



网络攻防技术与实践课程

课程6. 网络安全防护技术

诸葛建伟

zhugejw@gmail.com



内容

1. 安全模型-P2DR模型

2. P: 防御技术

3. D: 检测技术

4. R: 响应技术

5. 作业6: 分析蜜网网关中防火墙与入侵检测系统配置规则



信息安全技术与安全模型的发展

- **COMSEC – 通信安全**
 - 保护军事等机密信息，机密性(**Confidentiality**)
 - 专门针对机密性的**BLP(Bell-La Padula)**多级安全策略模型
- **COMPSEC -计算机安全**
 - 完整性(**Integrity**)也被纳入了核心安全属性
 - 引入身份认证、访问控制技术
 - 针对完整性保护的**Biba**模型和**Clark-Wilson**模型
- **NETSEC - 网络安全**
 - 网络信息服务的可用性(**Availability**)也上升成为核心的安全属性
- **IA – 信息保障(NSA)**
 - 机密性、完整性、可用性、真实性、不可抵赖性
 - 信息保障体系；纵深安全防护体系



安全评估模型与标准

□ 安全评估

- 评估信息系统是否能够满足特定的安全需求和属性

□ 安全评估模型

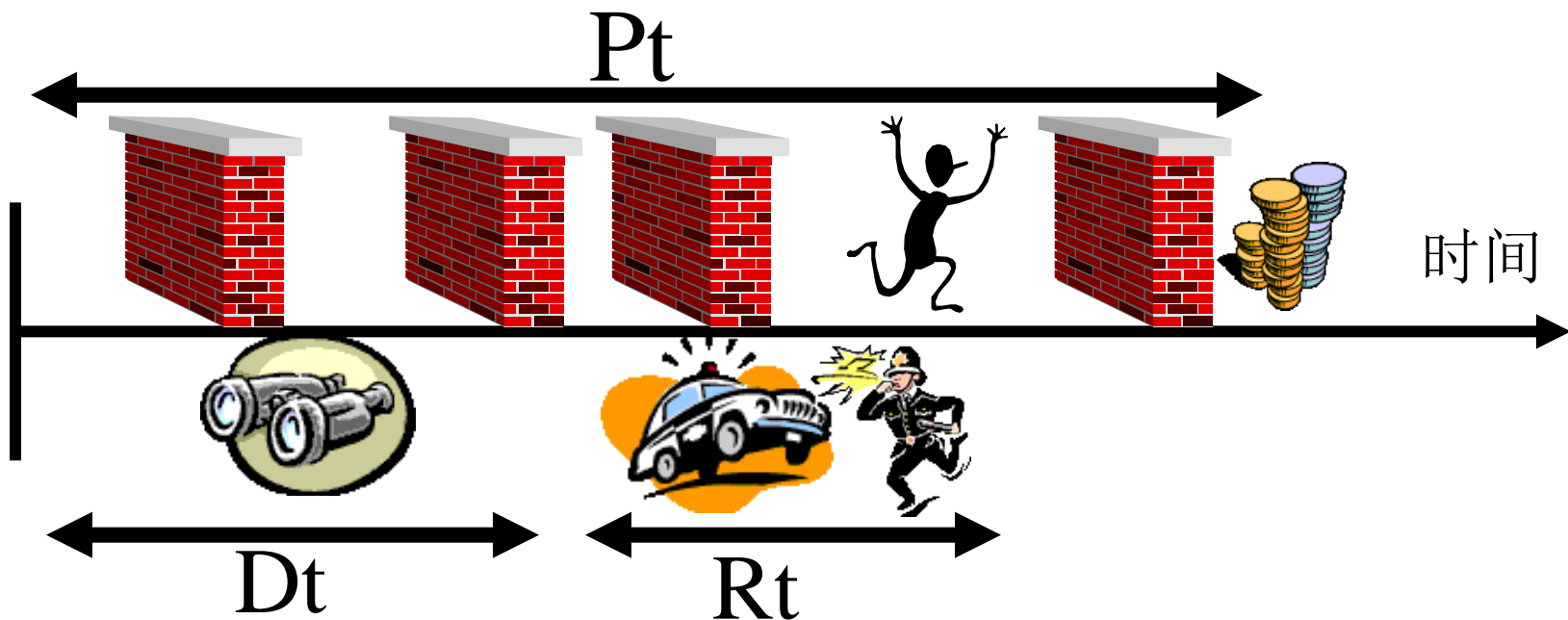
- **1985**: 美国 可信计算机安全评估准则**TCSEC**《桔皮书》
 - 分级评估: **A\B\C\D**
- **199x**: 欧洲 **ITSEC**安全测评标准
- **1999**: **Common Criteria(CC)**标准
- **1999**: **GB17859**《计算机信息系统安全保护等级划分标准》

□ 静态安全模型 **VS.** 动态安全模型(可适应安全模型)

PDR安全模型

□ PDR: 基于时间的安全(Time-based Security)

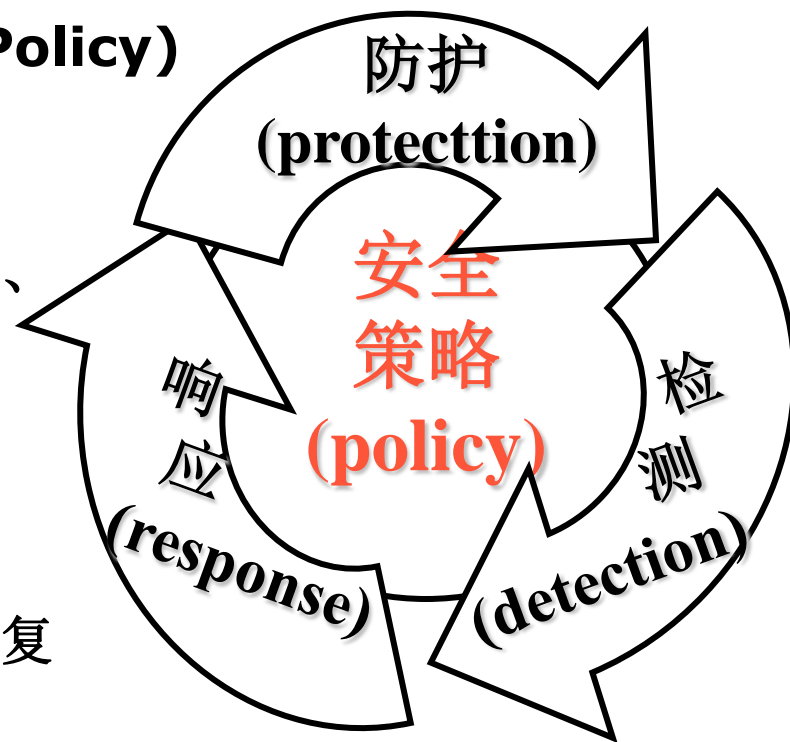
- 可以量化，可以计算
- 防御延缓攻击时间，及时的检测和响应



网络安全不等式: $P_t > D_t + R_t$

P²DR安全模型

- 网络安全是相对的，没有绝对的安全
- 这是一个动态模型
 - 根据风险分析制定安全策略(**Policy**)
 - **PDR**构成动态闭环
- **P**: 执行安全防护策略
 - 防火墙、身份认证、访问控制、加密
- **D**: 实时检测
 - 漏洞评估、入侵检测
- **R**: 实时响应
 - 应急响应、备份恢复、灾难恢复





内容

1. 安全模型-P2DR模型

2. P: 防御技术

3. D: 检测技术

4. R: 响应技术

5. 作业6: 分析蜜网网关中防火墙与入侵检测系统配置规则



防御技术

- 网络防御 - 边界网络安全设备
 - 网络访问控制：防火墙，**VPN**
 - 网络内容控制：**SCM**
 - **IPS**（入侵防御系统），**IMS**(入侵管理系统), **UTM**（统一威胁管理）
- 主机防御
 - 漏洞扫描和补丁管理
 - 个人防火墙
 - 防病毒软件
 - 系统诊断与恢复软件
- 安全产业“老三样”：防火墙、入侵检测、防病毒
- 安全产业“新三样”：安全管理平台、安全服务、个人安全防御



防火墙(Firewall)

□ 防火墙

- 建筑工程学领域，指的一类由防火材料制成的、在分割的建筑单元之间减缓火灾蔓延的防火屏障墙
- 计算领域中的防火墙
 - 置于不同的网络安全域之间，对网络流量或访问行为实施访问控制的安全组件或设备
 - 一种网络访问控制机制
 - 大楼“门卫”

□ 技术关键特性

- 只能对流经的网络数据进行检查控制：边界部署
- 不具备主动检测网络攻击数据能力，需合理设计安全控制策略
- 并非“一劳永逸”的“安全最终解决方案”



防火墙功能

- 在网络协议栈的各个层次上实施网络访问控制机制
 - 网络层：包过滤
 - 传输层：电路级代理
 - 应用层：应用层代理/网关
- 基本功能：控制在计算机网络中不同信任程度网络域间传送的数据流
 - 检查控制进出网络的网络流量
 - 防止脆弱或不安全的协议和服务
 - 防止内部网络信息的外泄
 - 对网络存取和访问进行监控审计
 - 防火墙可以强化网络安全策略并集成其他安全防御机制



防火墙的不足

- 作为网络边界防护机制而先天无法防范的安全威胁
 - 来自网络内部的安全威胁
 - 通过非法外联的网络攻击
 - 计算机病毒传播
- 由于技术瓶颈问题目前还无法有效防范的安全威胁
 - 针对开放服务安全漏洞的渗透攻击
 - 针对网络客户端程序的渗透攻击
 - 基于隐蔽通道进行通信的特洛伊木马或僵尸网络



防火墙技术类型

- 包过滤防火墙 (**packet filter**)
 - **1988, DEC**
 - 网络包粒度, 工作在网络层, 主要实现形式为路由器**ACL**
- 状态防火墙 (**stateful firewall**)
 - **1980s底, AT&T Bell**
 - 网络会话粒度, 工作在传输层
 - 目前防火墙最主要实现方式
- 电路级代理技术
- 应用层代理防火墙 (**application layer firewall**)
 - **Paper: 1990 Purdue, AT&T**
 - **Product: 1991 DEC**
 - 应用层代理, 工作在应用层



包过滤防火墙

□ 基本的思想很简单

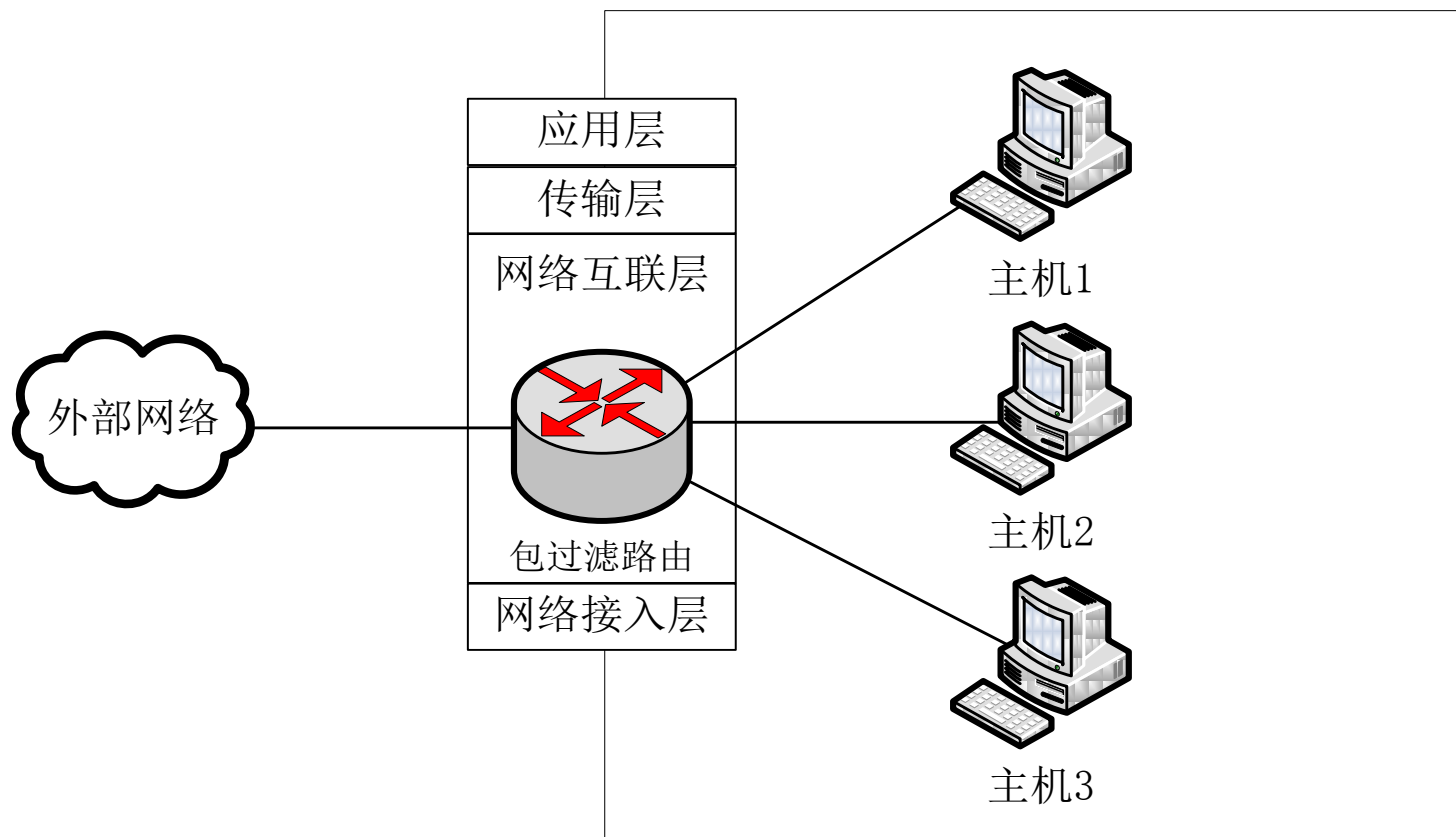
- 对于每个进来的包，适用一组规则，然后决定转发或者丢弃该包
- 往往配置成双向的

□ 如何过滤

- 过滤的规则以**IP**和传输层的头中的域(字段)为基础，包括源和目标**IP**地址、**IP**协议域、源和目标端口号
- 过滤器往往建立一组规则，根据**IP**包是否匹配规则中指定的条件来作出决定。
 - 如果匹配到一条规则，则根据此规则决定转发或者丢弃
 - 如果所有规则都不匹配，则根据缺省策略

