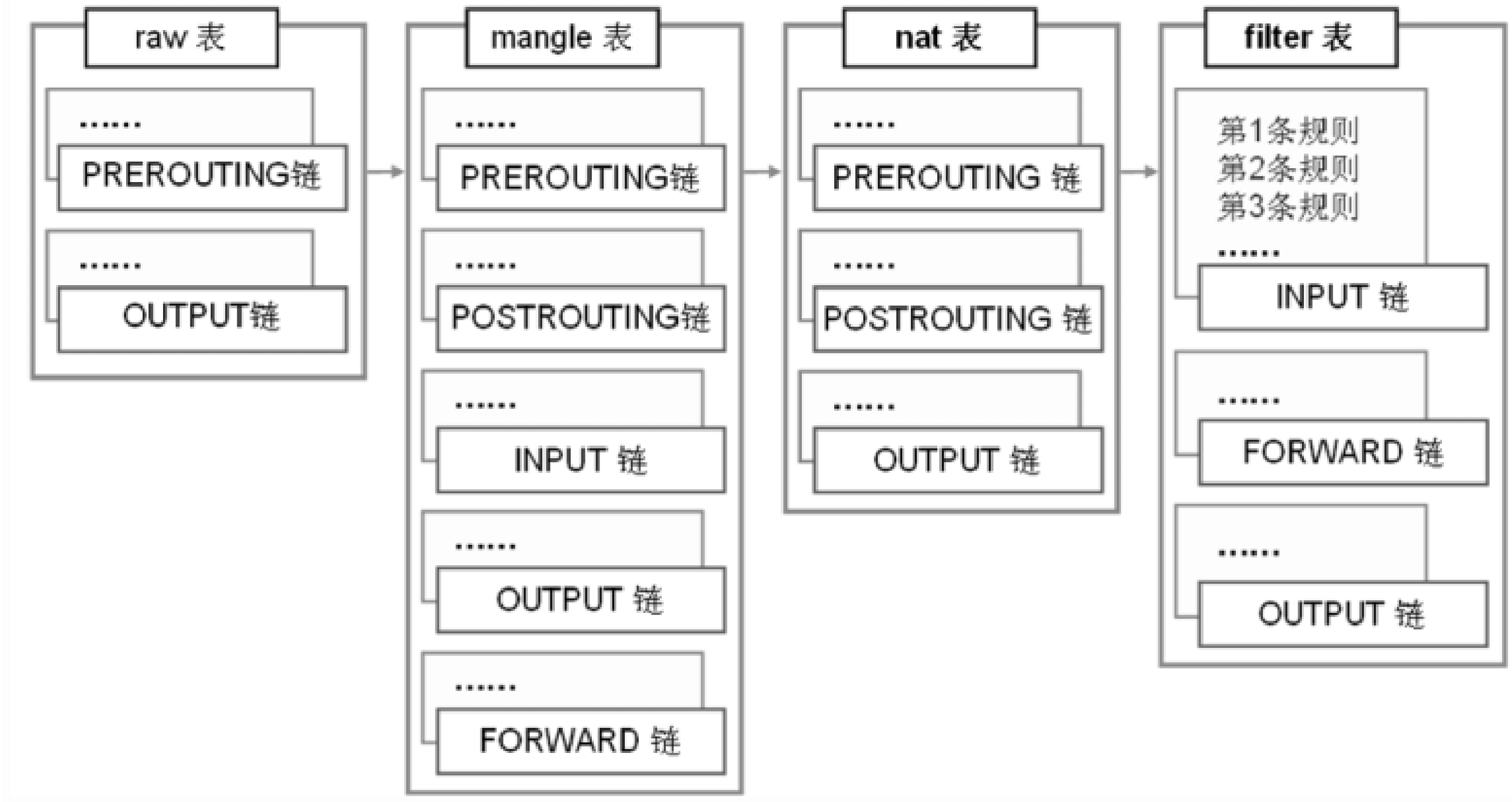
iptables

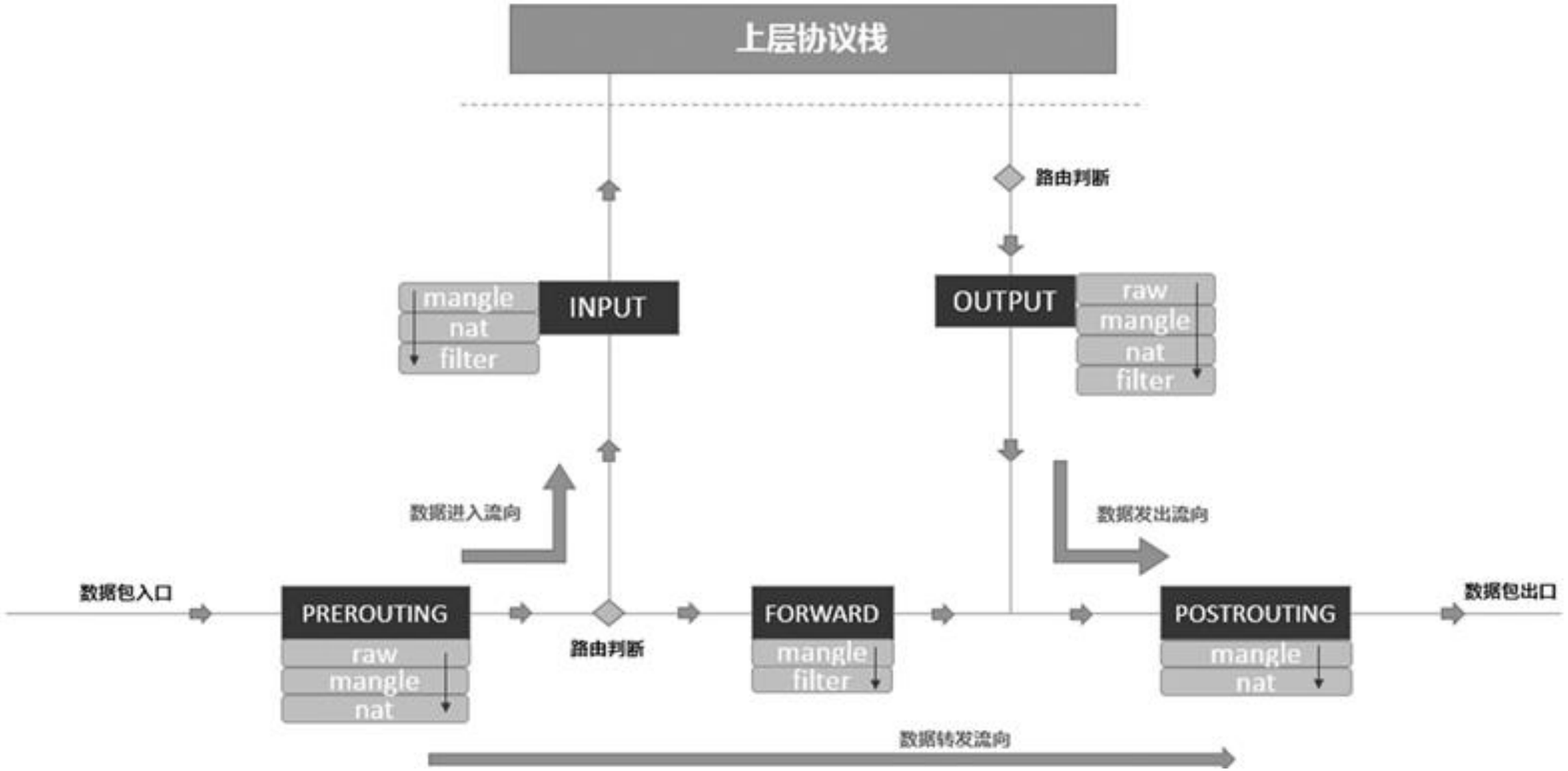
- Iptables是与Linux内核集成的IP信息包过滤系统，它是一种功能强大的工具。Iptables信息包过滤系统由两个组件netfilter和Iptables组成。
- netfilter组件也称为内核空间(kernelspace)，是内核的一部分，由一些信息包过滤表组成，这些表包含内核用来控制信息包过滤处理的规则集，这些规则是在做信息包过滤决定时，防火墙所遵循和组成的规则。

iptables

- 主要的规则表有四个，分别是**mangle表、filter表、nat表和raw表**。mangle表主要用来修改数据包，我们可以改变不同的包及包头的内容，比如TTL，TOS或MARK；filter表是专门过滤包的；nat表的主要用处是网络地址转换，做过NAT操作的数据包的地址就被改变了，当然这种改变是根据我们的规则进行的；raw表主要用作数据跟踪处理。
- 链是数据包传播的路径，每一条链其实就是众多规则中的一个检查清单，每一条链中可以有一条或数条规则。iptables从链中第一条规则开始依次检查，看该数据包是否满足规则所定义的条件。
- 主要的规则链有五个分别为：INPUT——进来的数据包应用此规则链中的策略；OUTPUT——外出的数据包应用此规则链中的策略；FORWARD——转发数据包时应用此规则链中的策略；PREROUTING——对数据包作路由选择前应用此链中的规则；POSTROUTING——对数据包作路由选择后应用此链中的规则。

iptables


raw 表
.....
PREROUTING链
.....
OUTPUT链
mangle 表
.....
PREROUTING链
.....
POSTROUTING链
.....
INPUT 链
.....
OUTPUT 链
.....
FORWARD 链
nat 表
.....
PREROUTING 链
.....
POSTROUTING 链
.....
OUTPUT 链
filter 表
第1条规则
第2条规则
第3条规则
.....
INPUT 链
.....
FORWARD 链
.....
OUTPUT 链



