电子科技大学

University of Electronic Science and Technology of China

第9章 入侵检测

为什么需要IDS

- IDS：防火墙之后第二道防线
- 单一防护产品的弱点或局限
  - 防御方法和防御策略的有限性
  - 网络环境动态多变
  - 威胁来自外部和内部
- 防火墙局限
  - 网络边界设备，只能抵挡外部入侵；据统计安全威胁80%来自内部
  - 自身存在弱点，可被攻破（穿透）或绕开
  - 对某些攻击保护很弱
  - 仅能拒绝非法连接請求，合法使用者仍可能非法使用系统，甚至提升自己的权限
  - 对于入侵者的攻击行为仍一无所知

为什么需要IDS

- 入侵很容易
  - 入侵教程随处可见
  - 各种工具唾手可得
- 传统信息安全方法
  - 采用严格访问控制和数据加密策略来防护，在复杂系统中不充分。
  - 系统安全不可缺的部分，但不能完全保证系统的安全
- 网络安全模型：需要防护，也需要检测

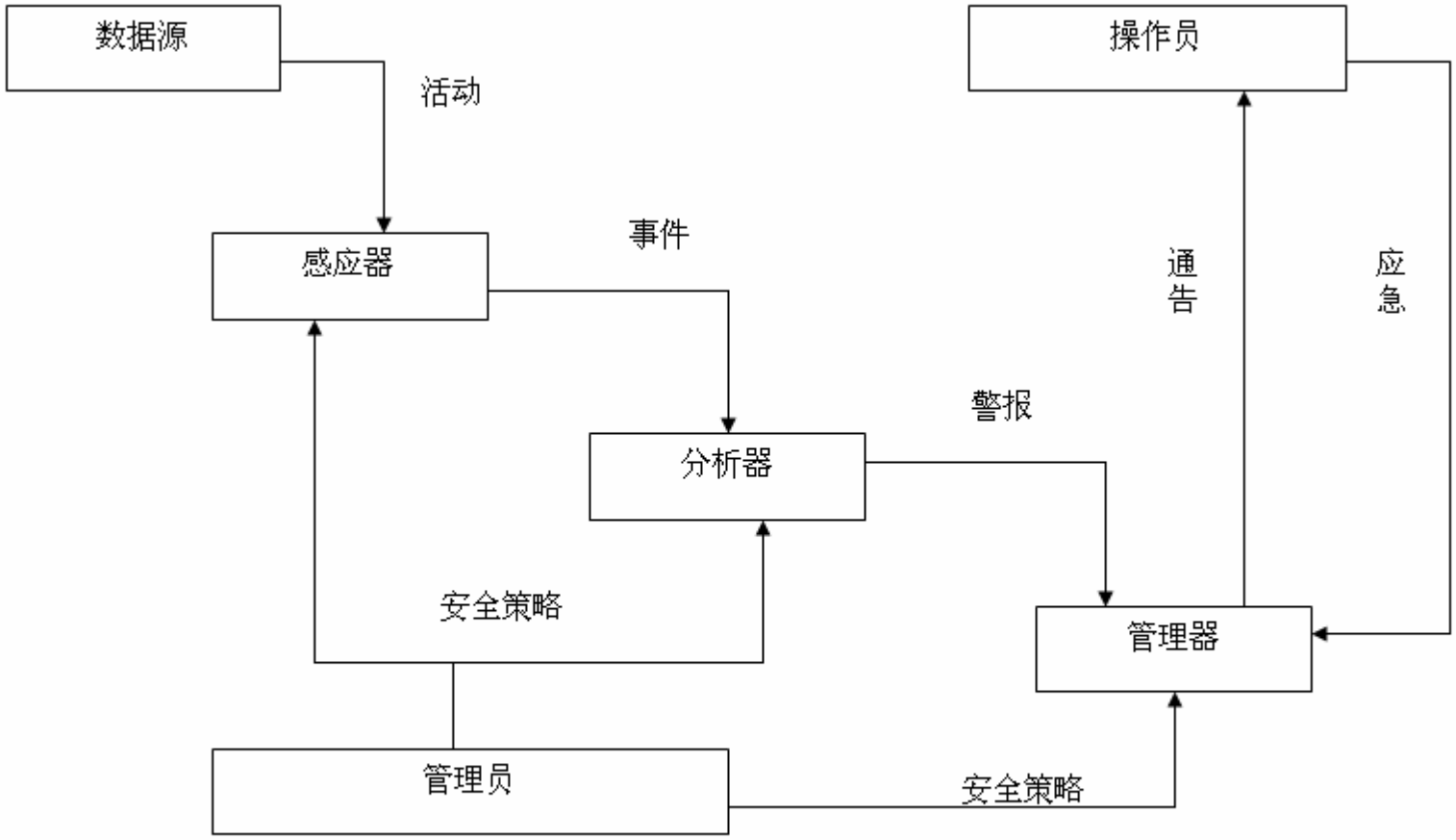
网络安全工具的特点

	优点	局限性
防火墙	可简化网络管理，产品成熟	无法处理网络内部的攻击
IDS	实时监控网络安全状态	误警率高，缓慢攻击，新的攻击模式
Scanner	了解网络现有的安全水平，简单可操作，帮助系统管理员和安全服务人员解决实际问题，	并不能真正了解网络上即时发生的攻击
VPN	保护公网上的内部通信	加密解密
防病毒	针对文件与邮件，产品成熟	功能单一

入侵检测

- 对入侵行为的发觉：
  - 对网络和系统的运行状态进行监视
    - 从网络或系统关键点收集信息并进行分析
  - 从中发现网络和系统中是否有违反安全策略的行为和被攻击的迹象
    - 攻击企图、攻击行为或者攻击结果
  - 以保证系统资源的机密性、完整性和可用性
- 入侵检测系统IDS（Intrusion Detection System）：
  - 进行入侵检测的软件与硬件组合