包过滤防火墙示意图

网络安全边界





包过滤防火墙

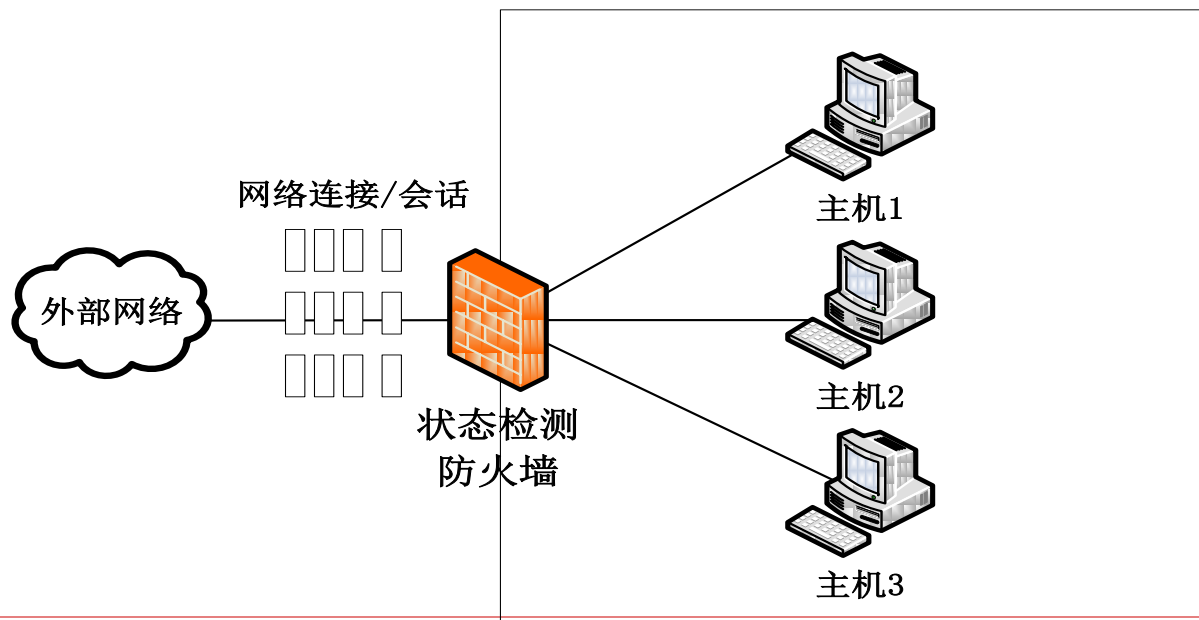
- 通常在路由器上实现
 - 实际上是一种网络层的访问控制机制
 - 路由器**ACL**机制
- 在网络层上进行监测
 - 仅仅根据数据包自身包含的信息(协议头部)进行检查和过滤
 - 并没有考虑连接状态信息
- 优点:
 - 实现简单
 - 对用户透明
 - 效率高
- 缺点:
 - 正确制定规则并不容易
 - 不可能引入认证机制

状态防火墙

□ 状态防火墙

- 跟踪网络会话(连接)状态, 判断报文合法性
- 在网络会话粒度上匹配和实施防火墙规则
- 特性: 状态报文检查(**SPI: stateful packet inspection**)

网络安全边界





状态防火墙(2)

- 状态防火墙机制
 - 跟踪和维护网络连接状态信息(**CT: connection table**)
 - **TCP**网络连接
 - **SYN包: NEW connections**
 - 经过三次握手: **ESTABLISHED connections**
 - **UDP**会话
 - 一般处理**UDP**包时, 马上设置为**ESTABLISHED**
 - 在网络访问配置规则中支持对状态的匹配
- 目前防火墙产品的主流应用技术
 - 国外厂商: **Check Point/Cisco/FORTINET/Juniper...**
 - 国内厂商: 天融信/联想/方正/...
 - 开源软件: **Netfilter*/IPTables* (Linux)**



代理(proxy)技术

- 代理(proxy)实际上也是一种安全防护技术
 - 允许客户端通过代理与网络服务进行非直接的连接
 - 在代理服务器上可以进行访问控制和内容检查

- 不同类型的代理技术
 - 应用层：应用层代理 (**HTTP代理**)
 - 传输层：电路级代理 (**Socks代理**)
 - 网络层：**NAT代理** (**NAT网关**、拨号上网路由器)

应用层代理

□ 应用层代理

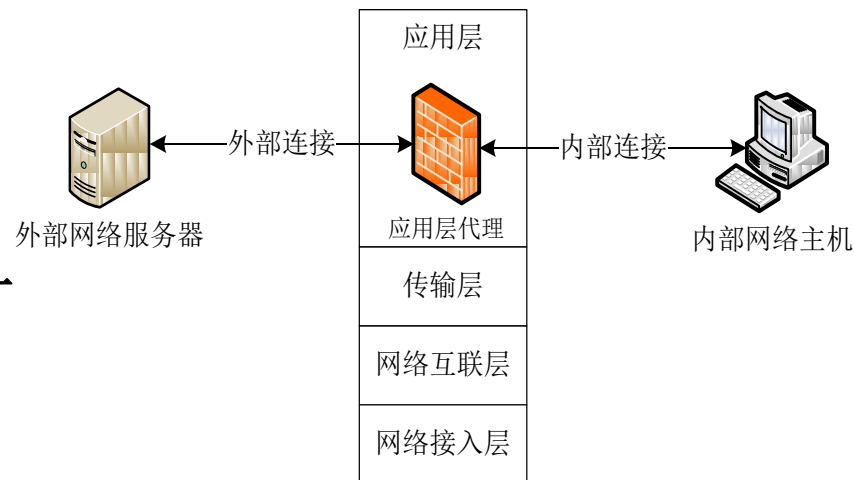
- 也称为应用层网关、代理服务器
- 特定应用层网络服务(**HTTP/Email...**)
- **MSP – Microsoft Proxy Server、Squid**

□ 应用层代理优势

- 隐藏内部网络信息
- 通讯中转，严格内容审查
- 存储转发机制，在线审计
- 用户级身份认证机制

□ 应用层代理不足

- 不通用、不透明、处理速度较慢、部署代价较高





Squid

□ 关于Squid

- 是一个功能全面的**Web Proxy Cache**
- 可以运行在各种**UNIX**版本上
- 是一个自由软件，而且源码开放
- 功能强大，除了**http**协议之外，还支持许多其它的协议，如**ftp**、**ssl**、**DNS**等代理服务

□ 管理和配置

- **/usr/local/squid /etc/squid.conf**
- 默认端口**3128**

□ 一种使用方案

- **Linux+Squid**

电路级代理

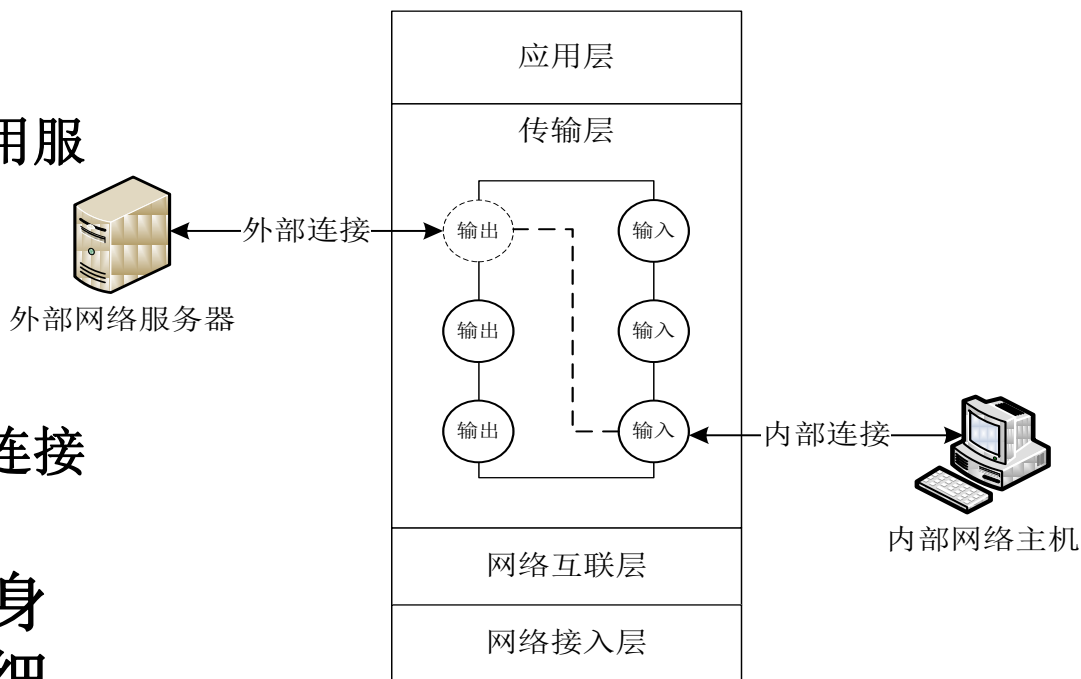
□ 电路级代理

- **Socks**代理
- 工作在传输层
- 同时为多种不同的应用服务提供支持

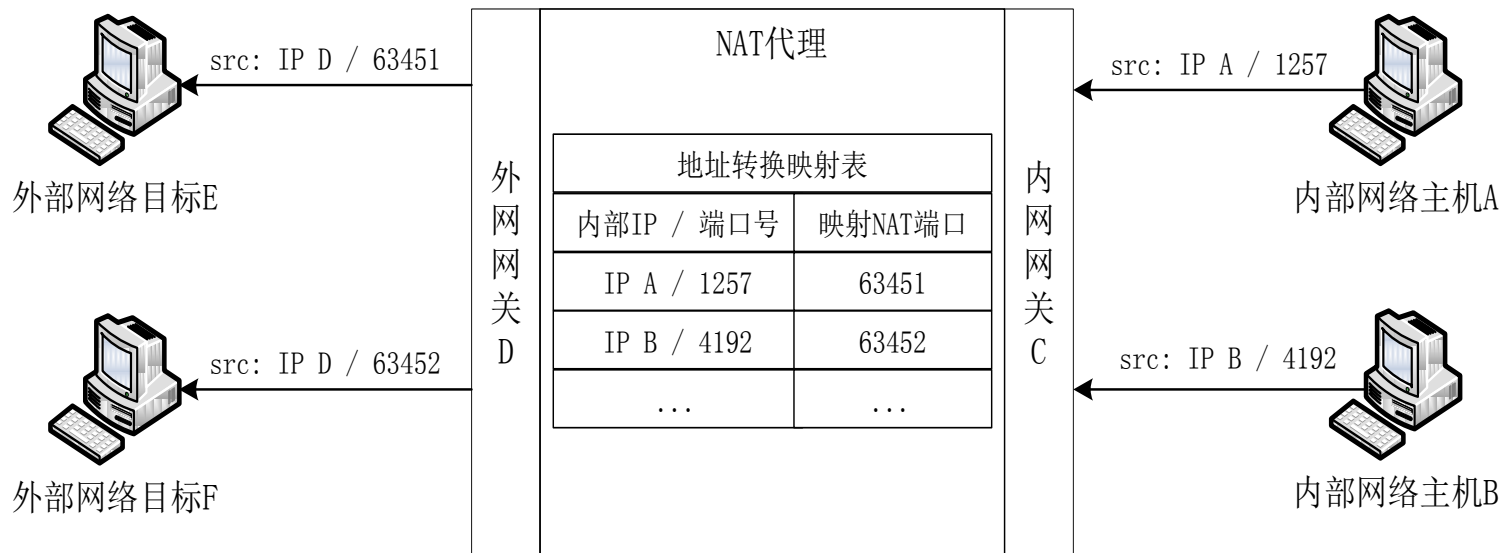
□ 工作机制

- **TCP**层中继
- 建立外部连接，并在连接会话间转发数据

□ 差异：通用、用户级身份认证，但无法进行细致内容审查



NAT代理-网络地址转换



□ NAT(网络地址转换)

- 允许多个用户分享少量或单一的**IP**地址（源**NAT**）
- 允许将网络服务映射到内部服务网络**IP**和端口（目的**NAT**）

□ NAT代理优势

- 方便：任意使用私有网段**IP**地址，无需申请，无冲突
- 安全：对外隐藏内部网络信息

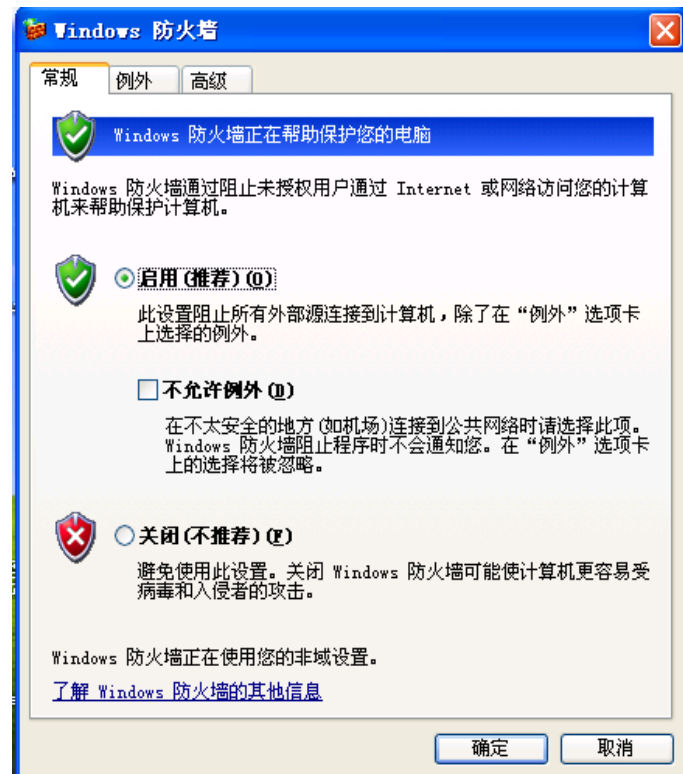
防火墙产品

□ 防火墙产品类别

- 集成包过滤功能的路由器
- 基于通用操作系统的防火墙软件产品
 - 软件防火墙
- 基于安全操作系统的防火墙
 - 软件防火墙+硬件Box
- 硬件防火墙设备
 - 硬件防火墙

□ 个人防火墙

- Windows个人防火墙
- 天网防火墙
- 360安全卫士/瑞星





防火墙部署方法

- 包过滤路由器
- 双宿主堡垒主机
- 屏蔽主机
- 屏蔽子网
- 几个基本概念
 - 堡垒主机(**Bastion Host**): 对外部网络暴露, 同时也是内部网络用户的主要连接点
 - 双宿主主机(**dual-homed host**): 至少有两个网络接口的通用计算机系统
 - **DMZ(Demilitarized Zone, 非军事区或者停火区)**: 在内部网络和外部网络之间增加的一个子网