- 1）　当一个数据包进入网卡时，它首先进入PREROUTING链，内核根据数据包目的IP判断是否需要转送出去。
- 2）　如果数据包就是进入本机的，它就会沿着图向下移动，到达INPUT链。数据包到了INPUT链后，任何进程都会收到它。本机上运行的程序可以发送数据包，这些数据包会经过OUTPUT链，然后到达POSTROUTING链输出。
- 3）　如果数据包是要转发出去的，且内核允许转发，数据包就会如图所示向右移动，经过FORWARD链，然后到达POSTROUTING链输出。

防火墙：iptables

#iptables [-AI 链] [-io 网卡接口] [-p 协议] [-s 源ip/网段] [-d 目标ip/网段] -j [ACCEPT/DROP]

- 参数说明：

- -AI 链：针对某条链进行规则的“插入”或“添加”-A：新增加一条规则。-I：插入一条规则。
- 例如原本有四条规则，使用-I插入一条规则后则刚插入的那条变成第一条，原来的四条规则都向后移一位。例如在-I指定了顺序（-I INPUT 3），则插入后，此条规则变成了第三的位置。
- -io 网络接口：-i:进入口　　-o：传出口
- -p 协议：如:tcp/udp/icmp/all
- -s 源IP/网段：设置此规则的数据包来源地址，可以是IP，也可以是一个网段
- -d 目标IP/网段：同-s一样，只是这里指的是目标IP/网段
- -j:后面接操作，主要的操作有接受（ACCEPT）、丢弃（DROP）、记录（LOG）

防火墙：iptables

```
why@ubuntu:~$ sudo iptables -A INPUT -p tcp --dport 23 -j DROP
why@ubuntu:~$ sudo telnet 192.168.91.128
Trying 192.168.91.128...
telnet: Unable to connect to remote host: Connection timed out
```

- 拒绝其他主机访问本地Telenet服务(TCP 23端口)，格式为：
- iptables -A INPUT -p tcp --dport 23 -j DROP
- -A代表append添加规则，后面带INPUT代表这种数据包，-p代表使用的协议为tcp，--dport代表端口为23，-j代表使用DROP还是ACCEPT，DROP为拒绝。

防火墙：iptables

- 规则举例1:
  - 来自192.168.1.0/24可接受，但来自来192.168.1.10的所有包丢弃

```
#iptables -A INPUT -i eth0 -s 192.168.1.10 -j DROP
#iptables -A INPUT -i eth0 -s 192.168.1.0/24 -j ACCEPT
```

- 上述这个范例很重要，因为与顺序有关，要先丢弃192.168.1.10后再允许1.0这个网段。

防火墙：iptables

- 规则举例2:
  - 允许端口（upd port 137,138 tcp port 139,445）数据访问。

```
#iptables -A INPUT -i eth0 -p upd --dport 137:138 -j ACCEPT
#iptables -A INPUT -i eth0 -p tcp --dport 139 -j ACCEPT
#iptables -A INPUT -i eth0 -p tcp --dport 445 -j ACCEPT
```

防火墙：iptables

- use this module to avoid various denial of service attacks (DoS) with a faster rate to increase responsiveness

Syn-flood protection:

iptables -A FORWARD -p tcp --syn -m limit --limit 1/s -j ACCEPT

Furtive port scanner:

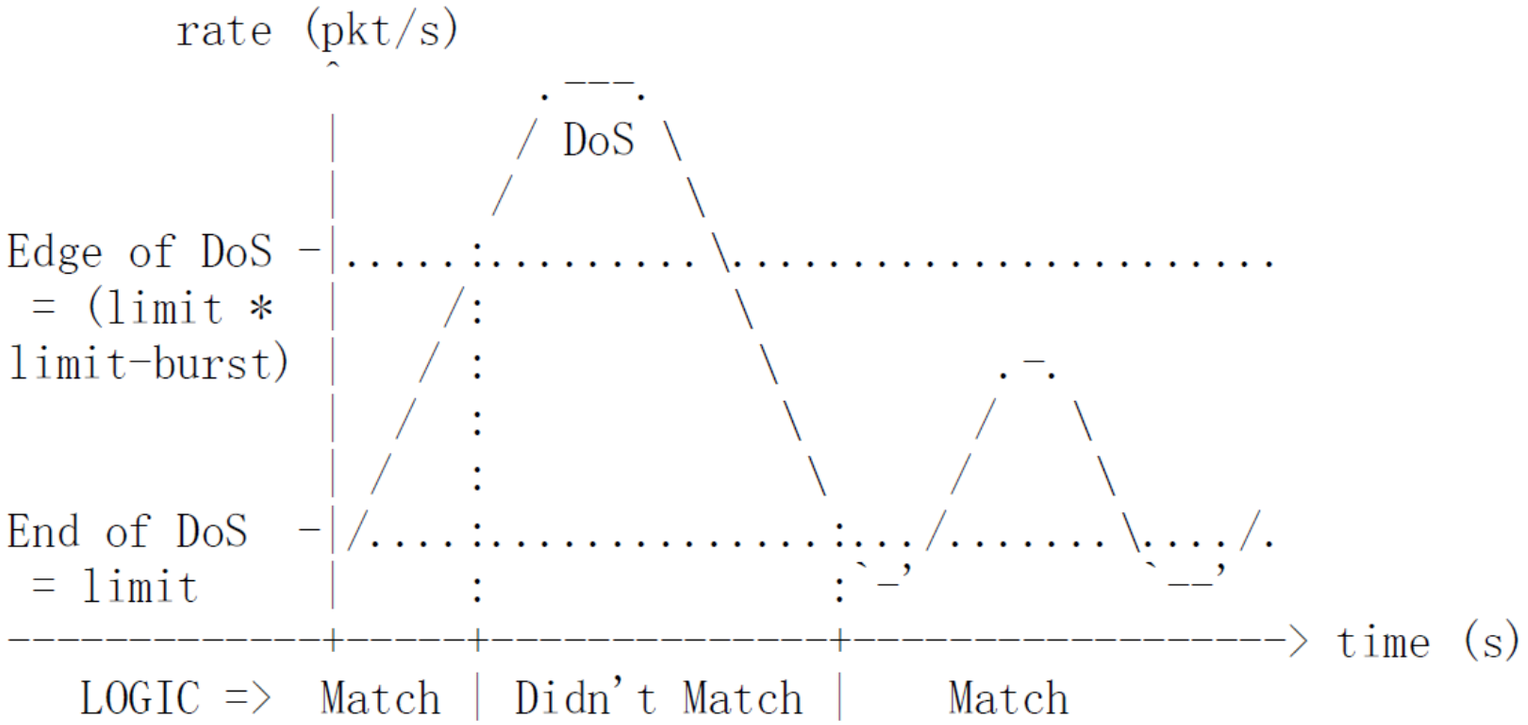
iptables -A FORWARD -p tcp --tcp-flags SYN,ACK,FIN,RST RST -m limit --limit 1/s -j ACCEPT

Ping of death:

iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT

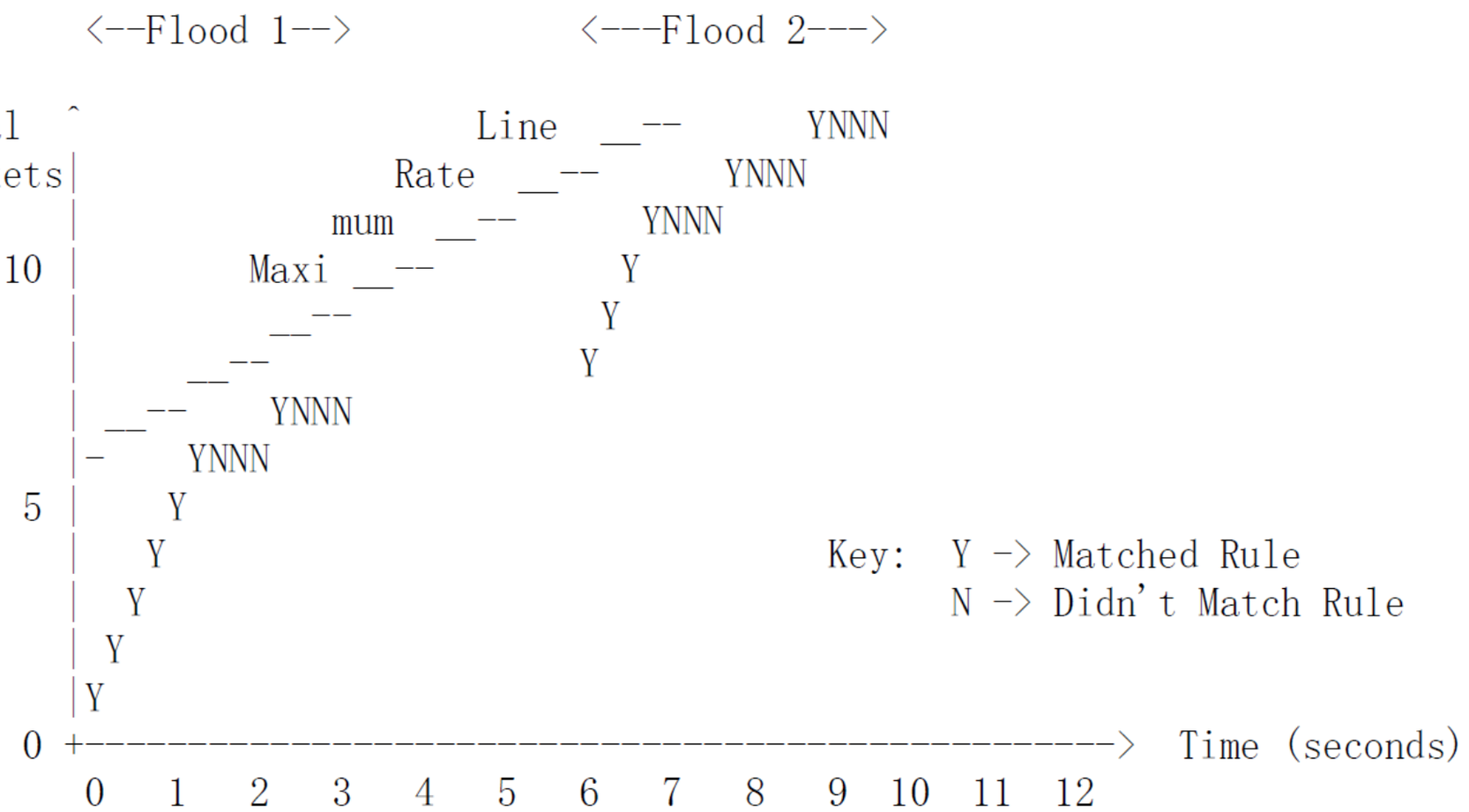
防火墙：iptables

This module works like a "hysteresis door", as shown in the graph below.



Say we say match one packet per second with a five packet burst, but packets start coming in at four per second, for three seconds, then start again in another three seconds.

防火墙：iptables



You can see that the first five packets are allowed to exceed the one packet per second, then the limiting kicks in. If there is a pause, another burst is allowed but not past the maximum rate set by the rule (1 packet per second after the burst is used).

- iptables -A INPUT -p icmp -m limit --limit 10/min --limit-burst 5 -j ACCEPT
- iptables -A INPUT -p icmp -j DROP
- 这两条匹配语句除了针对的协议改变为ICMP，其他部分与上面防御SYN flood攻击的匹配语句相同。

```
$ ping 172.16.135.144
PING 172.16.135.144 (172.16.135.144): 56 data bytes
64 bytes from 172.16.135.144: icmp_seq=0 ttl=64 time=0.388 ms
64 bytes from 172.16.135.144: icmp_seq=1 ttl=64 time=0.405 ms
64 bytes from 172.16.135.144: icmp_seq=2 ttl=64 time=0.396 ms
64 bytes from 172.16.135.144: icmp_seq=3 ttl=64 time=0.421 ms
64 bytes from 172.16.135.144: icmp_seq=4 ttl=64 time=0.450 ms
Request timeout for icmp_seq 5
64 bytes from 172.16.135.144: icmp_seq=6 ttl=64 time=0.914 ms
Request timeout for icmp_seq 7
Request timeout for icmp_seq 8
Request timeout for icmp_seq 9
Request timeout for icmp_seq 10
Request timeout for icmp_seq 11
64 bytes from 172.16.135.144: icmp_seq=12 ttl=64 time=1.547 ms
Request timeout for icmp_seq 13
Request timeout for icmp_seq 14
Request timeout for icmp_seq 15
Request timeout for icmp_seq 16
Request timeout for icmp_seq 17
64 bytes from 172.16.135.144: icmp_seq=18 ttl=64 time=0.362 ms
```