IDS系统原理



入侵检测系统原理图

IDS基本结构

- 入侵检测系统包括三个功能部件
  - 信息收集
  - 信息分析
  - 结果处理

信息收集

- 在网络或系统的若干不同关键点（不同网段和不同主机）收集
  - 从一个来源的信息有可能看不出疑点
  - 尽可能扩大检测范围
- 入侵检测很大程度上依赖于收集信息的可靠性和正确性
  - 要保证信息收集软件（组件）的完整性
  - 入侵检测系统软件本身应具有相当强的坚固性，防止被篡改而收集到错误的信息

信息收集的来源

- 系统或网络的日志文件
- 网络流量
- 系统目录和文件的异常变化
- 程序执行中的异常行为

系统或网络的日志文件

- 系统或网络日志文件常留下攻击者踪迹
- 日志文件记录各类行为，每类日志包含不同信息
  - 如“用户活动”类日志：包含登录、用户ID改变、用户对文件的访问、授权和认证信息等内容
  - 不正常或不期望的行为：重复登录失败、登录到不期望的位置以及非授权的企图访问重要文件等

系统目录和文件的异常变化

- 网络环境下的文件系统中的重要信息文件和私有数据文件经常是黑客修改或破坏的目标
- 目录和文件不期望的改变
  - 如修改、创建和删除，特别是那些正常情况下限制访问的，
  - 很可能就是一种入侵产生的指示和信号
- 入侵者经常替换、修改和破坏他们获得访问权的系统上的文件
  - 同时为了隐藏系统中他们的表现及活动痕迹，会尽力去替换系统程序或修改系统日志文件

信息分析方法

- 误用检测（模式匹配）
- 统计分析（异常检测）
- 完整性分析

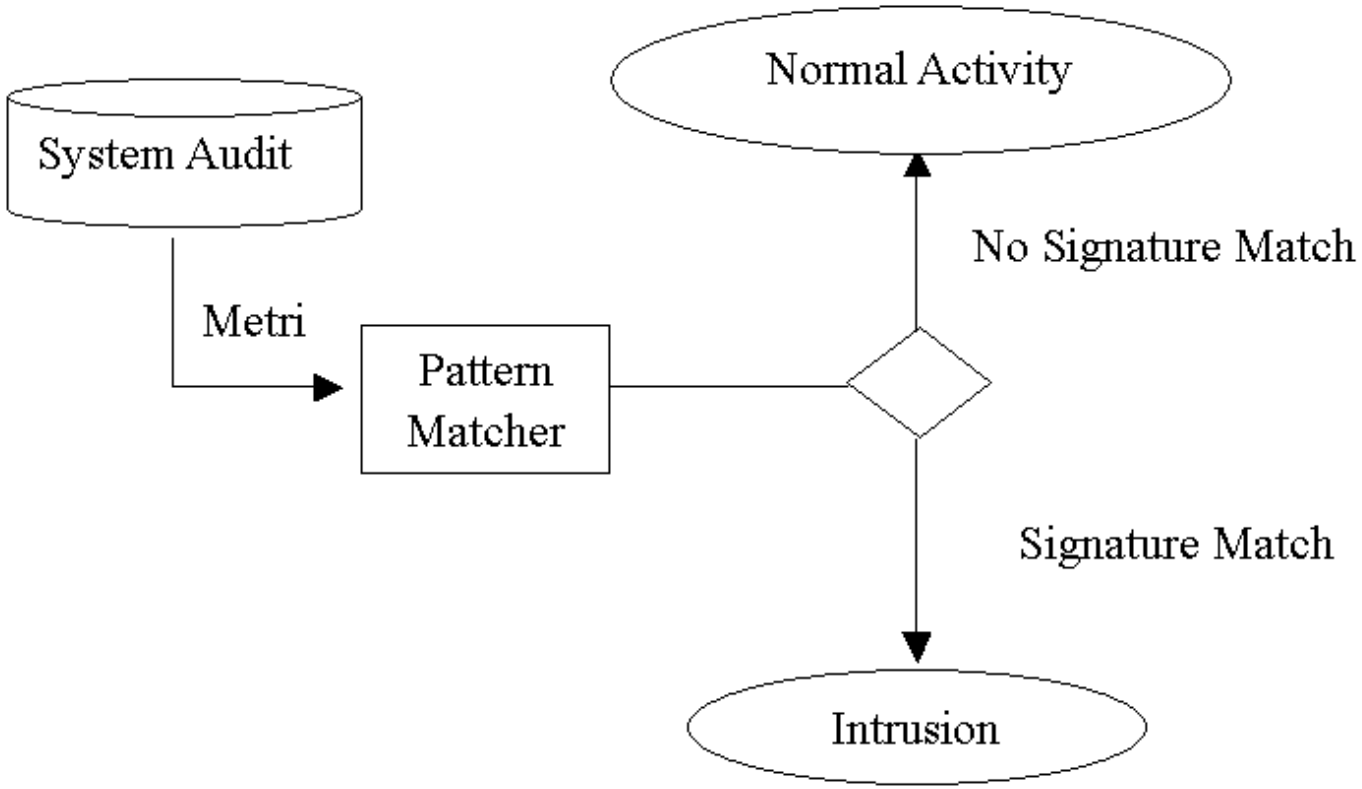
入侵检测性能关键参数

- 误报(false positive)：
  - 错误将正（异）常活动定义为入侵
- 漏报(false negative)：
  - 未能检测出入侵行为

模式匹配

- 误用检测模型（Misuse Detection）：
  - 收集非正常操作的行为特征，建立（入侵或攻击）特征库（误用模式数据库）
  - 监测用户或系统行为与特征库中记录匹配（指纹识别，从而发现违背安全策略的行为
- 一般来讲，一种攻击模式可以用一个过程（如执行一条指令）或一个输出（如获得权限）来表示。
  - 该过程可以很简单（如通过字符串匹配以寻找一个简单的条目或指令），
  - 也可以很复杂（如利用正规的数学表达式来表示安全状态的变化）