包过滤路由器部署方案

□ 包过滤防火墙功能的路由器

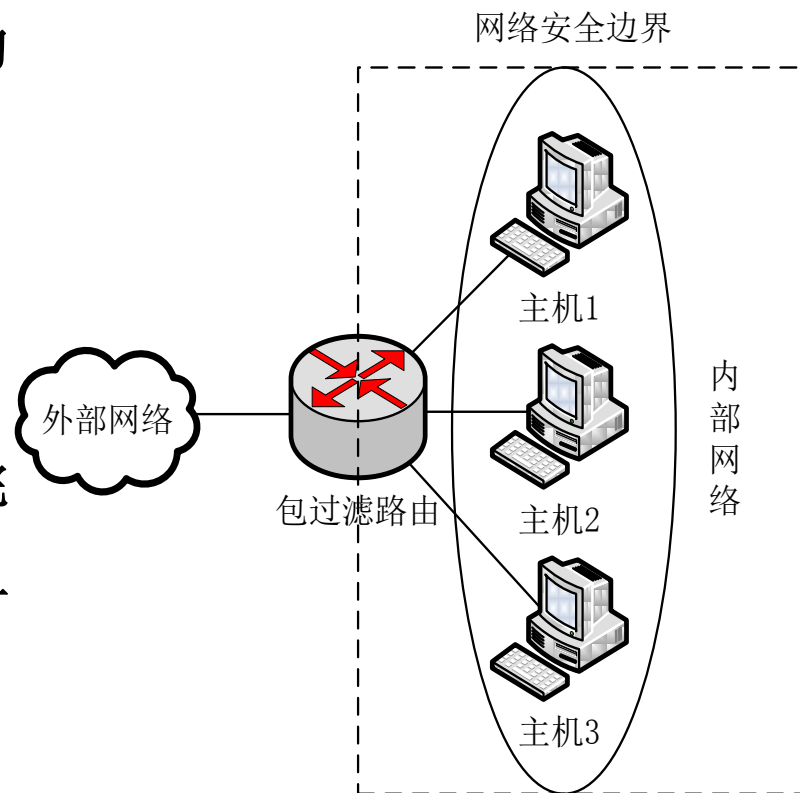
- 内部网络和外部网络之间的唯一连接点
- 路由+**ACL**

□ 优势

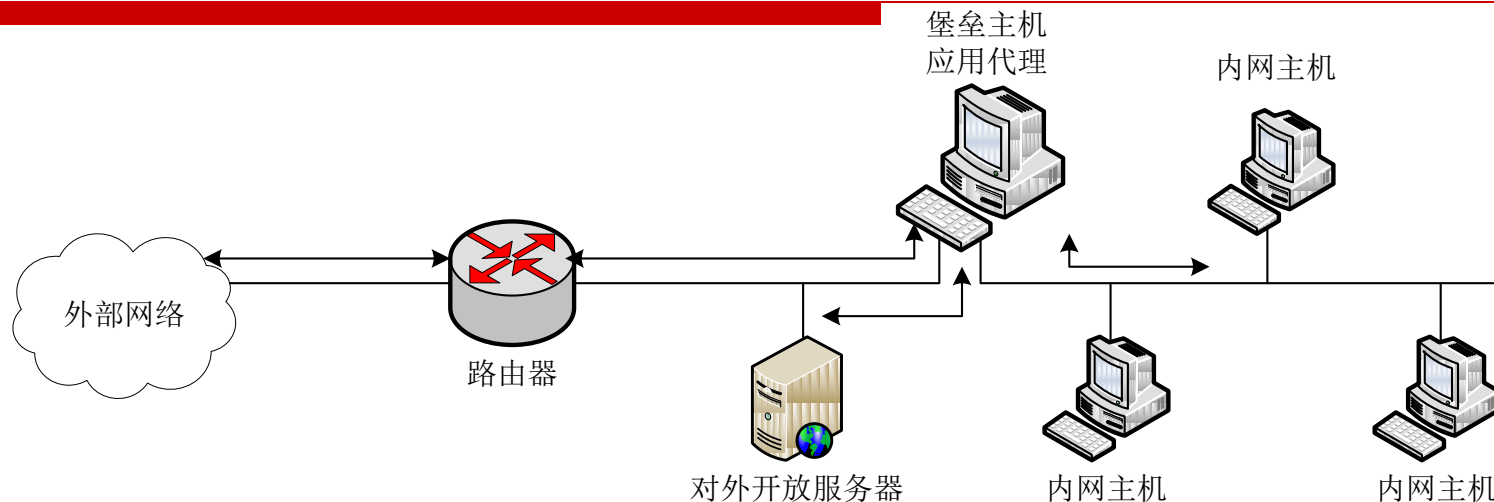
- 成本低、易于使用

□ 缺点

- 一旦路由器被攻破，内网完全暴露
- 内部网络信息对外公开，可攻击开放的主机和服务

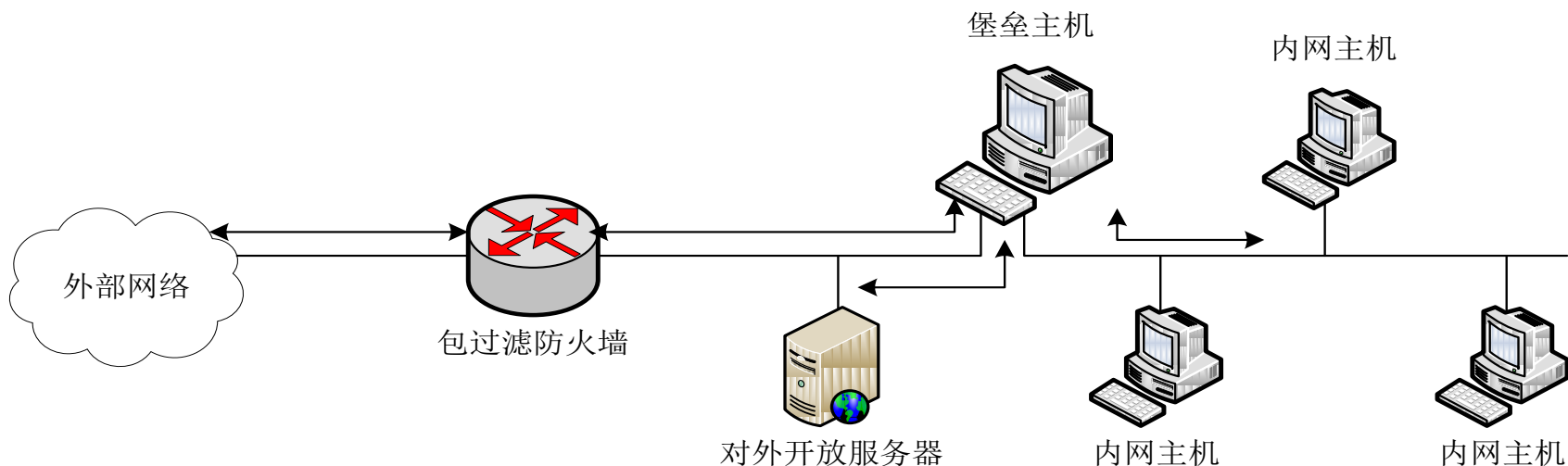


双宿主堡垒主机部署方案



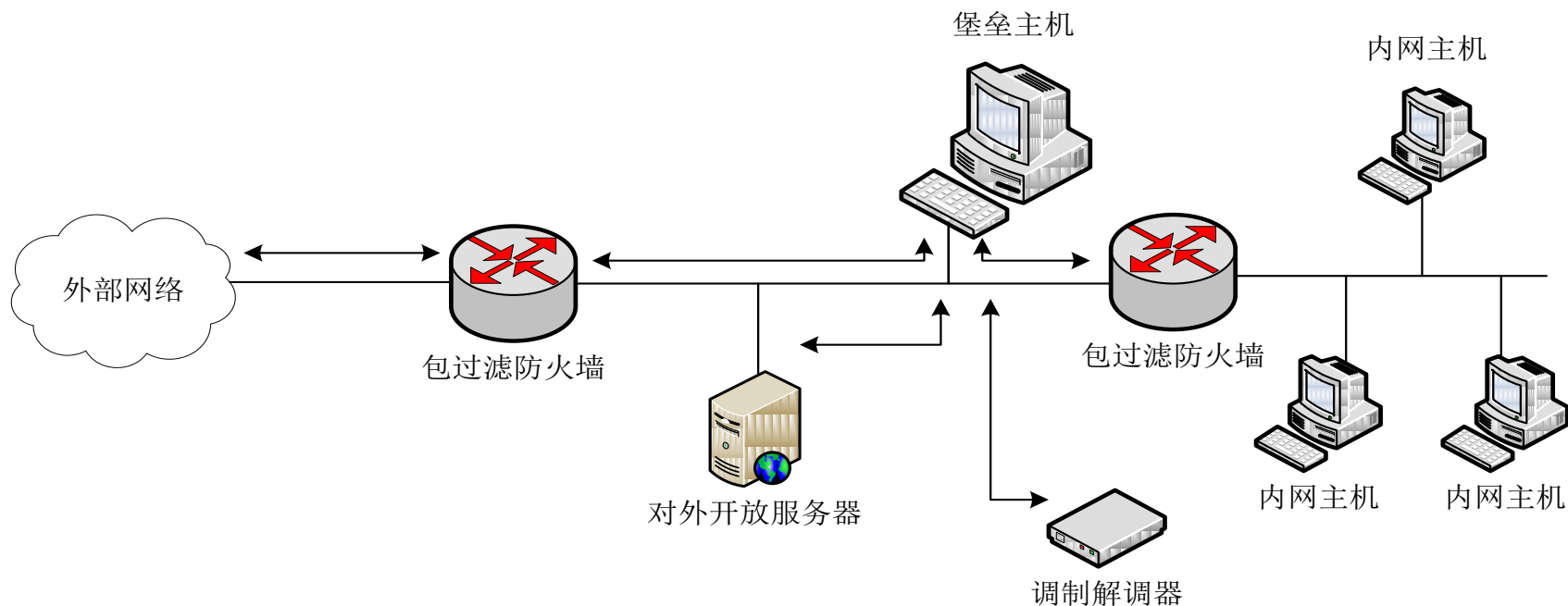
- 使用应用代理网关作为双宿主堡垒主机
 - 一个使用公网**IP**地址连接外部网络
 - 一个使用私有**IP**地址连接内部网络
 - 由应用代理服务器程序为特定的网络应用提供代理
- 优点：对外屏蔽内网信息、用户级身份认证和行为审计
- 缺点：内网对外访问控制过于严格、堡垒主机安全差、，一旦堡垒主机被攻破，内网也将全面地暴露

屏蔽主机部署方案



- 结合包过滤防火墙和应用层代理
 - 两层安全防护
 - 包过滤防火墙：网络层的访问控制
 - 应用层代理堡垒主机：进行应用安全控制
- 优势：双重安全可靠设计；
- 缺点：对外开放服务器成弱点

屏蔽子网部署方案



□ 屏蔽子网：**DMZ**区

- 应用代理及对外服务器
- 三层安全防护：外网防火墙、应用层代理、内网防火墙



Linux中的开源防火墙

□ Netfilter

- **Linux**内核中的防火墙模块

- **Netfilter**特性

 - 包过滤->状态报文检查

 - 灵活可扩展框架, 支持**NAT**网络地址转换, 提供多层**API**接口以支持第三方扩展

- **Netfilter**功能

 - 构建防火墙, **NAT**共享上网, 利用**NAT**构建透明代理, 构建**QoS**或策略路由器, ...

□ IPTables

- **Linux**应用层的防火墙配置工具

- **ipfwadm(2.0.x)→ipchains(2.2.x)→iptables(2.4.x/2.6.x)**

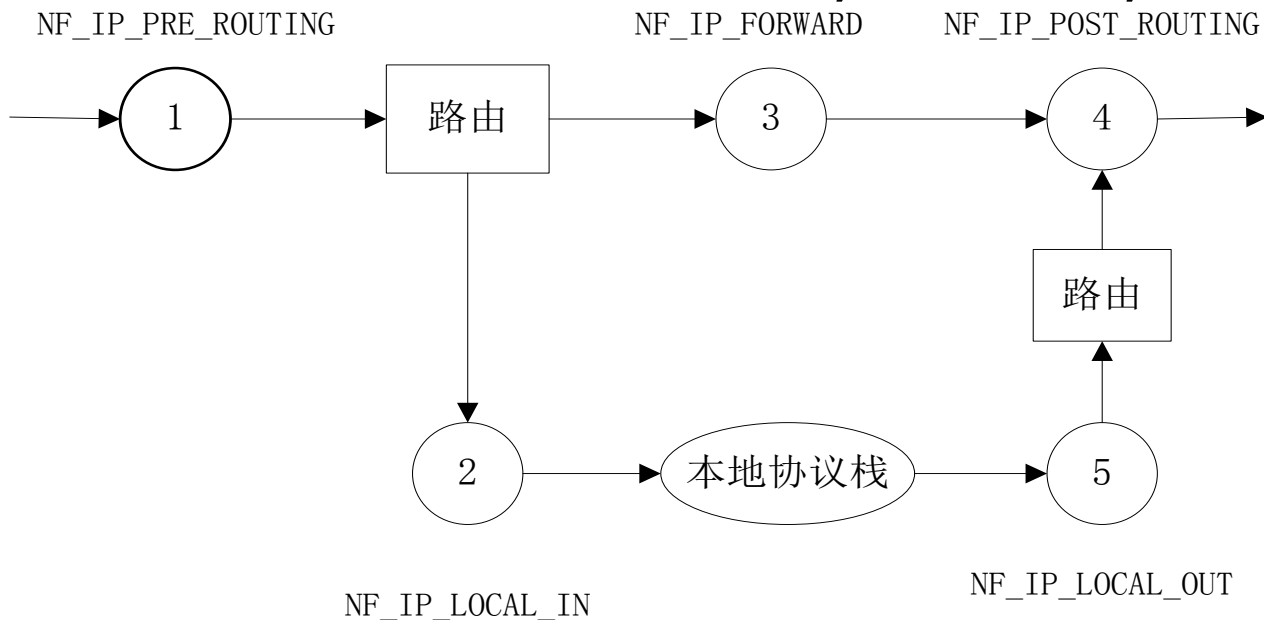


Netfilter/iptables的缺省规则表/链

规则表	规则链	规则处理对象	处理动作
filter (包过滤)	INPUT	发送给本地协议栈数据包	过滤、接受等
	OUTPUT	本地协议栈发出的数据包	过滤、接受等
	FORWARD	路由转发的数据包	过滤、接受等
nat (NAT代理)	PREROUTING	未经路由选择数据包	DNAT、NAPT
	POSTROUTING	已经路由选择数据包	SNAT等
	OUTPUT	本地协议栈发出即将路由数据包	本地数据包DNAT等
mangle (特殊数据包修改)	以上五个规则链	所有数据包	特殊目的的数据包头部修改

Netfilter在Linux协议栈中的hook检查点

- ❑ **PREROUTING:** 进入防火墙数据包, 路由转发前, 实现NAPT/DNAT
- ❑ **LOCAL INPUT:** 发往本地协议栈的数据包, 实现本地安全防护
- ❑ **FORWARD:** 经过防火墙转发的数据包, 实现网络流状态过滤
- ❑ **LOCAL OUTPUT:** 从本地协议栈发出的数据包, 限制对外访问
- ❑ **POSTROUTING:** 从防火墙发出数据包, 路由转发后, 实现SNAT





Netfilter的检查链和处理策略

□ Netfilter检查链

- 通过防火墙转发流量: **PREROUTING→FORWARD→POSTROUTING**
- 传入防火墙本机流量: **PREROUTING→INPUT**
- 防火墙本机传出流量: **OUTPUT→POSTROUTING**

□ Netfilter处理策略

- **ACCEPT:** 允许数据包经过网络协议栈
- **DROP:** 静默地丢弃数据包
- **QUEUE:** 通过**nf_queue**机制将数据包传送至应用层供上层应用处理
- **STOLEN:** 保持数据包直到特定条件后处理, 用于处理**IP**分片等
- **REPEAT:** 使得数据包重新进入**hook**点