防火墙——

网络隔离技术

两类隔离技术

- 物理隔离的指导思想与防火墙截然不同：防火墙的思路是在保障互联互通的前提下，尽可能安全，而物理隔离的思路是在保证必须安全的前提下，尽可能互联互通。
- 物理隔离技术实质就是一种将内外网络从物理上断开，但保持逻辑连接的信息安全技术。这里，物理断开表示任何时候内外网络都不存在连通的物理连接，逻辑连接表示能进行适度的数据交换。“物理隔离”是指内部网不直接通过有线或无线等任何手段连接到公共网，从而使内部网络和外部公共网络在物理上处于隔离状态的一种物理安全技术。

两类隔离技术

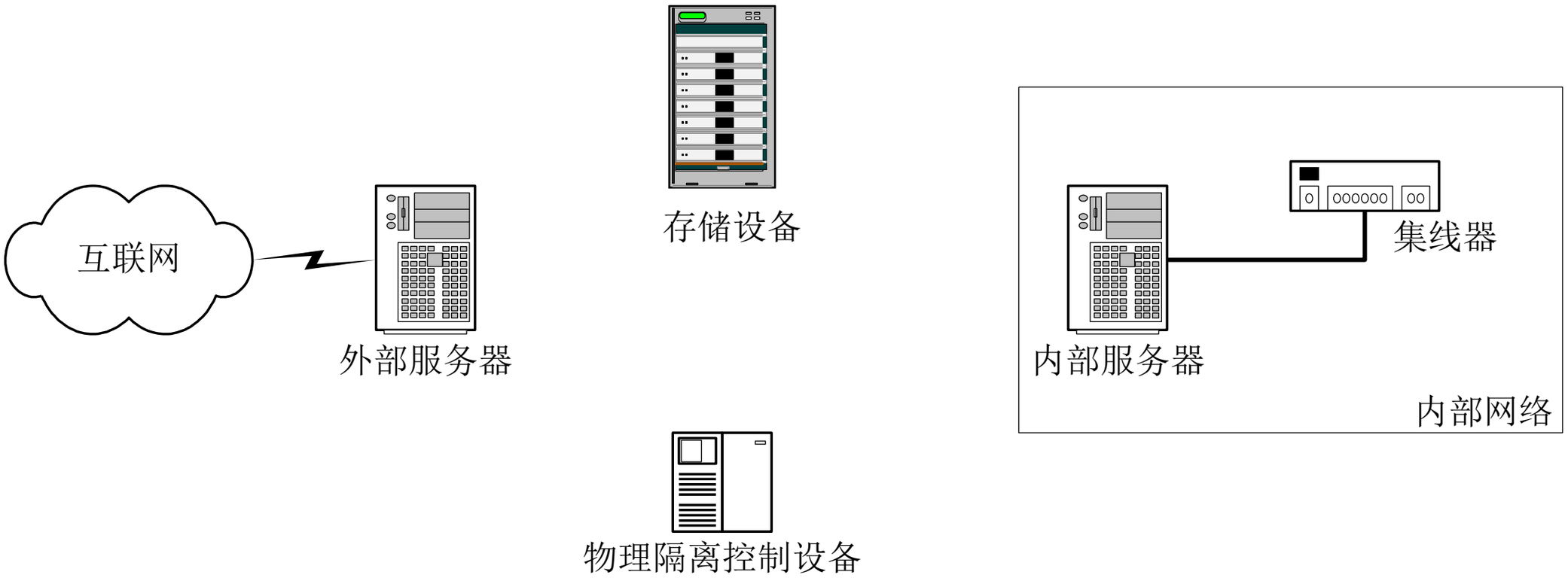
- 物理隔离在安全上的要求主要有三点：
- (1) 在物理传导上使内外网络隔断，确保外部网不能通过网络连接而侵入内部网：同时防止内部网信息通过网络连接泄漏到外部网；
- (2) 在物理辐射上隔断内部网与外部网，确保内部网信息不会通过电磁辐射等方式泄漏到外部网；
- (3) 在物理存储上隔断两个网络环境，对于断电后会遗失信息的部件，如内存、处理器等暂存部件，要在网络转换时做清除处理，防止残留信息泄漏；对于断电非遗失性设备如磁带机、硬盘等存储设备，内部网与外部网信息要分开存储。

网络隔离原理

- 物理隔离实现的原理主要包括以下两类：数据二极管技术和存储池交换技术。
- (1) 数据二极管技术
- 这是一种在物理断开的网络之间进行单向桥接的专用安全技术。这种单向桥接技术是通过单工的连接完成的。这种连接只在数据源计算机具有一个数据发送源，在数据目标计算机上具有一个数据接收器。这种设计也被人们称为对外部“完全不信任”设计。
- (2) 存储池交换技术
- 这也是一种桥接隔离网络之间连接的专用安全技术。这种技术使用一个可交换方向的电子存储池。存储池每次只能与内外网络的一方相连。通过内外网络向存储池拷贝数据块和存储池的摆动完成数据传输。这种技术实际上是一种数据镜像技术。这种技术在实现内外网络数据交换的同时，保持了内外网络的物理断开。