误用检测模型



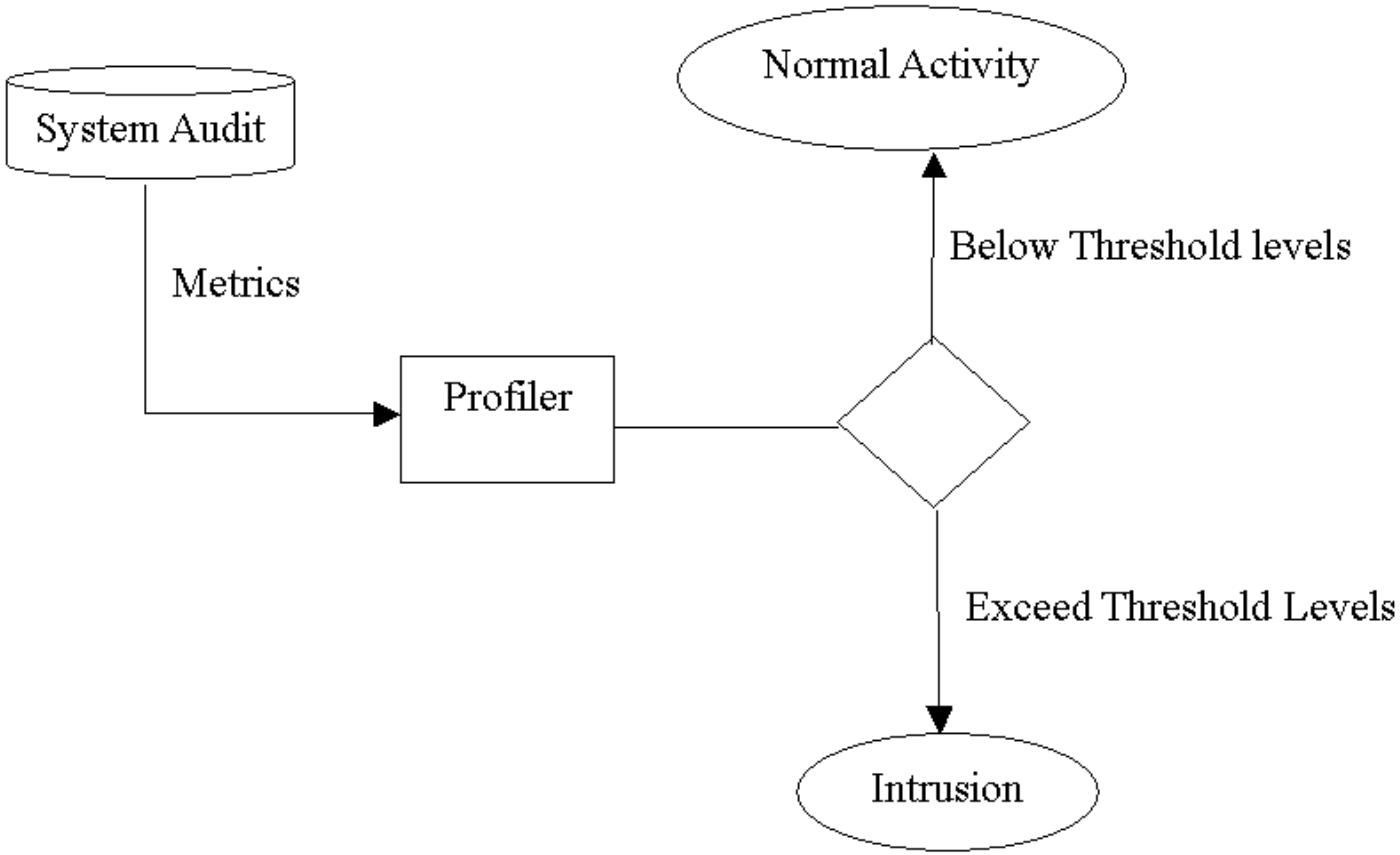
误用检测模型特点

- 误报：特征与正常的用户行为匹配
- 漏报：没有特征与某种新的攻击行为匹配
- 能明显降低误报率，但漏报率随之增加。
- 攻击特征的细微变化，会使得误用检测无能为力

统计分析

- 异常检测模型（Anomaly Detection ）：
  - 给系统对象（如用户、文件、目录和设备等）**创建统计描述，**统计正常操作应具有特征（用户轮廓）。
    - 定时采样系统正常使用时的一些测量属性，包括会话登录、退出、CPU和内存占用，硬盘使用，访问次数、操作失败次数和延时等
  - 统计测量属性的平均值和偏差被用来与网络、系统行为进行比较，用户活动与正常行为有重大偏离（观察值在正常值范围之外），就视为入侵 。