Netfilter的报文状态检查

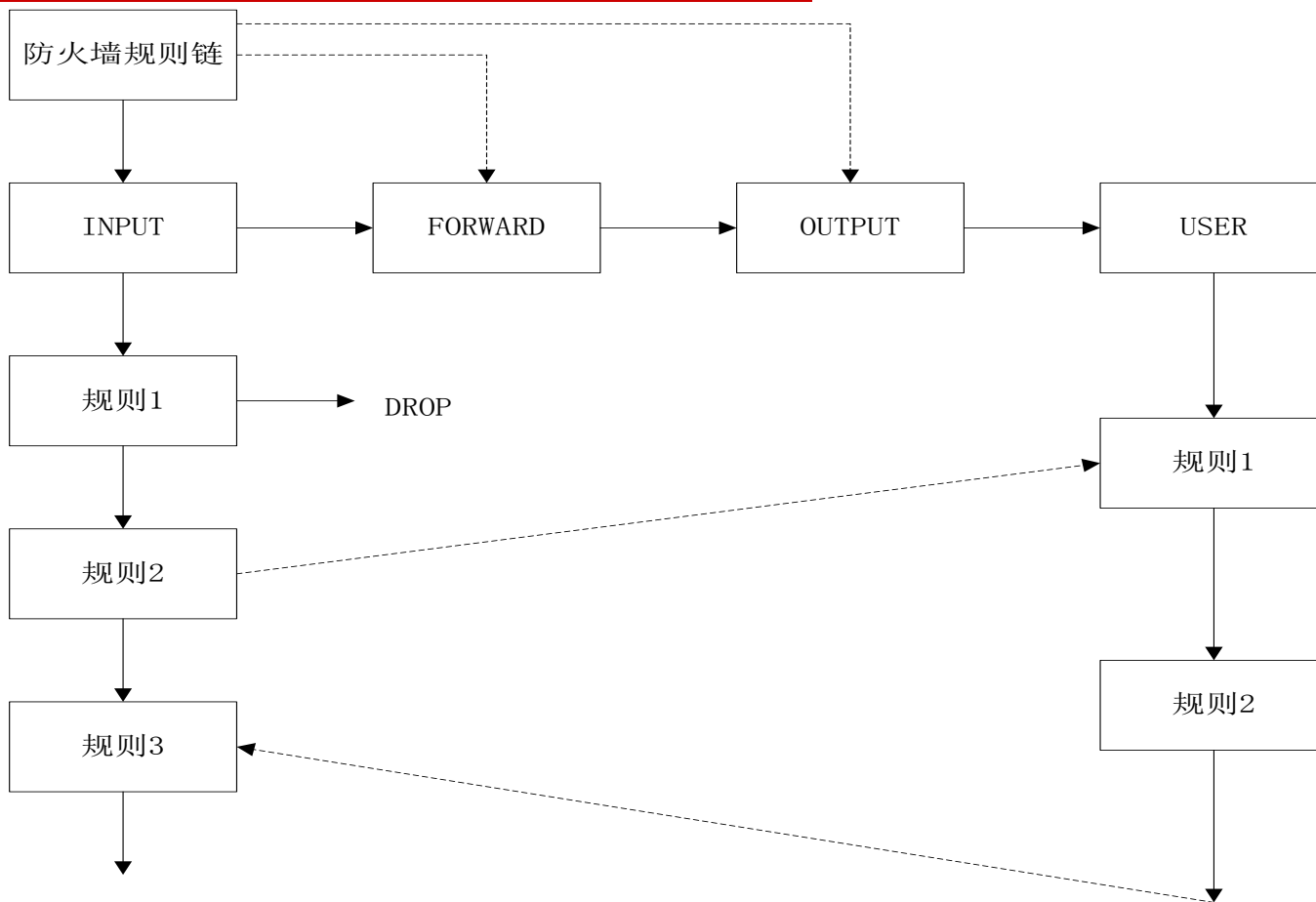
□ 支持的网络连接状态

- **NEW:** 新建连接, 连接初始报文或只看到一个方向的数据包
- **ESTABLISHED:** 已建连接, 双向通讯
- **RELATED:** 相关连接, 用于处理**FTP**等协商端口的网络协议
- **INVALID:** 非法状态

□ 实现机制

- 使用**hash**表等实现**CT**表, 支持对已记录连接的快速查找
- 在**PREROUTING, INPUT, POSTROUTING**等**hook**点上注册**callback**函数, 用于跟踪网络连接状态
- 支持用户态程序(**IPTables**)灵活配置各个**hook**点上的防火墙规则, 对网络连接进行访问控制

Netfilter防火墙规则过滤





Iptables

- 为用户提供了配置**netfilter**规则的命令行接口
 - **\$ iptables [-t table] command [match] [target]**
- **command**-规则配置动作
 - **-A**(添加)、**-D**(删除)、**-P**(缺省策略)、**-N**(新建链)、**-F**(清空)、**-L**(列举)
- **match**-规则匹配条件
 - 通用匹配和特定协议匹配
 - 支持“与”关系
- **Target**-目标操作
 - **ACCEPT、DROP、REJECT、RETURN**



Netfilter/iptables的过滤与报文状态检查机制

□ 两种策略

- (1) 设置缺省的通行策略为允许(**ACCEPT**)，然后定义禁止的网络流量和行为； - **Bad**
- (2) 设置缺省的通行策略为禁止(**DROP**)，然后定义允许的网络流量和行为； - **Good**

□ 静态包过滤

- **# iptables -t filter -A FORWARD -s 192.168.0.0/24 -d 192.168.1.0/24 -j ACCEPT**

□ 报文状态检查

- 状态： **NEW**、 **ESTABLISHED**、 **RELATED**、 **INVALID**
- **# iptables -t filter -A FORWARD -d [WEB_SERVER] -m state --state NEW -j ACCEPT**
- **# iptables -t filter -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT**



Netfilter/iptables的NAT机制

□ NAT机制类型

- **SNAT**: 源地址/端口NAT
- **DNAT**: 目的地址/端口NAT

□ NAT功能类型

- **IP伪装 (masquerading)**: 属于SNAT
 - **#iptables -t nat -A POSTROUTING -i eth1 -o eth0 -j MASQUERADE**
- **透明代理 (transparent proxying)**: 属于DNAT
 - **# iptables -t nat -A PREROUTING -i eth1 -j DNAT --to 5.6.7.8**
- **端口转发 (port forwarding)**
 - **#iptables -A PREROUTING -t nat -p tcp -d 1.2.3.4 --dport 8080 -j DNAT --to 192.168.1.1:80**

VPN

□ VPN—虚拟私有网(Virtual Private Network)

- 利用大规模网络(如**Internet**)上公用链路代替物理链路构建的安全专有网络。
- 大型跨地域企业构建企业网的常用方案。

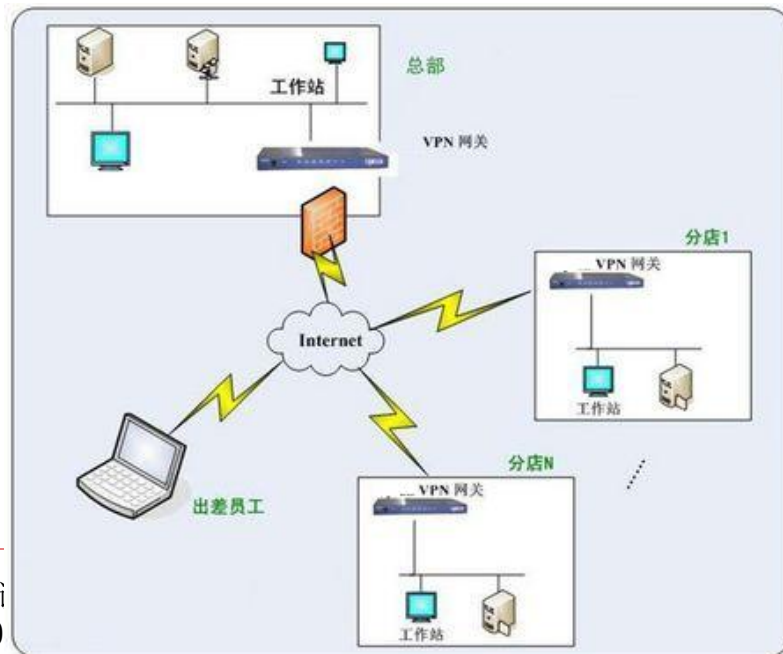
□ VPN类型

- **IPSEC VPN(网络层)**
- **SSL VPN(传输层)**

□ VPN产品

- 国内通常在防火墙上集成
- 专用**VPN**设备

□ 开源VPN—OpenVPN





北大VPN使用

FG50022405500006 SSL VPN Remote Access Web Portal - Microsoft Internet Explorer

文件(F) 编辑(E) 查看(V) 收藏(A) 工具(T) 帮助(H)

后退 前进 刷新 地址 搜索 收藏夹 转送到 链接 转到 SnagIt

地址: https://124.205.79.5/remote/index

Peking University VPN System

Fortinet SSL VPN Client 1.0

Link Status	Bytes Sent	Bytes Received
Up	4634	236

Install Uninstall Connect Disconnect Refresh now

Fortinet SSL VPN client connected to server

Connect to Web Server Go

Test for Reachability(Ping) Go

Telnet to Host Go

Vnc to Host Go

Rdp to Host Go

fortissl
速度: 100.0 Mbps
发送: 4,634 字节
收到: 236 字节

2011年 15:21



内网安全管理

- **70%以上安全事故是由网络内部原因造成的**
 - 防火墙等边界安全防护不能应对
 - 内网安全管理的必要性
- **内网安全管理**
 - 有效地对内网终端进行安全管理和健康状态监控，从而增强内部网络的安全性
 - 终端安全管理
 - 终端运维管理
 - 终端补丁分发管理
 - 系统日志管理

内容安全管理(SCM)

- 内容安全管理**SCM**
 - 关注对网络传输内容的安全性检查
- 网络行为监控
 - 网络行为监控审计
 - 绿色上网
- 防虫墙/防病毒网关
 - 防病毒、蠕虫网关
- 垃圾邮件过滤网关
- ...



北大计算机研究所自主研发的防虫墙产品



边界安全防御发展趋势-UTM&高性能

□ “胖” 防火墙→UTM

- **UTM**—统一威胁管理

- **Many features in one box**

- 网络访问与控制: 防火墙, ...

- 加密与身份认证: **VPN**, ...

- **SCM**: 垃圾邮件过滤, 防病毒, **IDS/IPS**, 上网监管, ...

□ 高性能问题

- 目前**x86**架构

- **RISC**架构

- **NP**架构+**x86**并行架构



课堂实践：防火墙配置

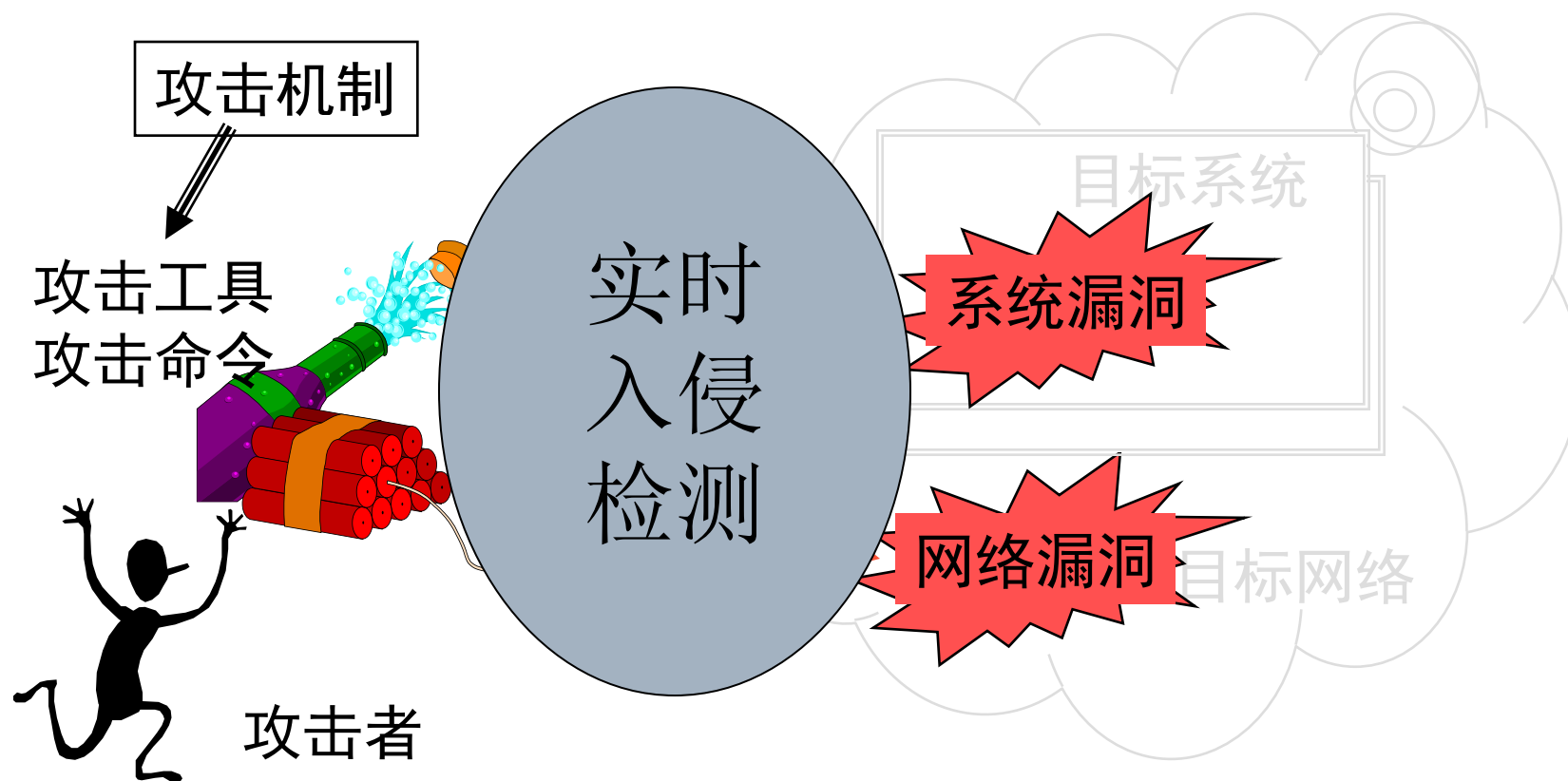
- 实践任务：配置**Linux**操作系统平台上(**Linux Metasploitable**)的**iptables**，或者**Windows**操作系统平台上(**Windows Metasploitable**)的个人防火墙，完成如下功能，并进行测试：
 - 过滤**ICMP**数据包，使得主机不接受**Ping**包；
 - 只允许特定**IP**地址(如局域网中的**Linux**攻击机**192.168.200.3**)，访问主机的某一网络服务(如**FTP**、**HTTP**、**SMB**)，而其他的**IP**地址(如**Windows**攻击机**192.168.200.4**)无法访问。



内容

- 1. 安全模型-P2DR模型**
- 2. P: 防御技术**
- 3. D: 检测技术**
- 4. R: 响应技术**
- 5. 作业6: 分析蜜网网关中防火墙与入侵检测系统配置规则**

入侵检测系统的用途





入侵检测系统**IDS**: **Intrusion Detection System**

- ❑ 入侵检测技术基本概念与发展过程
- ❑ 入侵检测技术
- ❑ 入侵检测系统的分类与部署
- ❑ 入侵防御系统**IPS**
- ❑ 著名的开源入侵检测系统-**Snort**



入侵检测系统基本概念

□ 入侵检测(**Intrusion Detection**)

- 入侵检测，顾名思义，就是对入侵行为的检测与发现。
- 入侵检测即为通过对计算机网络或计算机系统中若干关键点信息的收集和分析，从中发现入侵行为的一种安全技术。

□ 入侵(**Intrusion**)