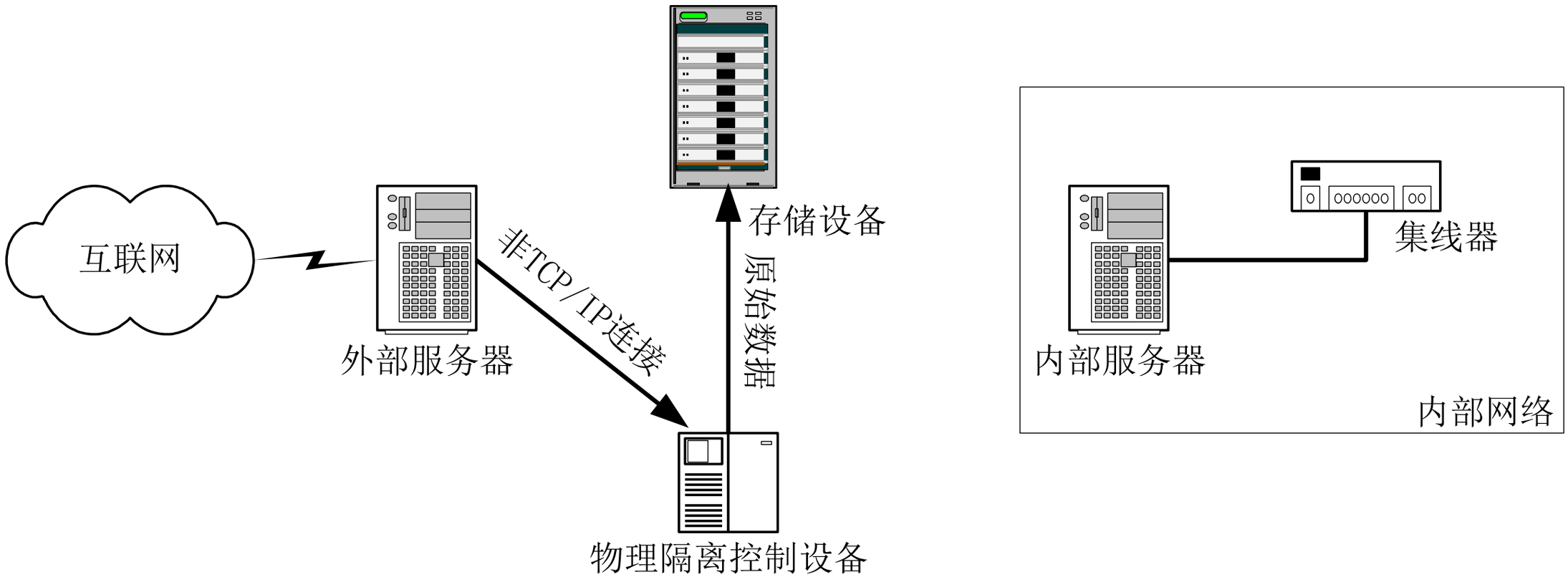
网络隔离原理

- 一个典型的物理隔离方案（处于完全隔离状态）



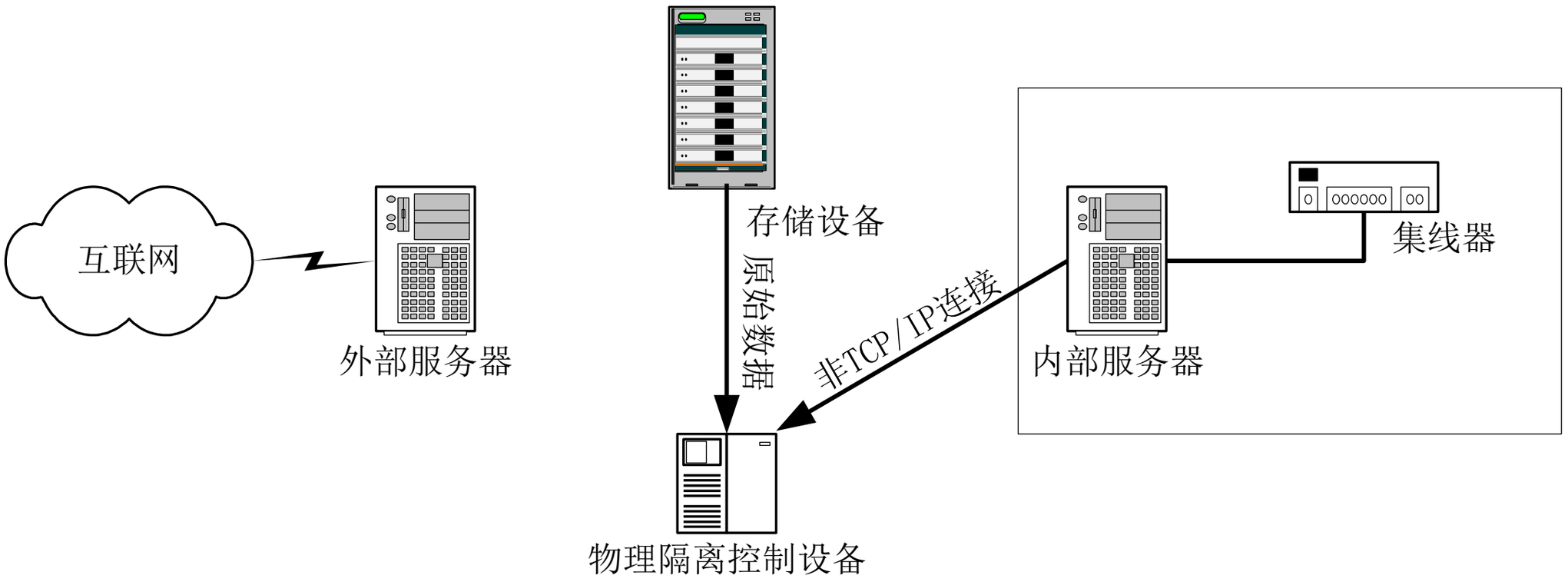
网络隔离技术

- 一个典型的物理隔离方案（隔离设备处于与外网相连状态）



网络隔离技术

- 一个典型的物理隔离方案（隔离设备处于与内网相连状态）



网络隔离技术分类

- 第一代隔离技术：完全的隔离。

➢ 此方法使得网络处于信息孤岛状态，做到了完全的物理隔离，需要至少两套网络和系统，更重要的是信息交流的不便和成本的提高，这样给维护和使用带来了极大的不便。

- 第二代隔离技术：硬件卡隔离。

➢ 在客户端增加一块硬件卡，客户端硬盘或其他存储设备首先连接到该卡，然后再转接到主板上，通过该卡能控制客户端硬盘或其他存储设备。而在选择不同的硬盘时，同时选择了该卡上不同的网络接口，连接到不同的网络。相比之下，单硬盘隔离卡更为先进，其实现原理是将原计算机的单个硬盘从物理层上分割为公共和安全两个分区，安装两套操作系统，从而实现内外网的安全隔离。用户可以根据自己的需要在不同的网络环境（内网或外网）中自由切换，操作时感受不到任何区别。但是，不管哪种隔离卡技术，仍然需要网络布线为双网线结构，这样势必存在着较大的安全隐患。

网络隔离技术