异常检测模型



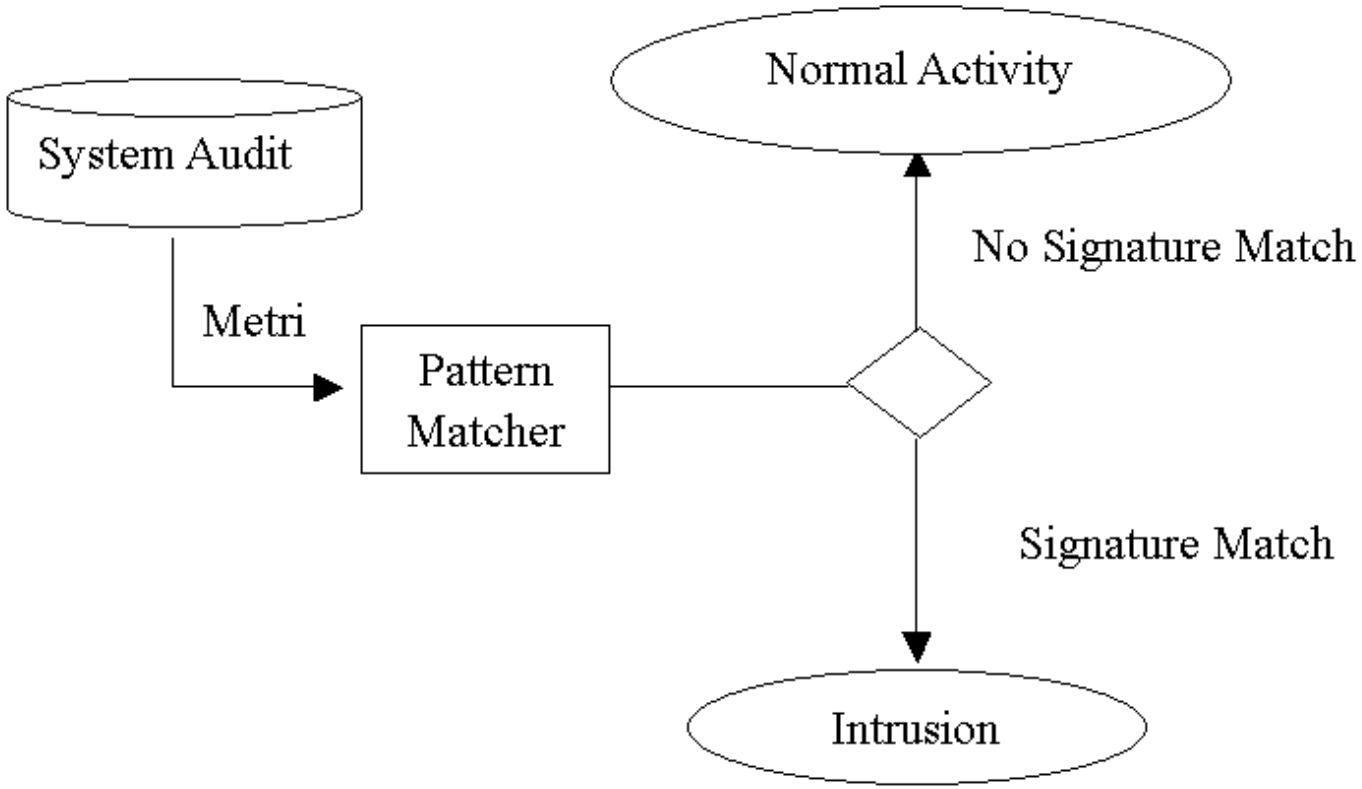
异常检测模型特点

- 效率：取决于用户轮廓完备性和监控频率
- 能有效检测未知入侵，不需要对每种入侵行为进行定义
- 能针对用户行为的改变进行自我调整和优化
- 随着检测模型的逐步精确，会消耗更多系统资源

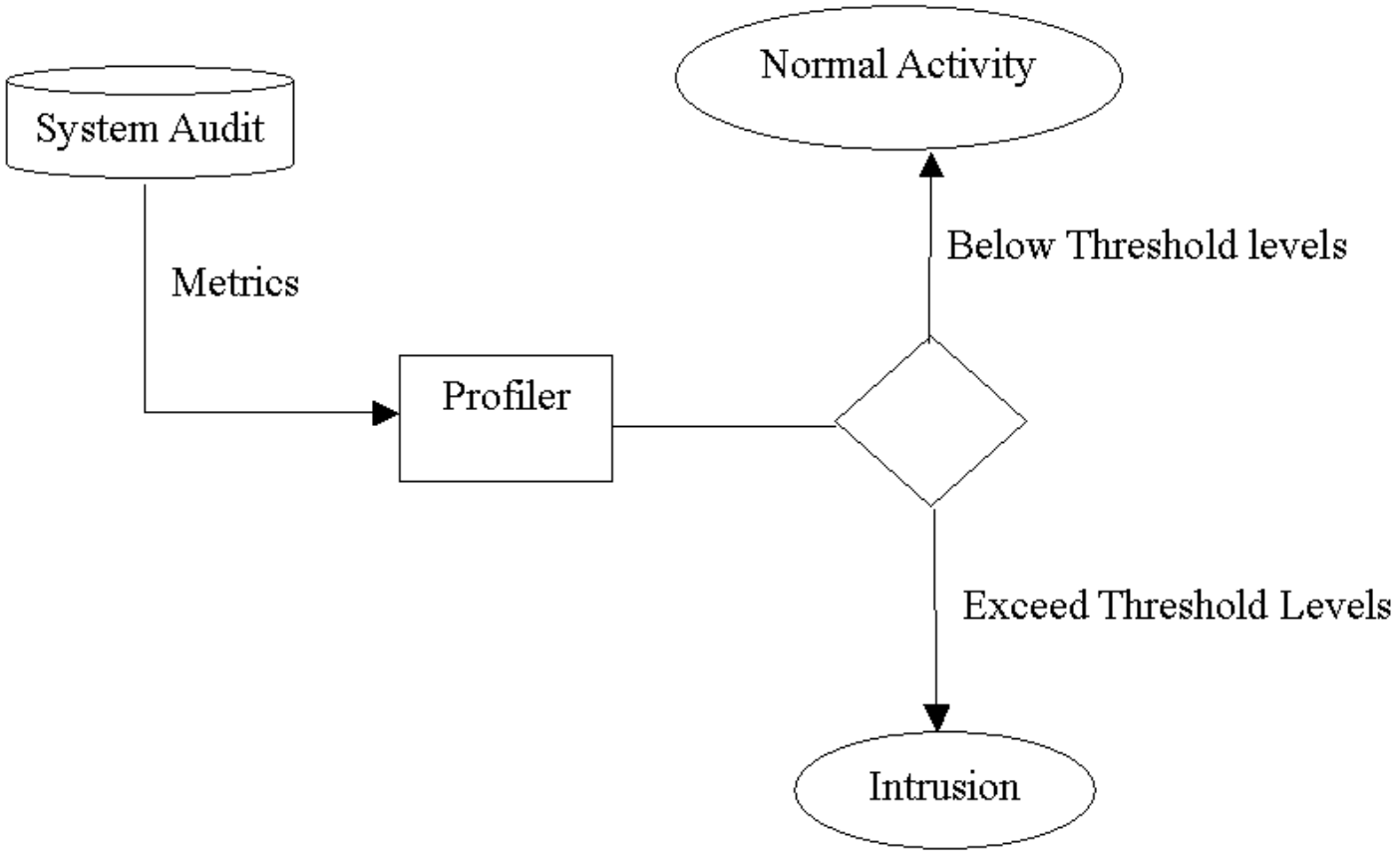
完整性分析

- 主要关注某个文件或对象是否被更改
  - 常包括文件和目录的内容及属性
- 在发现被更改的、被安装木马的应用程序方面特别有效
- 往往用于事后分析

温故而知新——误用检测模型



温故而知新——异常检测模型



结果处理–入侵响应

- 策略
- 方式

制订响应策略应考虑的要素

- 系统用户：
  - 可分为网络安全专家或管理员、系统管理员、安全调查员三类
  - 不同人员对系统的使用目的、方式和熟悉程度不同，必须区别对待
- 操作运行环境：
  - 入侵检测系统提供的信息形式依赖其运行环境
- 系统目标：
  - 为用户提供关键数据和业务的系统，需要部分地提供主动响应机制
- 规则或法令的需求：
  - 在某些军事环境里，允许采取主动防御甚至攻击技术来对付入侵行为