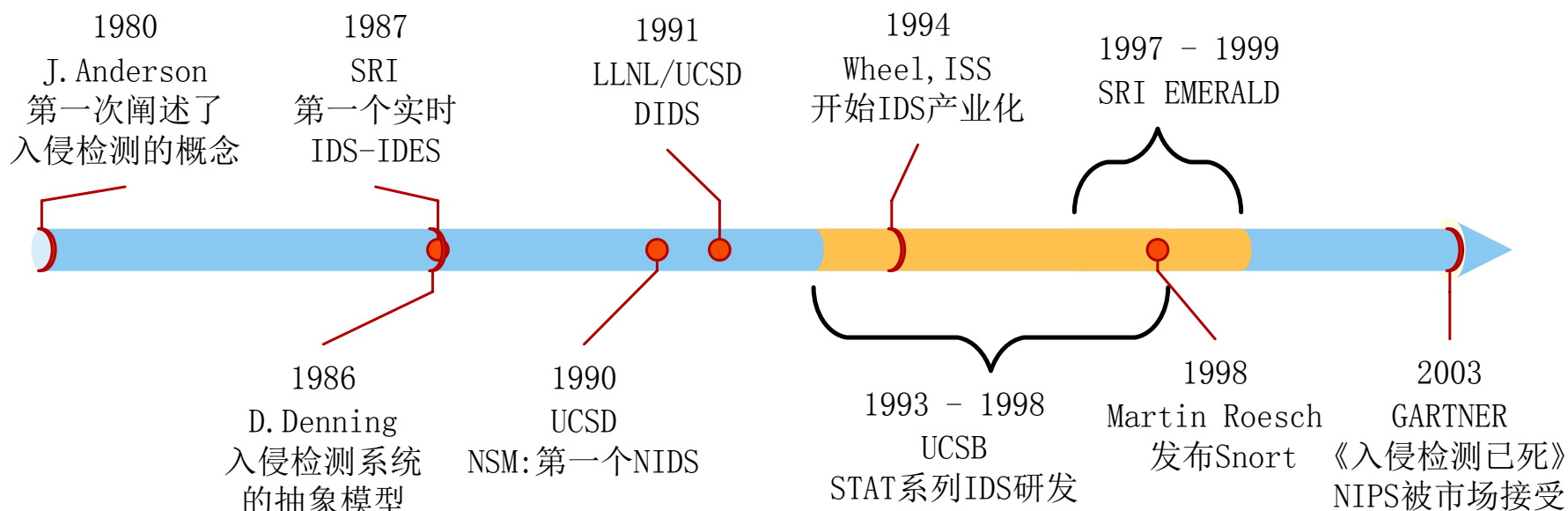
- 一次入侵可被定义为任何尝试破坏信息资源的保密性、完整性或可用性的行为。

□ 入侵检测系统(**Intrusion Detection System**)

- 实现入侵检测技术，专门用于入侵行为发现和处理的软件系统或硬件设备。

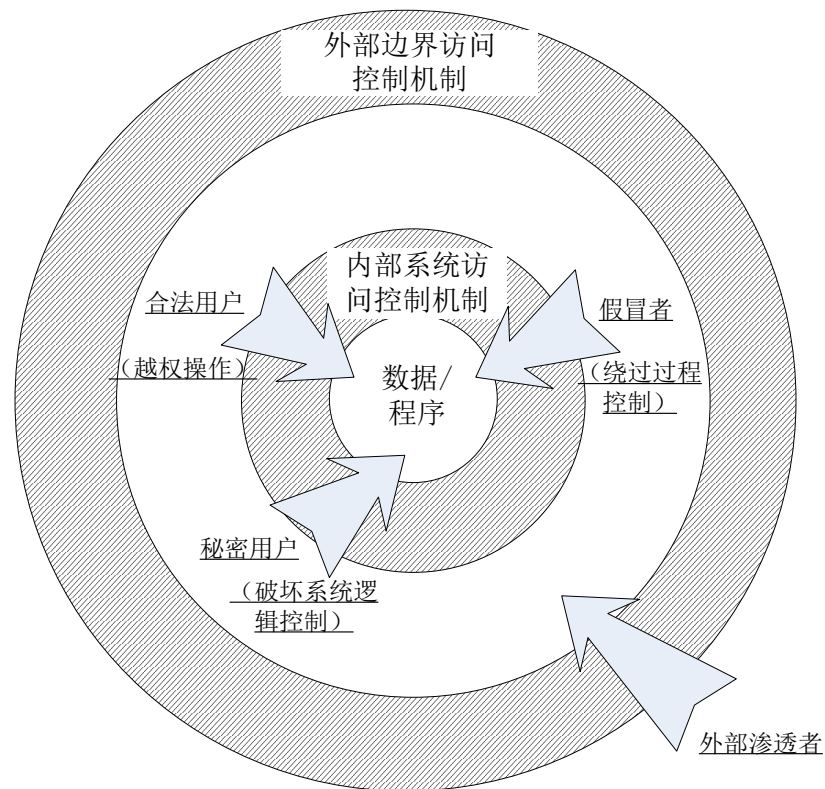
□ 防火墙 **VS IDS**: 门卫 **VS** 巡逻队

入侵检测技术发展历程

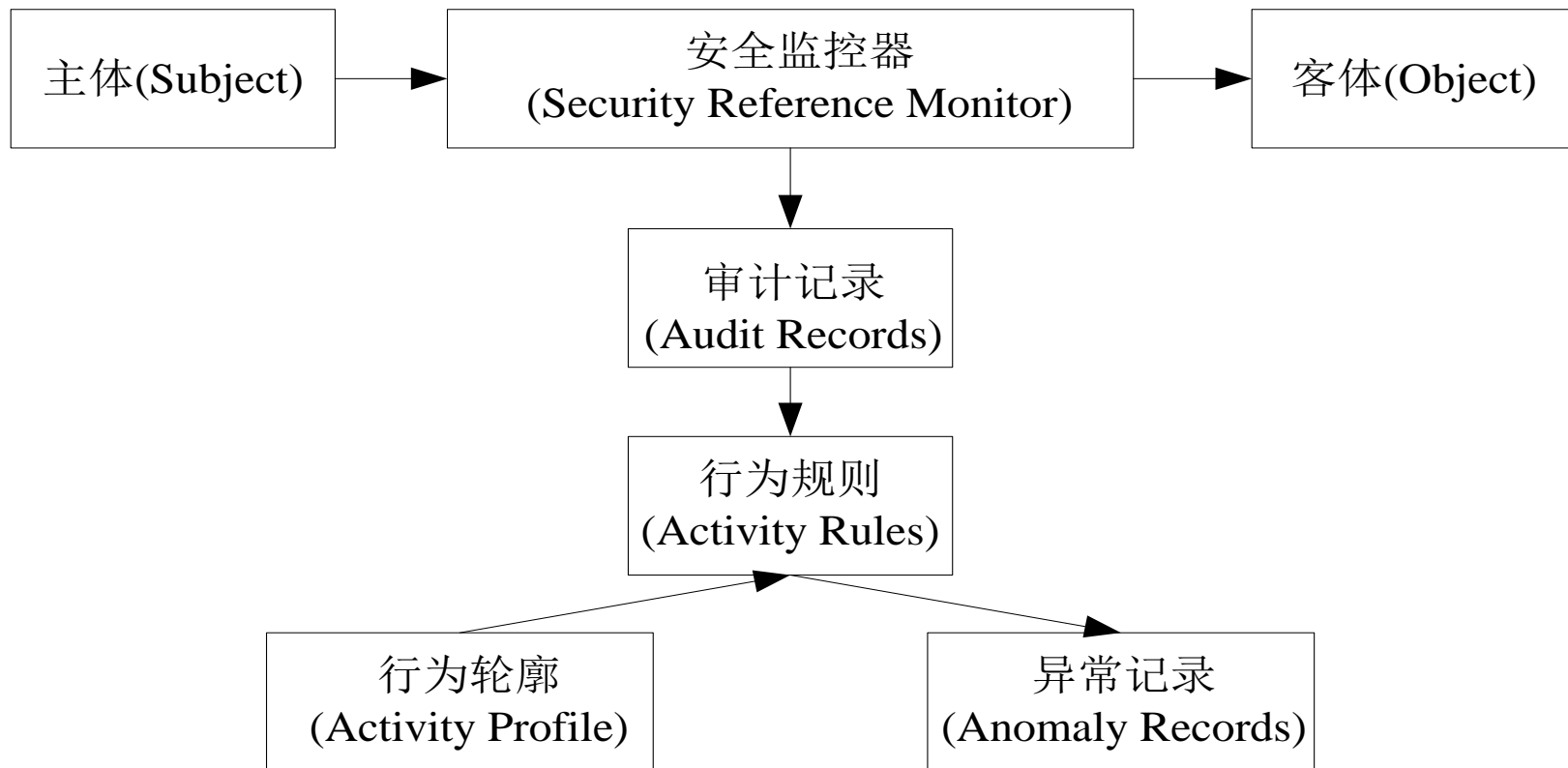


入侵威胁分类图

- **1980 J. Anderson**入侵检测开山之作
 - **“Computer Security Threats Monitoring and Surveillance”**
- 入侵者分类
 - 外部渗透者：攻破外部访问控制
 - 内部渗透者：假冒者(攻破过程控制), 违法者(误用访问), 秘密用户(攻破逻辑控制)
- 检测分析模型
 - 误用检测：监视违反特定规则的权限误用行为，违法者检测
 - 异常检测：基于统计方法建立用户正常行为轮廓，假冒者检测



入侵检测抽象理论模型





入侵检测的数据源

□ 初期：基于主机

- 审计记录、系统日志、系统调用、应用程序日志...
- **HIDS**: 基于主机的入侵检测系统

□ 90年代：基于网络

- 网络监听数据
- **1990年**，加州大学戴维斯分校的**NSM**
- **NIDS**: 基于网络的入侵检测系统

□ 融合：分布式入侵检测系统

- 劳伦斯伯克利国家实验室/加州大学戴维斯分校一起开发分布式入侵检测系统(**DIDS**)



入侵检测技术与产业化

- **20世纪90年代**
- 研究领域的百花齐放
 - 误用检测：**UCSB: STAT**系列；**SRI: P-BEST**和**EMERALD**分布式入侵检测系统
 - 异常检测：**IDES**统计异常检测器；**NIDES**、**Haystack**等、基于神经网络方法的**NNID**、基于免疫学方法、数据挖掘方法
- 产业化
 - **1994**年，**Wheel Group**开始推出商业化的入侵检测产品，后被**Cisco**所收购
 - **1997**年**ISS**公司推出著名的**RealSecure**入侵检测系统产品
 - 开源：**1998**年**Snort**，**1999**年**Bro**
- 入侵检测技术的转型
 - **2003**年商业咨询公司**GARTNER**《入侵检测已死》
 - 入侵防御系统(**IPS**)被市场接受

入侵检测技术评估指标

□ 检测率(True Positive)

- 攻击事件的检出效果：检测出攻击事件数和全部攻击数之比
- 漏报率 (**false negative**)：攻击事件没有被检测到

□ 误报率(False Positive)

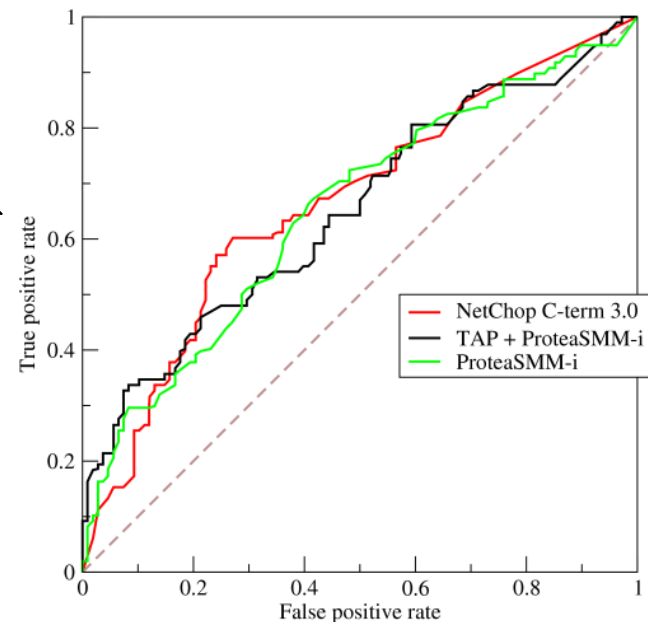
- 把正常事件识别为攻击并报警

□ 检测率和误报率往往不能同时很好

- “基调悖论(**base-rate fallacy**)”

□ IDS准确率评判标准：ROC曲线

- 通过检测率和误报率构建**ROC**曲线
- 对比**ROC**曲线所围成的面积





入侵检测技术

❑ 误用检测(**misuse detection**)

- 也称为基于特征的检测 (**signature-based detection**)
- 建立起已知攻击的特征库
- 判别当前行为活动是否符合已知的攻击特征

❑ 异常检测(**anomaly detection**)

- 也称为基于行为的检测 (**behavior-based detection**)
- 首先建立起系统的正常模式轮廓
- 若实时获得的系统或用户的轮廓值与正常值的差异超出指定的阈值，就进行入侵报警

误用检测

□ 目前研究工作比较多，并且已经进入实用

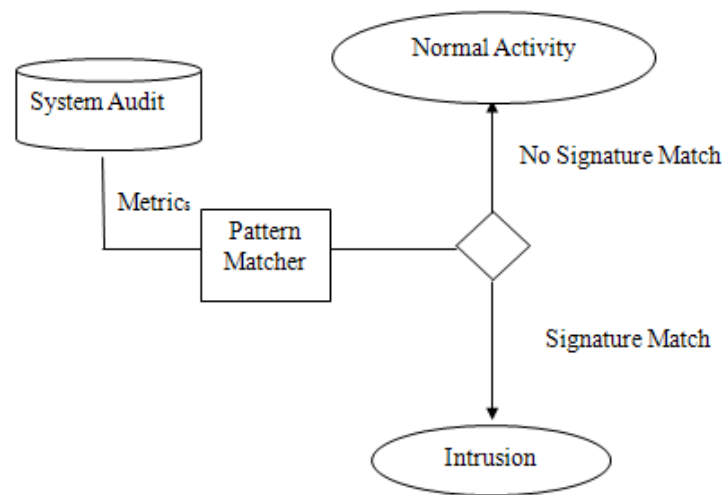
- 建立起已有攻击的模式特征库
- 难点在于：如何做到动态更新，自适应

□ 常用技术

- 基于简单规则的模式匹配技术
- 基于专家系统的检测技术
- 基于状态转换分析的检测技术

□ 攻击特征提取

- 专家提取
- 自动提取方法(研究热点)

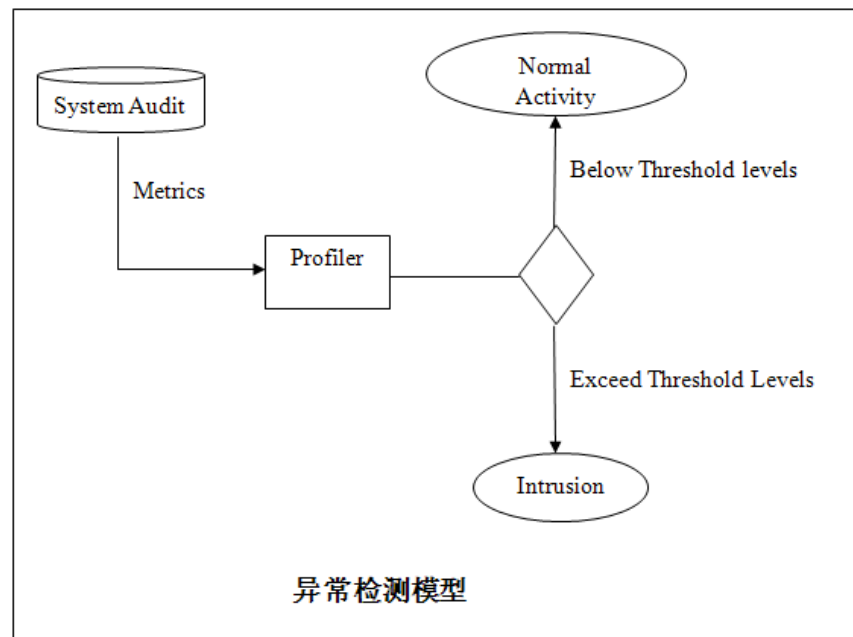


误用检测模型

异常检测

- 比较符合安全的概念，但是实现难度较大
 - 正常模式的知识库难以建立
 - 难以明确划分正常模式和异常模式

- 常用技术
 - 统计方法
 - 预测模式
 - 神经网络





入侵检测系统的种类

□ 基于网络的入侵检测系统(**NIDS**)