- 第三代隔离技术：数据转播隔离。

➢ 利用转播系统分时复制文件的途径来实现隔离，切换时间非常之久，甚至需要手工完成，不仅明显地减缓了访问速度，更不支持常见的网络应用，失去了网络存在的意义。

- 第四代隔离技术：空气开关隔离。

➢ 它是通过使用单刀双掷开关，使得内外部网络分时访问临时缓存器来完成数据交换的，但在安全和性能上存在有许多问题。

- 第五代隔离技术：安全通道隔离，又称为网闸技术。

➢ 此技术通过专用通信硬件和专有安全协议等安全机制，来实现内外部网络的隔离和数据交换，不仅解决了以前隔离技术存在的问题，并有效地把内外部网络隔离开来，而且高效地实现了内外网数据的安全交换，透明支持多种网络应用，成为当前隔离技术的发展方向。

网络隔离技术应用

- 客户端的物理隔离
- 集线器级的物理隔离

➢ 集线器级的物理隔离产品需要与客户端的物理隔离产品结合起来应用，可以在客户端的内外双网的布线上使用一条网络线来通过远端切换器连接内外双网，实现一台工作站连接内外两个网络的目的，并在网络布线上避免了客户端计算机要用两条网络线连接网络。

- 服务器端的物理隔离

➢ 服务器端的物理隔离技术是一种崭新的高级隔离技术，现在国外的产品已经应用，它通过复杂的软硬件技术实现了在服务器端的数据过滤和传输任务。第四、第五代物理隔离技术都是基于服务器的产品。

问题和讨论