响应方式

- 被动响应
  - 记录安全事件
  - 产生报警信息
  - 记录附件日志
  - 激活附加入侵检测工具
- 温和主动响应
  - 隔离入侵者IP
  - 禁止被攻击对象的特定端口和服务
  - 隔离被攻击对象
- 严厉主动响应
  - 警告攻击者
  - 跟踪攻击者
  - 断开危
  - 攻击攻击者

与其他安全产品交互

入侵检测分类

- 检测方法
- 数据来源
- 系统架构
- 时效性

入侵检测的分类（1）

- 按照分析（检测）方法
  - 异常检测模型（Anomaly Detection，统计分析）：
  - 误用检测模型（Misuse Detection，模式匹配）：

入侵检测的分类（2）

- 按照数据来源：
  - 基于主机：
    - 监控主机上的活动，以该主机为保护目标
  - 基于网络：
    - 数据来源是网络传输数据包，保护网络的运行
  - 混合型

基于主机

- 验证操作系统与应用调用及检查日志文件、文件系统信息与网络连接。
- 监视与分析主机的审计记录
- 可不运行在监控主机上
- 实时性：能否及时采集到审计记录
- 保护攻击目标主机审计系统

基于网络

- 对网络通信数据进行侦听采集数据
  - 共享式
  - 非共享网络（交换式）
- 主机资源消耗少
- 提供对网络通用的保护
- 如何适应高速网络环境

两类IDS监测软件

- 网络IDS
  - 侦测速度快
  - 隐蔽性好
  - 视野更宽
  - 较少的监测器
  - 占资源少
- 主机IDS
  - 视野集中
  - 易于用户自定义
  - 保护更加周密
  - 对网络流量不敏感

入侵检测的分类（3）

- 按系统各模块的运行方式
  - 集中式：
    - 系统模块，包括数据收集、分析集中在一台主机上运行
  - 分布式：
    - 系统模块分布在不同的计算机和设备上