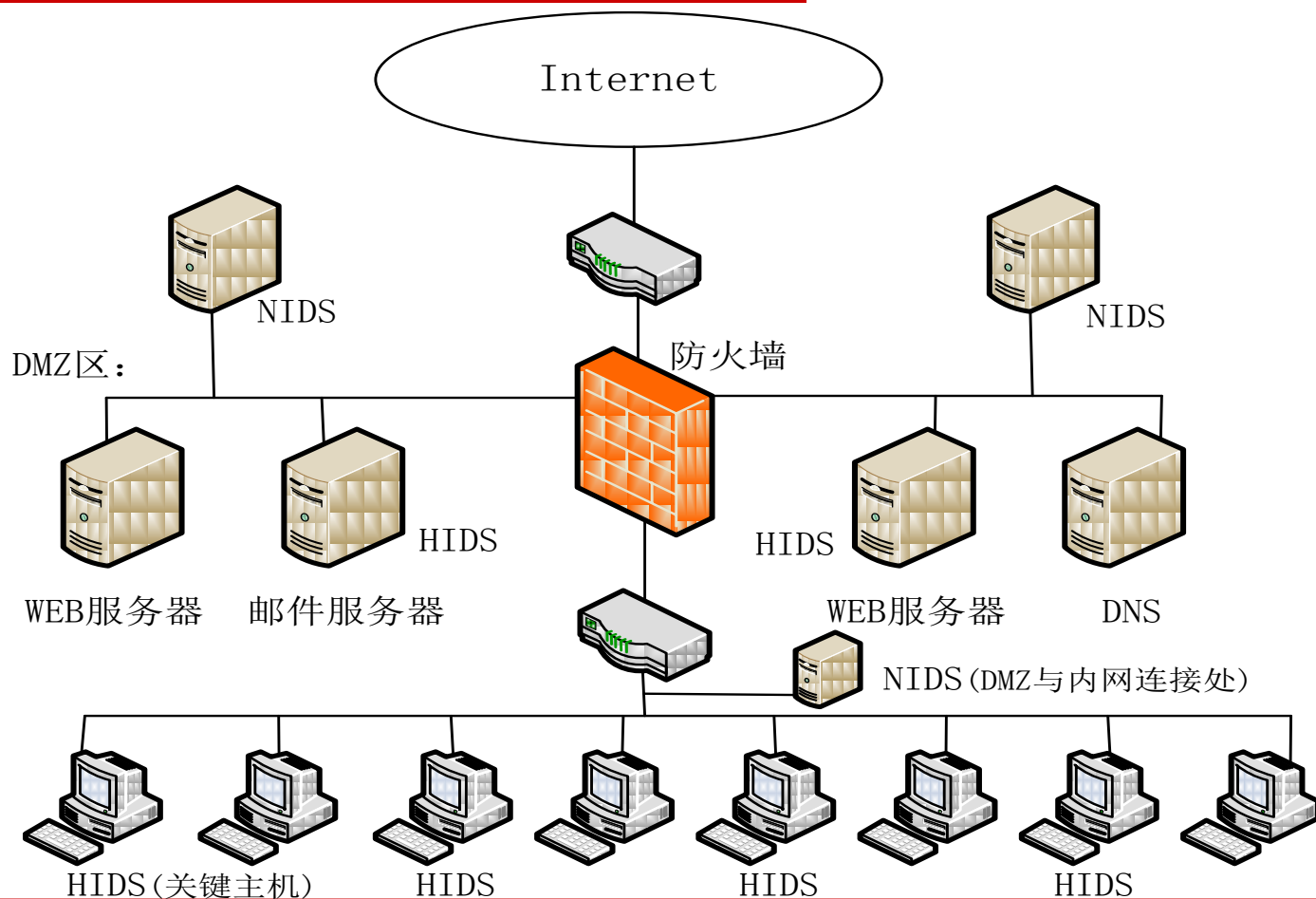
- **IDS**可以放在防火墙或者网关的后面，以网络嗅探器的形式捕获所有的对内对外的数据包

□ 基于主机的入侵检测系统(**HIDS**)

- 安全操作系统必须具备一定的审计功能，并记录相应的安全性日志
- 基于内核
 - 从操作系统的内核接收数据
- 基于应用
 - 从正在运行的应用程序中收集数据

□ 分布式入侵检测系统(**DIDS**)

入侵检测系统的部署位置



网络入侵防御系统IPS

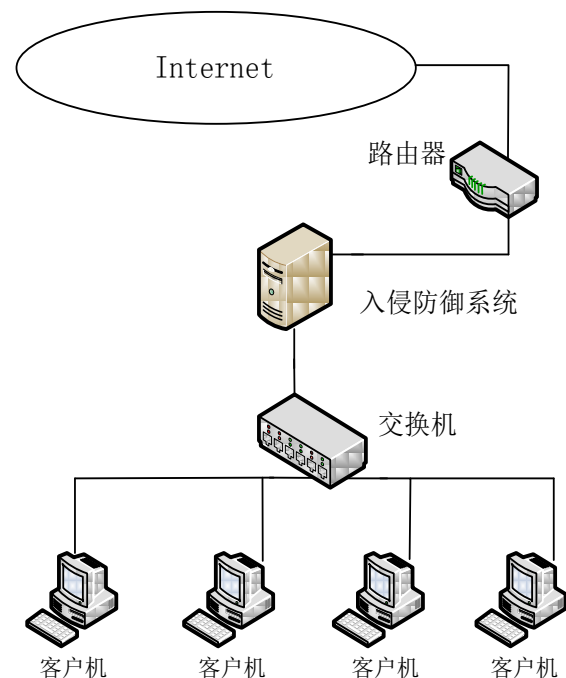
□ IDS与IPS

■ IDS: 旁路监听, 只起到 Detection机制

- 侧重低漏报率, 造成误报率较高
- 对使用者技术水平要求较高, 应急响应及时

■ IPS: 内联模式, 实时处置数据包

- 侧重低误报率 (对正常业务不造成影响)
- 高效的处理性能
- 即插即用, 无需使用者参与





Snort网络入侵检测系统

□ Snort开源网络入侵检测系统

- 最知名的开源入侵检测系统
- 网络入侵检测/入侵防御系统事实标准
- 三个主要功能：数据包嗅探、数据包分析与记录、网络入侵检测

□ Snort发展史

- **Martin Roesch, 1998**年底开始发布**Snort**, 最早仅仅是个**sniffer, v1.5**轻量级**NIDS**
- **2001**年成立**Sourcefire**公司
- **2004**年**Snort v2.0**: 应用新的检测引擎和多模匹配算法
- **2008**年**7月 SnortSP v3.0 Beta**: 全新的框架
 - **Snort Security Platform: traffic analysis framework**
 - **Snort 2.8.x detect engine: traffic analysis module**
- 目前最新版本: **Snort v2.9.0**



Snort v2.x的特性

□ Snort v2.x: 目前稳定可用的开源NIDS

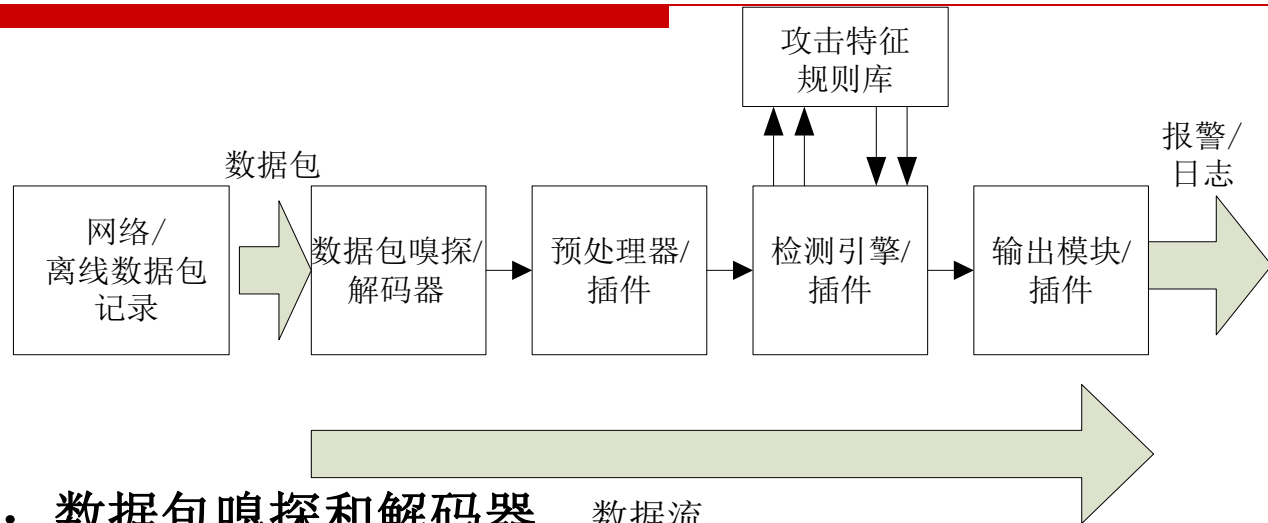
- 源码开放，跨平台(C语言编写，可移植性好)
- 利用libpcap作为捕获数据包的工具

□ 框架特点

- 设计原则：性能、简单、灵活
- 包含四个子系统：**Sniffer**、预处理器、检测引擎、输出模块
- 采用一套源码级插件机制，作为系统扩展的手段(预处理、日志)
- 检测方法：以误用检测为主，结合异常检测方法
- 命令行方式运行，也可以用作一个**sniffer**工具

□ Know More: 源码阅读+<<Snort 2.0入侵检测>>

Snort v2.x基本框架结构



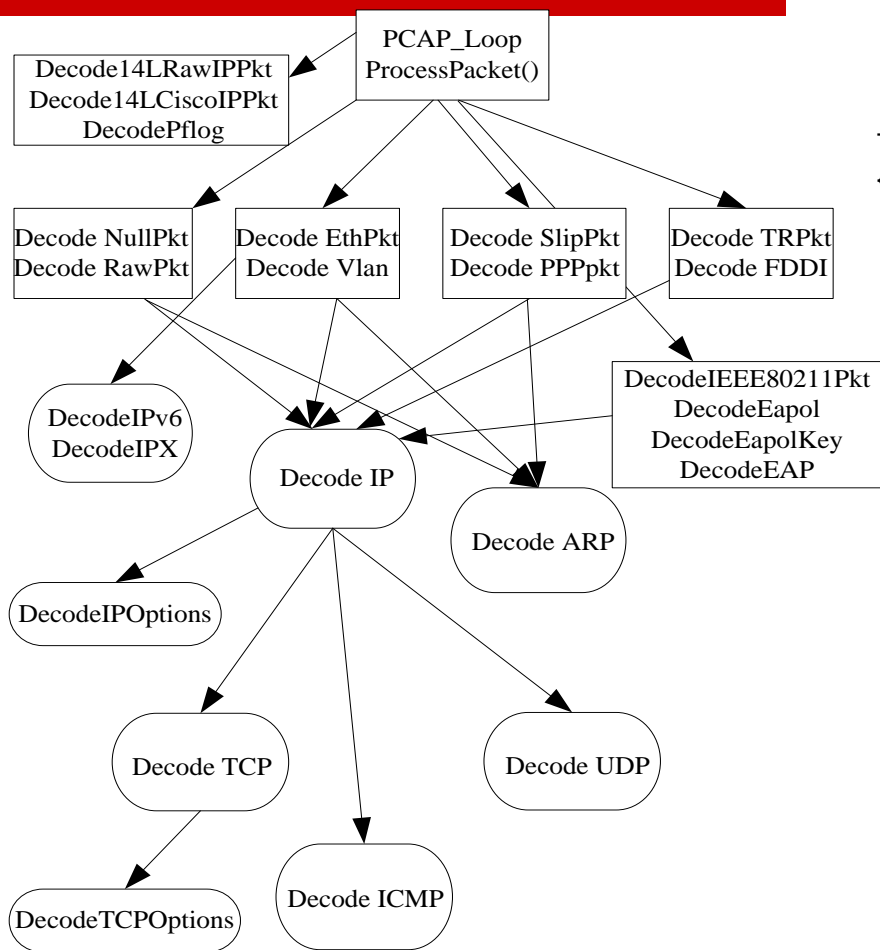
- ❑ **Sniffer:** 数据包嗅探和解码器 数据流
- ❑ 预处理器/插件(**preprocessors**)
- ❑ 检测引擎/插件: 多模匹配算法(内容检测) + 规则检查插件(包头字段内容)
- ❑ 输出模块/插件: 通过插件机制支持文本文件、二进制、**syslog**、数据库多种报警日志机制
- ❑ 攻击特征规则库: 包含已知攻击的检测规则



Snort v2.x – 数据包嗅探

- 网卡设置混杂模式
- 基于**libpcap**库
 - **libpcap**在链路层从网卡取得数据包
 - 提供接口允许开发人员解析和分析数据包
- **Snort和libpcap**的连接
 - **Snort.c**: 进入**pcap**主执行循环**pcap_loop**
 - 回调函数**ProcessPacket()**
 - 当**pcap**接收到数据包，即调用回调函数
 - **Snort**通过**ProcessPacket()**函数对数据包进行处理
→**decode.c**解析->**detect.c/fpdetect.c**检测

Snort v2.x – 包解析



```

typedef struct _Packet
{
    struct pcap_pkthdr *pkth;
    u_int8_t *pkt;
    EtherHdr *eh;
    //标准TCP/IP/Ethernet/ARP包头
    VlanTagHdr *vh;
    WifiHdr *wifih;
    //无线LAN包头
    EtherARP *ah;

    IPHdr *iph,*orig_iph;
    u_int32_t ip_options_len;
    u_int8_t *ip_options_data;

    TCPhdr *tcph,*orig_tcph;
    u_int32_t tcp_options_len;
    u_int8_t *tcp_options_data;

    UDPhdr *udph,*orig_udph;...
    
```




Snort v2.x – 预处理器与插件

- **IP分片/TCP流处理/流跟踪**
 - **frag#/stream#/flow**
- **应用层协议分析**
 - **telnet, rpc, http**
 - **工作: 应用层协议解码, 应用层协议规范异常检查, Unicode解码, ...**
- **异常检测**
 - **Portscan类: portscan#/sfportscan**
 - **Arpspoof检测, BO探测**



Snort v2.x – 核心检测引擎

- **核心检测引擎: Signature-based (Rule-based)**
 - **Snort规则库:** 描述已知攻击的数据包/流特征
 - **检测引擎:** 基于模式匹配算法查看当前数据包/流是否满足已知攻击规则
- **Snort规则库**
 - **Sourcefire VRT Certified Rules:**
<http://www.snort.org/vrt/>
 - ***snortrules-snapshot-CURRENT_s.tar.gz* :** 当前公开发布最新规则库
 - **VRT Certified Rules for Snort: *snortrules-snapshot-CURRENT.tar.gz*:** Sourcefire 付费服务
 - **Snort规则知识库:** www.snort.org/snort-db/



Snort v2.x的攻击规则(1)

- ❑ 示例规则-**wuftpd**格式化字符串攻击检测规则
- ❑ **alert tcp \$EXTERNAL_NET any -> \$HOME_NET 21
(msg:"FTP EXPLOIT wu-ftpd 2.6.0 site exec format
string overflow Linux";
flow:to_server, established;
content:"|31c031db31c9b046cd8031c031db|";
reference:bugtraq,1387;reference:cve, CAN-2000-
0573;
classtype:attempted-admin;
sid:344;rev:4;)**



Snort v2.x的攻击规则(2)

□ 规则头

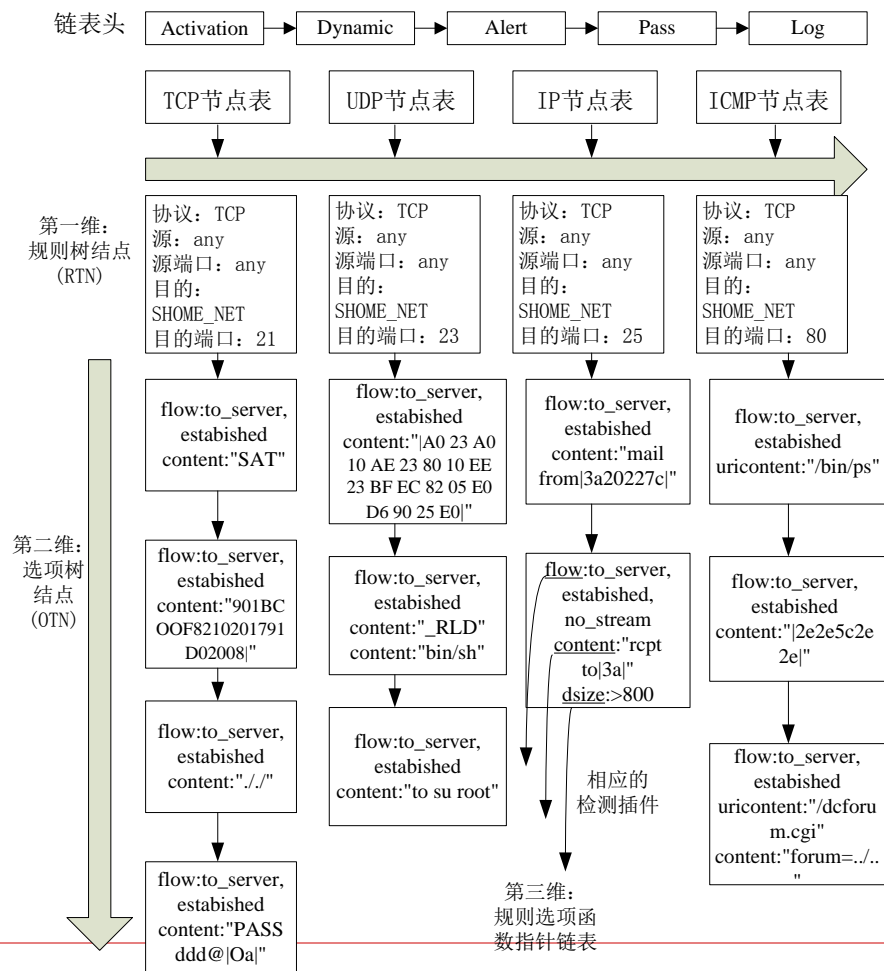
- **alert:** 报警/日志类型 (**alert, log, pass, dynamic, activate**)
- **tcp:** 协议类型 (**tcp, udp, icmp, ip**)
- **\$EXTERNAL_NET:** 源IP地址匹配范围; **any:** 源端口
- **->:** 流方向
- **\$HOME_NET:** 目标IP地址匹配范围; **21:** 目标端口

□ 规则体

- 检测规则选项
 - **flow:to_server, established;** 匹配数据包为从客户端至服务器端, 已建连接
 - 流状态检测规则选项 → **spp_clientserver.c**检测插件
 - **content:"|31c031db31c9b046cd8031c031db|";**
 - 匹配内容模式→多模式匹配算法检测当前数据包中是否含有该模式
- 报警信息选项: **msg:"FTP EXPLOIT wu-ftpd ... overflow Linux"**
- 索引和分类选项: **reference, classtype**
- 规则ID和版本: **sid, rev**

Snort v2.x的规则树解析建立

- 三维链表结构
- 第一维：规则树节点
RTN
 - 同一应用协议规则集
- 第二维：选项树节点
OTN
 - 一条特定检测规则
- 第三维：规则选项检查函数指针链表
 - 特定规则上的所有检测选项函数





Snort v2.x的性能优化

-多规则检测引擎

- 单模式匹配算法
 - **Knuth-Morris-Pratt(KMP)**: “快速”模式匹配
 - **Boyer-Moore (BM)**算法
- 多模式匹配算法
 - 问题描述: 一组**pattern**, 一个**string**, 判断**string**中所有**pattern**的匹配
 - **Aho-Corasick (AC)**算法
 - **Wu-Manber (WM)**算法
- **Snort v2.x**中的多规则检测引擎
 - 将目标端口的一组规则(同一**RTN**下的**OTN**节点链表)进行组织, 并构成一个多模式库
 - 数据包进入某一**RTN**后, 首先进行多模匹配, 回调函数中判断其他规则项是否匹配



Snort v2.x的规则项检测插件

□ detection-plugins

- **IP协议: id, fragbit, proto, same_check, tos, ipoption, ttl, ...**
- **ICMP协议: id, type, code, seq, ...**
- **TCP协议: win, seq, flag, ack, client_server, dsize ...**
- **应用层协议: urilen, rpc_check, ftp_bounce**
- **数据检测插件: isdataat, byte_jump, byte_check, session, ...**



Snort v2.x: 输出模块

□ 报警/日志输出源

- 数据包解码模块：不符合协议规范的数据包
- 预处理器：应用协议规范异常检测报警、异常检测报警...
- 检测引擎：触发已知攻击特征规则的报警

□ 报警/日志模块机制

- 报警方式(**alert**)/日志方式(**log**)
- **event*.c/log.c**
- 基于插件机制可扩展的输出模块：**output-plugins**

□ 输出模块列表

- 文件：**alert_fast, alert_full, csv, unified, tcpdump, ...**
- 网络：**unixsock, syslog, sf_socket, prelude, ...**
- 数据库：**MYSQL, POSTGRESQL, ODBC, ORACLE, MSSQL**



Snort的安装

☐ Linux平台

- 源码包安装: 标准的./configure;make; make install
- 依赖: libpcap, libpcre, ...
- RPM包安装: rpm -Uvh snort-2.x..x.rpm
- Apt-get, yum在线安装与升级
- Snort规则文件: snort.org注册用户获取
 - ☐ 在线升级机制? -Roo蜜网网关(onikmaster)

☐ Windows平台

- 依赖winpcap, 获取snort for win32安装文件并安装



Snort的使用

□ Snort的三种使用模式

- 嗅探器模式: **./snort -vde**
- 包解码器模式: **./snort -vde -l ./log**
- 网络入侵检测系统: **./snort -c snort.conf**

□ Snort配置

- **Snort.conf**配置文件 – **Snort**的参数配置
- 定制变量参数: 本地网段、**Web**服务器列表等
- 预处理器配置: 打开哪些预处理器, 参数配置
- 配置规则文件列表
- 输出插件定义: 输出报警到何处, 缺省**log**位置
/var/log/snort

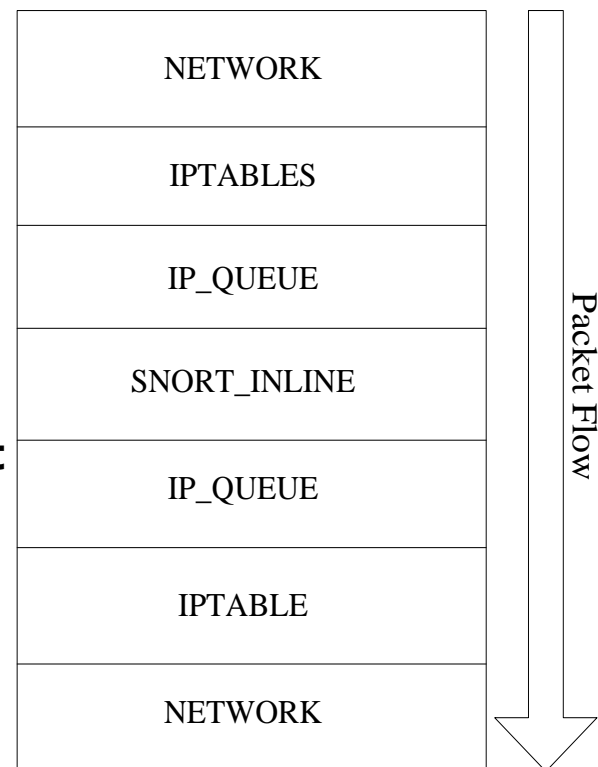
Snort-inline

□ 基于Snort的NIPS

- 透明网桥方式
- 对通过包做入侵检测
- 直接丢弃攻击包/流
- 实时阻断
- 提供安全防御/保护

□ ./configure --enable-inline

- use the **libipq** interface for inline snort
- libipq is a development library for [iptables](#) userspace [packet queuing](#)
- Libipq provides an [API](#) for communicating with **ip_queue**.
- Libipq has been deprecated in favour of the newer [libnetfilter_queue](#) in Linux kernel-2.6.14 onwards.



开源的HIDS - OSSEC

- 日志文件分析与关联
 - 灵活的基于**XML**的检测规则
 - 基于时间的报警
 - 大量现有已知攻击规则库: **SSH, Apache**等**32**类
- 完整性检查
 - 文件/目录: 访问权限、大小、拥有者、**md5sum**
 - **Windows**注册表
- **Rootkit**检测
 - **Files/Trojans database**; 隐藏进程/端口/混杂模式
- 支持**Windows: Win32 Agent/OSSEC server**
- 集成**Nmap**



课堂实践：Snort使用

- 实践任务：在**BT4 Linux**攻击机或**Windows Attacker**攻击机上使用**snort**，对给定的**pcap**文件进行入侵检测，获得报警日志：
 - 检测**Pcap**文件：作业**4.1**中的任意一个**pcap**
 - **Snort**运行命令提示
 - 从离线的**pcap**文件读取网络日志数据源
 - 在**snort.conf**中配置明文输出报警日志文件
 - 指定报警日志**log**目录(或缺省**log**目录
=**/var/log/snort**)



内容

- 1. 安全模型-P2DR模型**
- 2. P: 防御技术**
- 3. D: 检测技术**
- 4. R: 响应技术**
- 5. 作业6: 分析蜜网网关中防火墙与入侵检测系统配置规则**



安全响应组织与机构

- 安全应急响应组织 — **CSIRT (Computer Security Incident Response Team)**
 - **CERT/CC**: 美国, 1989年在Morris蠕虫爆发后成立
 - 中国: **CCERT(1999), CNCERT/CC(2000)**
 - **FRIST** (国际性事件响应与安全组织论坛), **APCERT**(亚太地区**CERT**组织联盟)
- 法律机构
 - 公安部网络安全保卫局**&**网警
- 研究领域
 - 安全研究机构、团队
- 产业界
 - 安全公司
 - 反病毒公司



响应技术

- 计算机及网络取证技术
 - 分析攻击并寻找追溯线索
 - 保全并提取现场证据：法律执行部门
- 攻击追溯和响应
 - **Attacker Trace: very difficult, Step-stone attack**
 - 以牙还牙, 以暴制暴? – 并不可取
- 备份恢复
 - 建立良好的关键数据备份习惯
 - **RAID**冗余磁盘阵列->冷备份->双机热备（保持业务连续性）
- 灾难恢复
 - 重要性数据的异地容灾备份：**2/5**的公司经历大灾难后再也不能恢复运作
- 你做好备份了吗？
 - 寝室/实验室笔记本被窃？火灾？...？未雨绸缪！



被黑网站搜索

Google

allintitle: hacle

Google

d3triment

Google 搜索

高级

网页

打开百宝箱...

『 <h1>Hacked By d3triment@l</h1>』

产品『. Hacked By d3triment@l.』
粮食. 产品简介, 精选本地山区无污染
www.jsnw.gov.cn/nbh/cpbl/cpdetail.

Hacked by ring04h, just for fun

今天, 很多用DISCUZ论坛的主页都
知名黑客团队“邪恶八进制”核心成
www.sixseo.cn/.../Hacked-by-ring0.

<h1>mc_intikam sanalordu.or

首页 ▪ Hacked by Vezir.04 ▪ Hac
划统计 ▪ 计生双评 ▪ 政策法规. 兰
www.lhjd.gov.cn/jisheng/www/list.as

网页

打开百宝箱...

搜索 d3triment

『 <h1>Hacked By d3triment@l</h1>』农产品详细资料

Hacked By d3triment@l. 所属分类, 粮食. 产品简介, 精选本地山区无污染的苦荞为原料, 所有
植物中只有荞麦含丰富的生物类黄酮, 而苦荞比荞麦高十几倍。 ...
www.jsnw.gov.cn/nbh/cpbl/cpdetail.asp?proid=38 - 网页快照

『 <h1>Hacked By d3triment@l</h1>』农产品详细资料

Hacked By d3triment@l. 所属分类, 苗木. 产品简介, 我園位于山西省晋中腹地距108国道731公
里段往北大运高速张兰出口五公里处。 现我園基础设施齐全, 技术力量雄厚, ...
www.jsnw.gov.cn/nbh/cpbl/cpdetail.asp?proid=20 - 网页快照

显示来自 www.jsnw.gov.cn 的更多搜索结果

上海碧泉化工实业有限公司 产品列表

Hacked By d3triment@l. Hacked By d3triment@l. Hacked By d3triment@l ... Hacked By
d3triment@l. 高级搜索. 推荐产品. 分类列表. Hacked By d3triment@l ...
www.bqchem.com/eBusiness/GB/product_list.asp?catalogid=15 - 网页快照