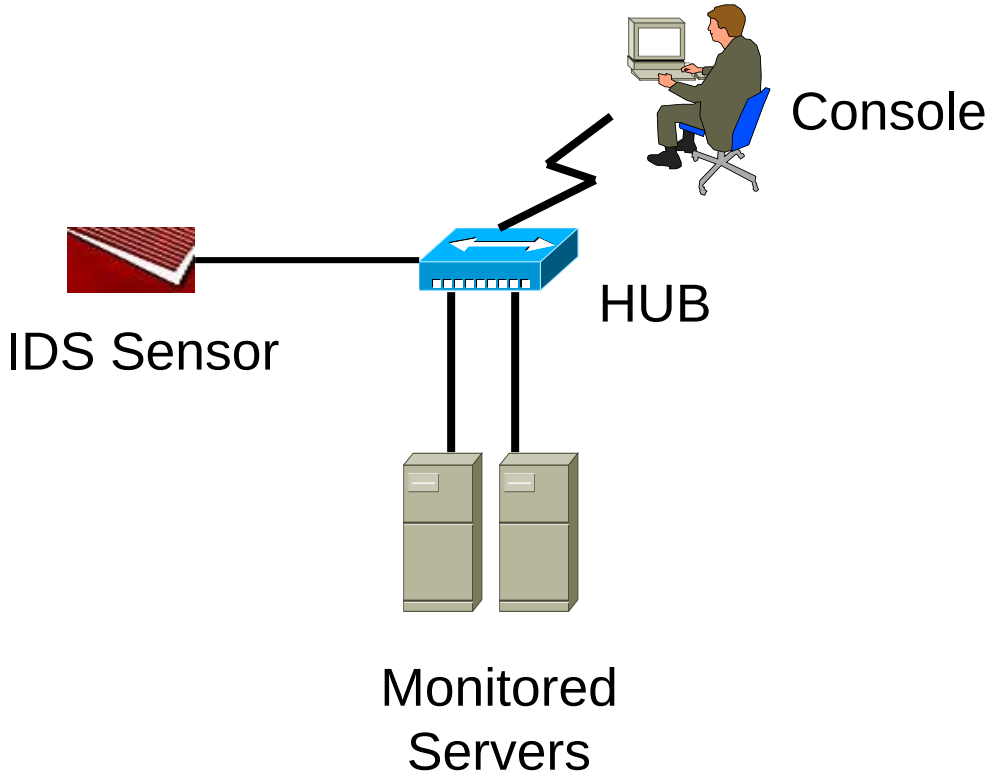
入侵检测的分类（4）

- 根据时效性
  - 脱机分析：
    - 入侵行为发生后，对产生的数据进行分析
  - 联机分析：
    - 在数据产生的同时或者发生改变时进行分析

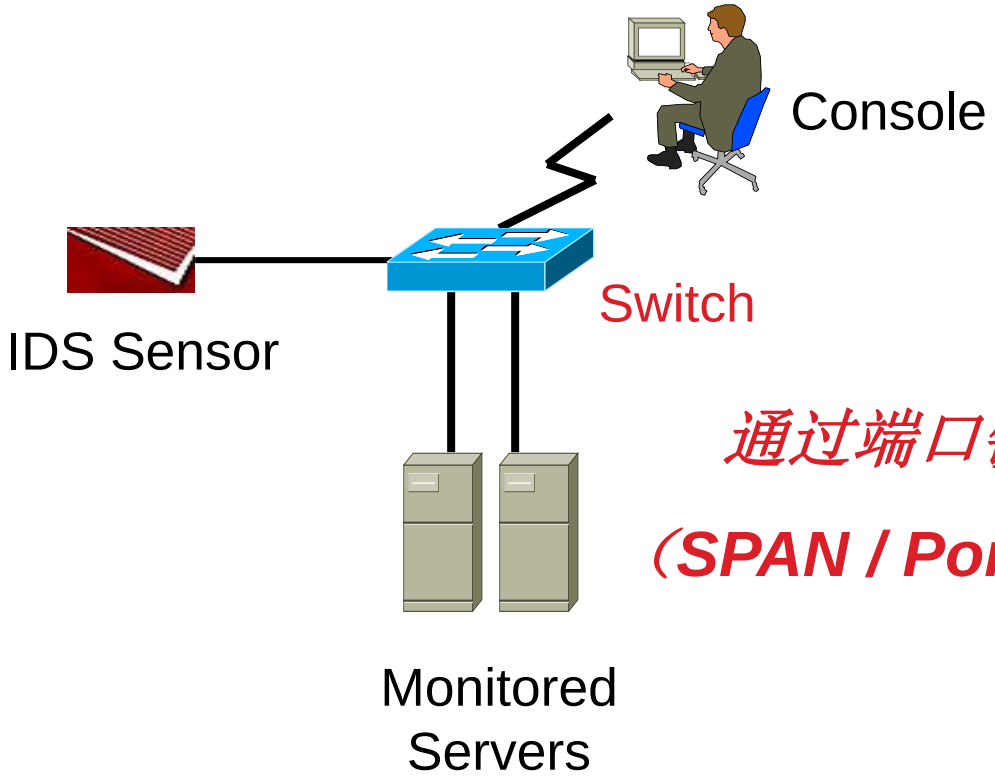
部署

- NIDS的位置必须要看到所有数据包
  - 共享媒介HUB
  - 交换环境
- 分布式结构
  - Sensor
  - Console

共享媒介



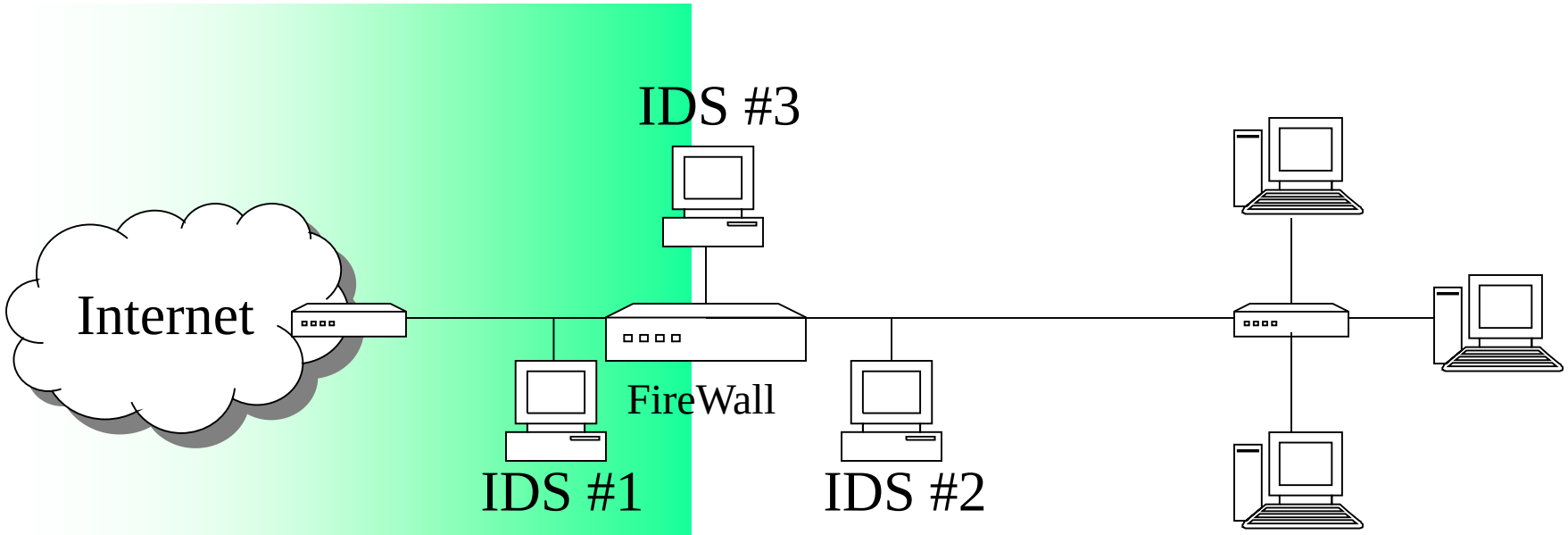
交换环境



通过端口镜像实现

（SPAN / Port Monitor）

Deployment of IDS

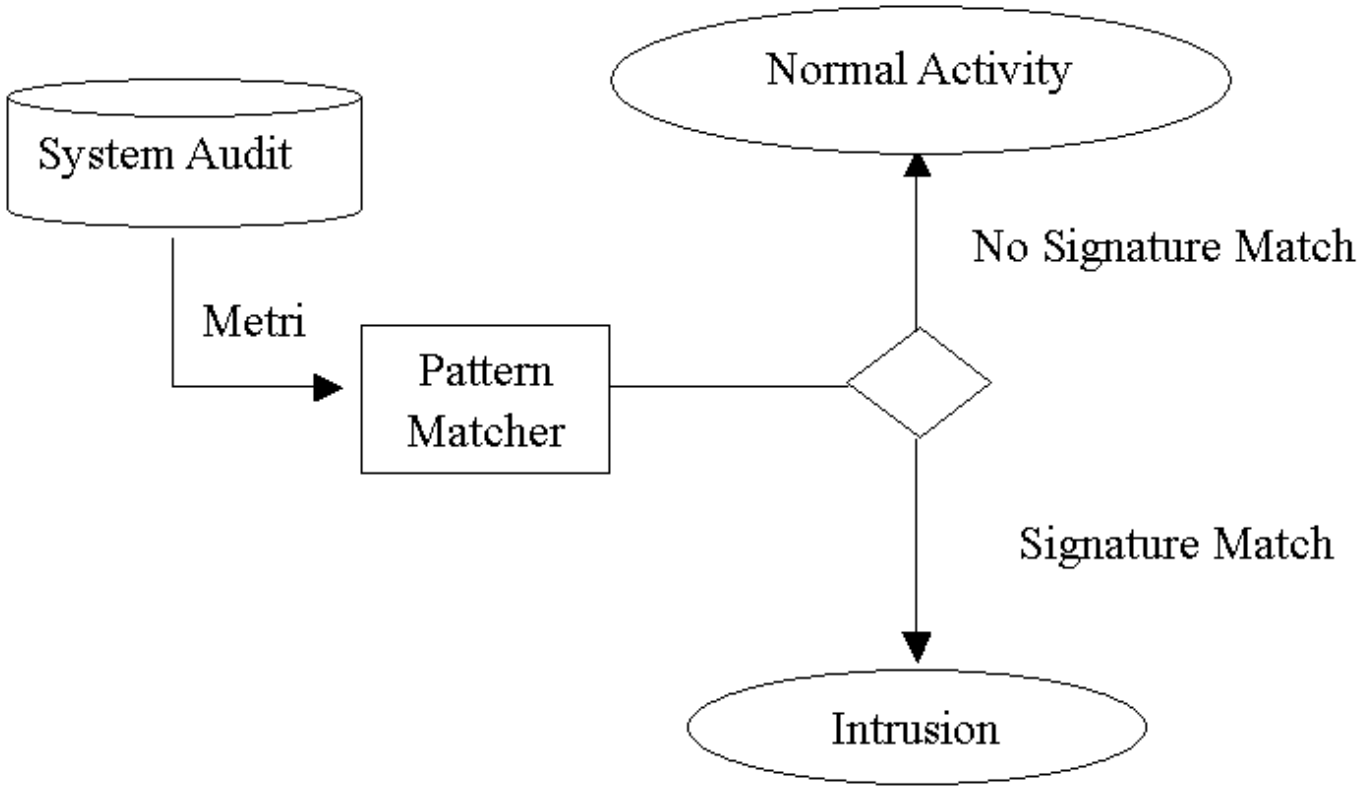


IDS#1 Monitor of External Traffic

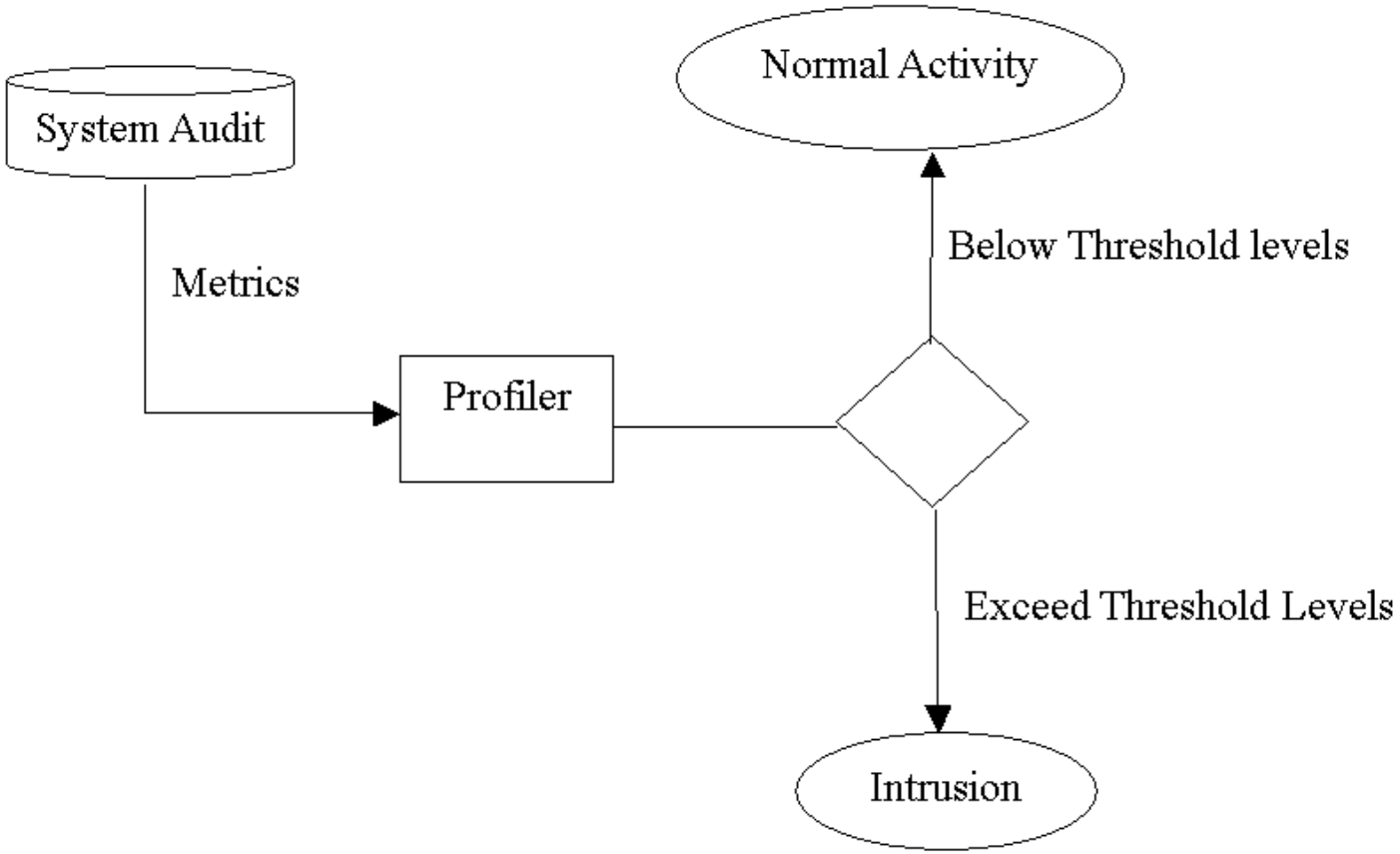
IDS#2 Monitor of Internal Traffic

IDS#3 Monitor of Firewalls

温故而知新——误用检测模型



温故而知新——异常检测模型



Snort

- C语言编写的开放源代码软件，作者为Martin Roesch。
- 跨平台、轻量级的误用网络入侵检测软件
- 基于libpcap的网络数据包嗅探器和日志记录工具。