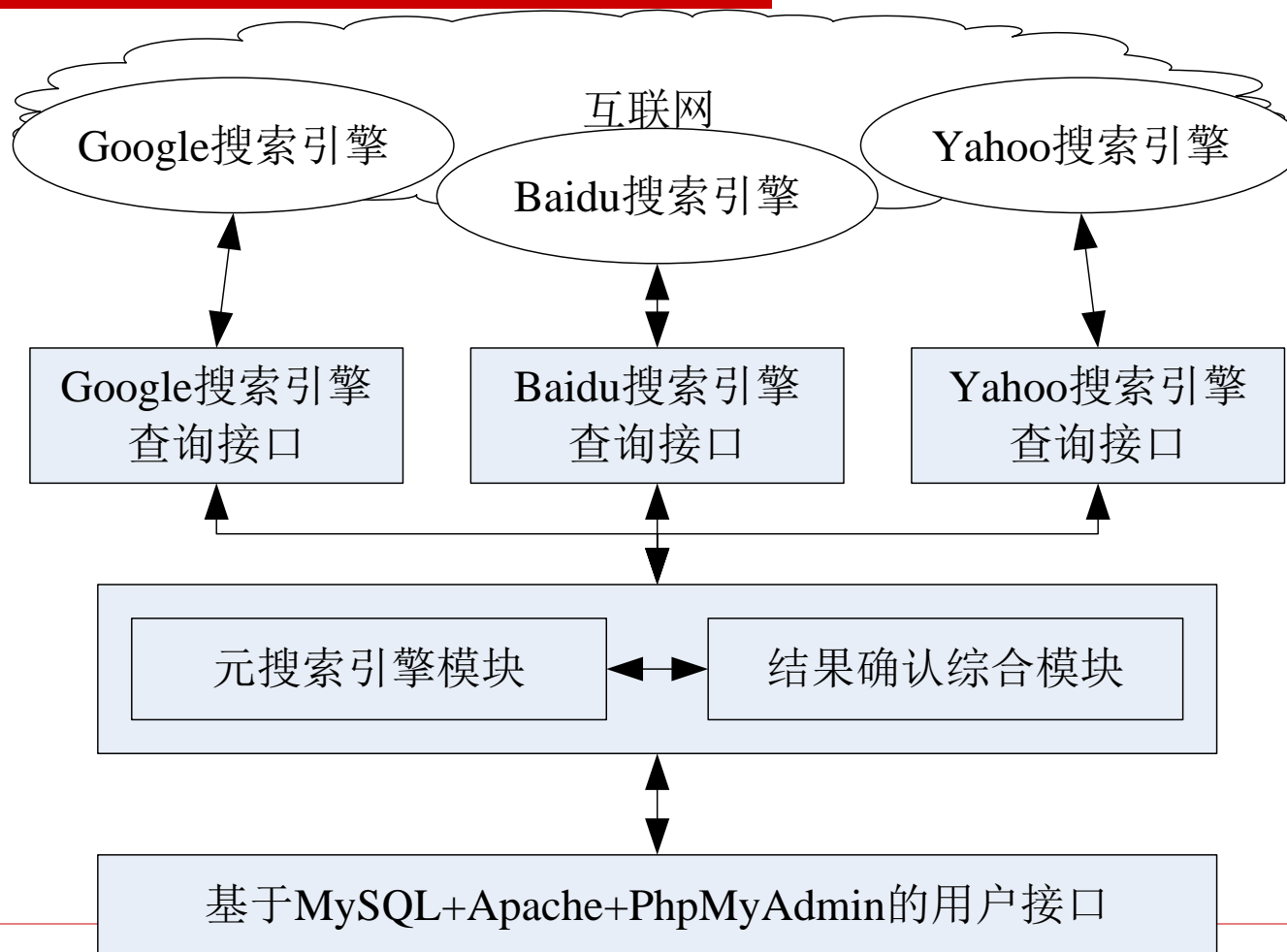
上海碧泉化工实业有限公司 产品列表

Hacked By d3triment@l. Hacked By d3triment@l. Hacked By d3triment@l. Hacked By

2011年3月6日

Copyright (c) 2008-2009 诸葛建伟

基于元搜索技术的被黑网站发现



利用元搜索技术进行黑客调查剖析实验

□ 被篡改网站发现

- 百度, **Google**, **yahoo**
- 元搜索标题中包含**"hacked by"**
- 并下载搜索结果页面确认

□ 实验结果（1个月）

- 共发现**423**个被篡改网站
- 对攻击者进行了进一步元搜索
- 验证了元搜索技术的有效性

表 1 实验中发现的被篡改网站域名分布

Tab.1 The domain distribution of defaced websites tracked during the experiments

域名	发现被黑网站数量	所占比例
.cn	126	29.8%
.com.cn	126	29.8%
.gov.cn	112	26.5%
.org.cn	29	6.9%
.edu.cn	17	4.0%
.net.cn	13	3.1%
总计	423	100%



对黑客的搜索

表 4 chinahacker/starkun@D.H.T 的元搜索调查剖析档案

Tab.4 The profile of chinahacker/starkun@D.H.T by analyzing the web pages tracked by meta-searching tool

基本信息	人物调查档案
黑客代号	藍軒星坤
其他代号	starkun@D.H.T, chinahacker@D.H.T, Red Devils Crew@D.H.T, Redfreedom@HUC
Email 地址	starkun@whnews.cn
即时通讯	QQ:85926628, msn:dht@dhthacker.com
博客	http://blog.iweihai.cn/blog.asp?name=藍軒星坤 http://85926628.qzone.qq.com/
所在团队	红客联盟 HUC(2001-2003) -> D.H.T (2004-)
所属社群	灰帽子社群/计算机朋克
黑客动机	地缘政治诱因, 进入社会团体, 社区名声地位
擅长攻击方式	网站入侵、网站篡改 2001 年中国红客联盟核心成员, 参与“中美黑客大战”; 2003 年左右创建黑白网络(heibai.net), 现已易主;




对黑客的搜索(“人肉”)

黑客历程	2004 年遭中国黑客联盟通缉，知名度提升； 2004 年开始组建中国 D.H.T 黑客团队，2005 年对国外网站发动密集型攻击， 2005 年 9 月在 zone-h.org 站点统计中高居榜首，2006 年正式招收成员； 2006 年左右发布《再见了网络 这里不是我该呆的地方(一个黑客的话语)》； 2008 年 7 月左右入侵中国外交部网站，至今仍活跃地进行网站篡改等攻击。
撰写文章	2003 年《我是一个黑客》长篇连载及题记； 2004 年《一次详细全面的入侵》、《IPC\$入侵的高级手段与方法》等； 2007 年《QQ 密码忘了，制作自己的破解器》等； 2008 年《我对黑客技术的思考》等。

□ 真实身份？

对黑客的搜索 (3)

真实身份		
姓名		
性别	男	
出生年	1986年左右	
工作单位		
职业	程序员	
居住地	山东威海	
联系电话		
照片		http://blog.iweihai.cn/blog.asp?name=%CB[%DC%8E%D0%C7%C0%A4&si
毕业学校		
	2008年10月4日结婚	
DHT Logo	中国DHT网络安全应急小组	
	真正的黑客风范，低调不张扬，真正的“黑亦有道”不能仇视社会，不能给别人制造麻烦，不能给别人带来损失	
	<div> <div> <div>蓝轩星坤</div> <div>捍卫祖国主权， 维护正义和平， 抵抗外来侵扰， 打击反动势力， 祖国繁荣昌盛而不懈努力</div> </div> <div>  </div> </div>	

2011年3月6日



内容

1. 安全模型-P2DR模型

2. P: 防御技术

3. D: 检测技术

4. R: 响应技术

**5. 作业6: 分析蜜网网关中防火墙与
入侵检测系统配置规则**



作业6内容（团队作业:10+2）

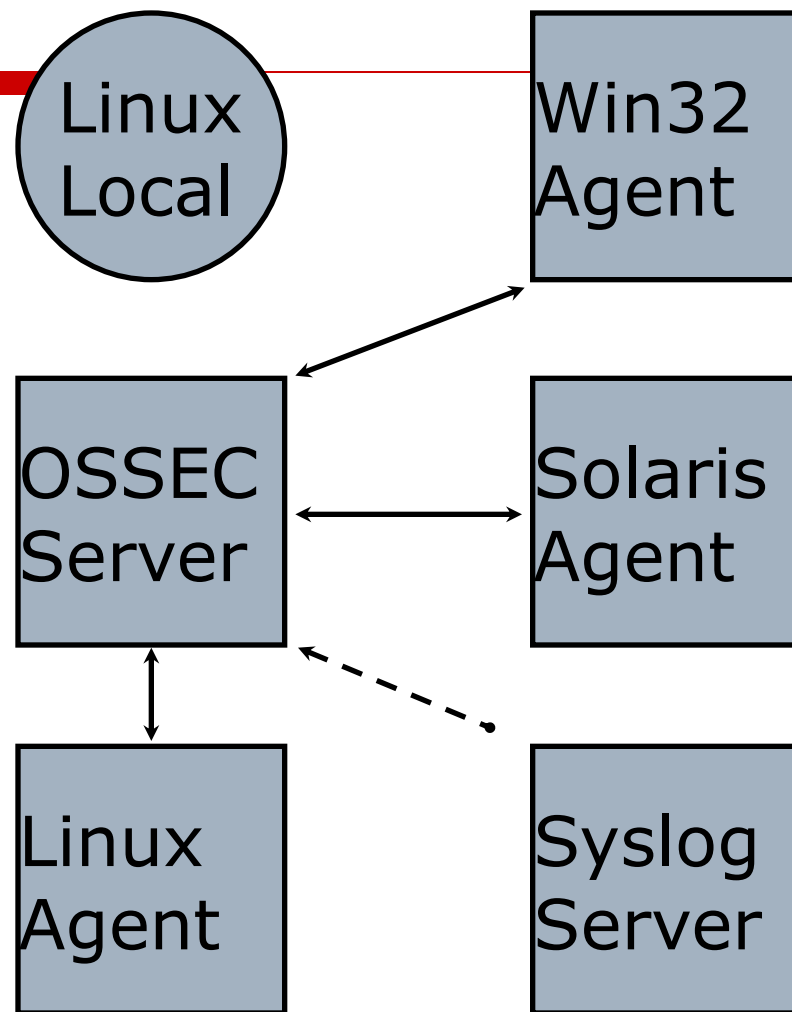
- 分析蜜网网关**ROO**中防火墙和**IDS/IPS**配置规则，编写分析报告
 - 防火墙: **netfilter+IPTables**
 - **/etc/init.d/rc.firewall**
 - 入侵检测系统: **snort**
 - **/etc/init.d/snortd**
 - **/etc/snort/snort.conf**
 - 入侵防御系统: **snort_inline**
 - **/etc/init.d/hw-snort_inline**
 - **/etc/snort_inline/snort_inline.conf**
 - 分析内容:
 - 上述脚本是如何实现蜜网网关的数据捕获和数据控制机制？
 - 获取**IPTables**的实际规则列表、**Snort**和**Snort_inline**的实际执行参数
 - 蜜网网关开机之后，防火墙、**NIDS**、**NIPS**是如何启动的？
 - **Bonus(2分)**: 蜜网网关中的**Snort**规则是如何自动升级的？
- 提示：参考课程**2**中给出的蜜网网关数据控制、数据捕获机制的讲解
- **Deadline**时间点: **11月17日下午17:00**

Thanks

诸葛建伟
zhugejw@gmail.com

OSSEC结构

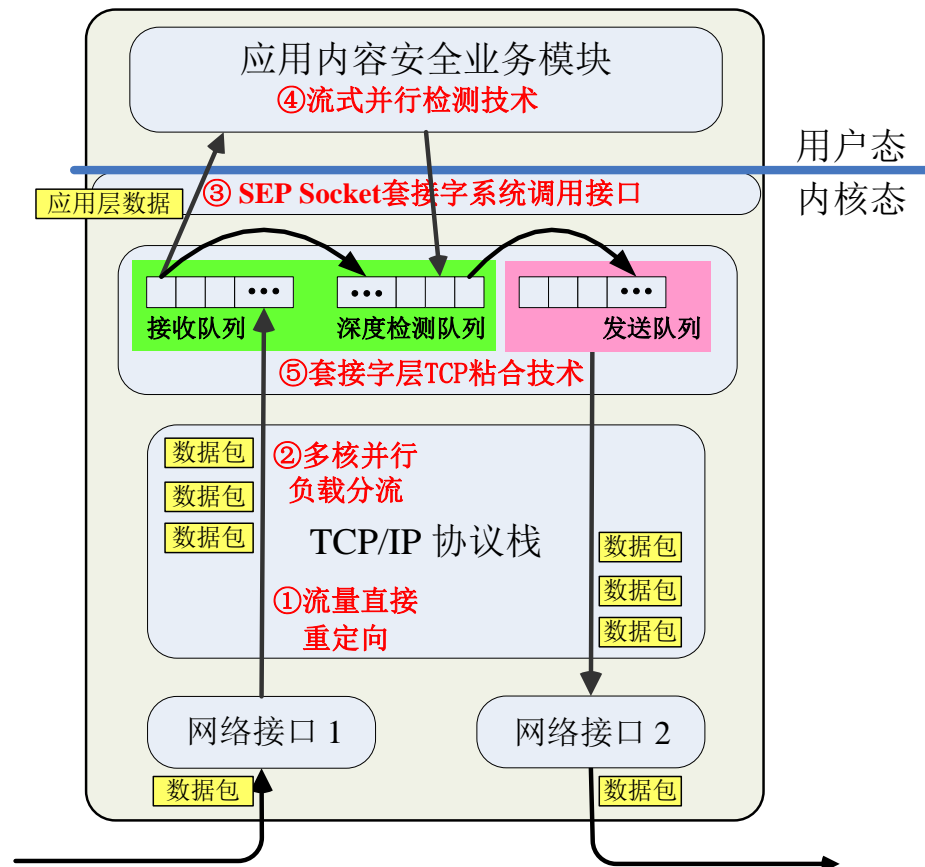
- Installation Types
 - Server
 - Agent
 - Local
- Connection Types
 - Secure
 - Syslog
- Non-Monitored Machines via Syslog
- Communications Security (PSK)



防火墙发展趋势

-透明代理&深度报文检测

- **SEP Socket (透明代理性能优化)**
 - 系统网络协议栈和套接字扩展，流量直接重定向
 - 混杂连接粘合方法，避免内核至用户态数据复制
 - 多核平台上的并行优化方法
- **深度报文检测(DPI)**
 - 应用服务精确识别
 - 报文深度协议分析
 - 协议规范异常检测
 - 协议滥用检测





主机防御

- 补丁！补丁！还是补丁！
 - 及时更新操作系统补丁
 - 常用应用软件的升级 – **360**软件管理
- 安全配置
 - 在安全性和易用性之间取得平衡
 - 禁止自动播放等常用攻击通道
- 主机防御软件
 - 个人防火墙
 - 主机防病毒软件
 - 安全诊断和恢复软件
- 安全意识与习惯
 - 不访问、下载或安装来历不明的文件/邮件/网页...
 - 慎用非可信的**U**盘，只读使用时设置**Lock**



防病毒软件

VirusTotal 是一款由独立的 IT 安全实验室 Hispasec Sistemas 所开发的服务. 它使用多种反病毒引擎的命令行版本, 并定期从对应的开发者更新官方的病毒定义库.

这份列表为参加 **VirusTotal** 服务的公司(和它们的反病毒引擎).

□ 防病毒软件

- 国外著名产品: 卡巴斯基, 诺顿, **Macfee, Eset, ...**
- 国内产品: 瑞星, 金山, 江民, ...

□ 防病毒软件的一般构成

- 查毒引擎
 - 脱壳引擎
 - 病毒特征码匹配引擎
 - 基于行为的检测引擎
- 病毒库: 需要实时在线更新

□ 在线查毒: **Virustotal**

- [AhnLab](#) (V3)
- [Aladdin](#) (eSafe)
- [ALWIL](#) (Avast! Antivirus)
- [Authentium](#) (Command Antivirus)
- [AVG Technologies](#) (AVG)
- [Avira](#) (AntiVir)
- [Bit9](#) (FileAdvisor)
- [Cat Computer Services](#) (Quick Heal)
- [ClamAV](#) (ClamAV)
- [CA Inc.](#) (Vet)
- [Doctor Web, Ltd.](#) (DrWeb)
- [Eset Software](#) (ESET NOD32)
- [ewido networks](#) (ewido anti-malware)
- [Fortinet](#) (Fortinet)
- [FRISK Software](#) (F-Prot)
- [F-Secure](#) (F-Secure)
- [G DATA Software](#) (GData)
- [Hacksoft](#) (The Hacker)
- [Hauri](#) (ViRobot)
- [Ikarus Software](#) (Ikarus)
- [K7 Computing](#) (K7AntiVirus)
- [Kaspersky Lab](#) (AVP)
- [McAfee](#) (VirusScan)
- [Microsoft](#) (Malware Protection)
- [Norman](#) (Norman Antivirus)
- [Panda Security](#) (Panda Platinum)
- [PC Tools](#) (PCTools)
- [Prevx](#) (Prevx1)
- [Rising Antivirus](#) (Rising)
- [Secure Computing](#) (SecureWeb)
- [BitDefender GmbH](#) (BitDefender)
- [Sophos](#) (SAV)
- [Sunbelt Software](#) (Antivirus)
- [Symantec](#) (Norton Antivirus)
- [VirusBlokAda](#) (VBA32)
- [Trend Micro](#) (TrendMicro)
- [VirusBuster](#) (VirusBuster)



安全诊断和恢复类软件

□ 安全诊断类软件

- **SReng/HijackThis**
- 安全卫士**360**/瑞星卡卡
- 未名**BBS Virus_Security**版

□ 恢复类软件

- **Norton Ghost**
- 影子系统(**PowerShadow**)
- 一键还原
- ...