Snort简介

- 由三个重要的子系统构成：
  - 数据包解码器
  - 检测引擎
  - 日志与报警系统

Snort规则

- 简单但灵活、高效的规则描述语言表述检测规则，两个逻辑部分：
- 规则头（Rule Header）
  - 规则动作（Alert、Log、Pass）
  - 协议、IP地址、端口号
  - 方向操作符“->”或“<-”
- 规则选项（Rule Options）
  - Snort系统入侵检测引擎的核心部分
  - 当前有三十几种关键字（msg、log、ttl、id、content、flags、seq等）
  - 包含一个警告消息和某数据包有关部分的信息。

规则示例

alert tcp any any -> 192.168.1.0/24 143 (content: "|90C8 C0FF FFFF|/bin/sh"; msg: "IMAP buffer overflow!";)

alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg: "mountd access";)

- 圆括号前：规则头部
- 圆括号内：规则选项
- 关键字“content”允许用户设定规则，搜索数据包有效负荷内的特定内容，并激发一个响应。

Snort规则集

- 以应用层协议分类的.rules文本文件，比如：
  - dos.rules文件存放拒绝服务攻击类的规则；
  - ftp.rules文件存放FTP服务相关的规则；
  - telnet.rules文件存放FTP服务相关的规则。
- 规则文件中每行定义一种攻击检测
  - alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP SITE CPWD overflow attempt"; flow:established,to_server; content:"SITE "; nocase; content:" CPWD "; nocase; content:!"|0a|"; within:100; reference:bugtraq,5427; reference:cve,CAN-2002-0826; classtype:misc-attack; sid:1888; rev:3;)

入侵检测面临的问题

- 1、随着能力提高，入侵者会研制更多攻击工具，以及使用更为复杂精致的攻击手段，对更大范围的目标类型实施攻击；
- 2、入侵者采用加密手段传输攻击信息；
- 3、日益增长的网络流量导致检测分析难度加大；
- 4、缺乏统一的入侵检测术语和概念框架；

面临的问题

- 5、不适当的自动响应机制存在着巨大的安全风险；
- 6、存在对入侵检测系统自身的攻击；
- 7、过高的错报率和误报率，导致很难确定真正的入侵行为；
- 8、采用交换方法限制了网络数据的可见性；
- 9、高速网络环境导致很难对所有数据进行高效实时分析

发展方向

- 1、更有效集成各种入侵检测数据源，包括从不同系统和不同传感器上采集的数据，提高报警准确率；
- 2、在事件诊断中结合人工分析，提高判断准确性；
- 3、提高对恶意代码的检测能力，包括email攻击，Java，ActiveX等；
- 4、采用一定的方法和策略来增强异种系统的互操作性和数据一致性；

发展方向

- 5、研制可靠的测试和评估标准；
- 6、提供科学的漏洞分类方法，尤其注重从攻击客体而不是攻击主体的观点出发；
- 7、提供对更高级的攻击行为如分布式攻击、拒绝服务攻击等的检测手段；

入侵防护系统（IPS）

- 防火墙与网络入侵检测技术的结合
- 位于网络主干位置，一般以透明网关形式存在，所有进出流量均需通过
- 基于IDS实现网络防护，阻断攻击
  - 使用IDS对数据包进行分析，对高层应用协议数据进行重组与协议追踪。
  - 处理存在问题的数据包并关闭相应连接。

IPS优点

- 实时阻断网络攻击
- 隐蔽数据检测，对通信双方透明
- 主干检测，避免绕过
- 透明模式，不会对网络拓扑造成影响

IPS不足

- 分析效率低，无法适应高速网络环境
- 继承IDS误报问题，易造成正常网络通信的影响，为减少误报，漏报增多
- 无法检测加密数据

IDS标准化工作

- 随着网络规模扩大，网络入侵方式、类型、特征各不相同，入侵活动变得复杂而又难以捉摸
- 网络安全要求IDS之间能够相互协作，能够与访问控制、应急、入侵追踪等系统交换信息，形成一个整体有效的安全保障系统
- 需要一个标准或规范来加以指导，系统之间要有一个约定

CIDF

- (The Common Intrusion Detection Framework) http://www.gidos.org/drafts
- 开放组织：
  - CIDF早期由美国国防部高级研究计划局赞助研究，现由CIDF工作组负责。已成为一个开放、共享资源
- 一套规范：定义了IDS表达检测信息的标准语言以及IDS组件之间的通信协议
  - 使IDS相互通信，共享检测信息，协同工作
  - 还可与其它系统配合实施统一的配置响应和恢复策略
- CIDF构建分布式IDS的基础
  - 集成各种IDS，使之协同工作，实现各IDS之间的组件重用

CIDF规格文档

- 由四部分组成，分别为：
  - 体系结构：阐述了一个标准IDS的通用模型
  - 规范语言：定义了一个描述各种检测信息的标准语言
  - 内部通讯：定义了IDS组件之间进行通信的标准协议
  - 程序接口：提供了一整套标准的应用程序接口

通信层次

- CIDF将各组件间的通信划分为三个层次结构：
  - GIDO层（GIDO layer）
    - 负责对传输信息的格式化，统一信息表达格式，以便各IDS间互操作
  - 消息层（Message layer）
    - 负责对传输的信息进行加密认证，然后可靠地从源传输到目的地，建立一个可靠的传输通道
  - 传输层（Negotiated Transport layer）
    - 不属于CIDF规范，可采用很多种现有的传输机制来实现

CISL公共入侵标准语言

- A Common Intrusion Specification Language：CIDF最核心、最重要的内容
- 表示原始事件信息、分析结果和响应指令，从而建立了IDS之间信息共享的基础