目录

[第一章 概述 8](#_Toc169969580)

[网络安全定义 8](#_Toc169969581)

[信息安全的三个级别 8](#_Toc169969582)

[安全的几个要素 8](#_Toc169969583)

[安全威胁的来源 8](#_Toc169969584)

[安全威胁的类型 9](#_Toc169969585)

[安全的目标 9](#_Toc169969586)

[安全漏洞简介 9](#_Toc169969587)

[网络安全的根源 9](#_Toc169969588)

[1.信息系统自身安全的脆弱性。 9](#_Toc169969589)

[2.信息系统面临的安全威胁 9](#_Toc169969590)

[3.安全管理问题 10](#_Toc169969591)

[4.黑客攻击 10](#_Toc169969592)

[5.网络犯罪 10](#_Toc169969593)

[6.为什么会感染恶意代码 10](#_Toc169969594)

[硬件与软件漏洞的例子 10](#_Toc169969595)

[软件： 10](#_Toc169969596)

[硬件： 11](#_Toc169969597)

[管理安全等级划分 11](#_Toc169969598)

[信息保障（PDRR） 11](#_Toc169969599)

[信息保障常用措施 11](#_Toc169969600)

[信息保障体系 12](#_Toc169969601)

[信息系统安全管理准则 12](#_Toc169969602)

[信息安全管理的地位 12](#_Toc169969603)

[信息安全管理的层次与内容 12](#_Toc169969604)

[信息安全管理的发展 12](#_Toc169969605)

[个人用户防护措施 12](#_Toc169969606)

[信息安全体系 13](#_Toc169969607)

[三个目标 13](#_Toc169969608)

[十个步骤 13](#_Toc169969609)

[网络安全策略 14](#_Toc169969610)

[制定安全策略的目的 14](#_Toc169969611)

[安全策略的必要性 14](#_Toc169969612)

[PPDR与PDRR区别 14](#_Toc169969613)

[网络安全策略的基本原则 14](#_Toc169969614)

[安全策略的特点 15](#_Toc169969615)

[网络安全技术 15](#_Toc169969616)

[第二章 网络攻击行径分析 16](#_Toc169969617)

[攻击事件分类 16](#_Toc169969618)

[破坏型攻击 16](#_Toc169969619)

[利用型攻击 17](#_Toc169969620)

[信息收集型攻击—p12 18](#_Toc169969621)

[网络欺骗攻击 18](#_Toc169969622)

[攻击步骤 19](#_Toc169969623)

[攻击目的 19](#_Toc169969624)

[第三章 网络侦查技术 20](#_Toc169969625)

[信息收集工具 20](#_Toc169969626)

[扫描器 20](#_Toc169969627)

[定义，功能 20](#_Toc169969628)

[重要性 20](#_Toc169969629)

[操作系统辨识 21](#_Toc169969630)

[操作系统辨识的动机 21](#_Toc169969631)

[如何辨识一个操作系统 21](#_Toc169969632)

[栈指纹技术 21](#_Toc169969633)

[扫描的类型 22](#_Toc169969634)

[地址扫描 22](#_Toc169969635)

[Ping: Packet InterNet Groper 22](#_Toc169969636)

[Traceroute 22](#_Toc169969637)

[端口扫描 23](#_Toc169969638)

[漏洞扫描 24](#_Toc169969639)

[ICMP协议 24](#_Toc169969640)

[用途 24](#_Toc169969641)

[特点 25](#_Toc169969642)

[数据包 25](#_Toc169969643)

[消息类型 25](#_Toc169969644)

[TCP协议 26](#_Toc169969645)

[网络监听 26](#_Toc169969646)

[防范方法 26](#_Toc169969647)

[检测手段 26](#_Toc169969648)

[以太网络的工作原理 28](#_Toc169969649)

[以太网卡的工作模式 28](#_Toc169969650)

[嗅探器 28](#_Toc169969651)

[交换式以太网 28](#_Toc169969652)

[共享式以太网 29](#_Toc169969653)

[应用程序抓包的技术 29](#_Toc169969654)

[Packet socket 29](#_Toc169969655)

[libpcap 29](#_Toc169969656)

[WinPcap 30](#_Toc169969657)

[口令破解 31](#_Toc169969658)

[第四章 拒绝服务攻击 32](#_Toc169969659)

[资源消耗破坏 32](#_Toc169969660)

[电子邮件轰炸 33](#_Toc169969661)

[分布式拒绝服务攻击DDoS 33](#_Toc169969662)

[DDoS和DoS区别 33](#_Toc169969663)

[DDoS特点 33](#_Toc169969664)

[被DDoS攻击时的现象 33](#_Toc169969665)

[DDoS工具 33](#_Toc169969666)

[DDoS攻击过程 34](#_Toc169969667)

[DDoS应对方式 34](#_Toc169969668)

[DDoS常用方式 34](#_Toc169969669)

[反弹技术 35](#_Toc169969670)

[攻击步骤 35](#_Toc169969671)

[反弹技术与传统DDoS区别 35](#_Toc169969672)

[第五章 缓冲区溢出攻击 36](#_Toc169969673)

[缓冲区定义 36](#_Toc169969674)

[缓冲区溢出的定义 36](#_Toc169969675)

[造成缓冲区溢出的根本原因 36](#_Toc169969676)

[缓冲区溢出的危害 36](#_Toc169969677)

[缓冲区溢出的目标 36](#_Toc169969678)

[缓冲区溢出工作原理 37](#_Toc169969679)

[进程在内存中的布局 37](#_Toc169969680)

[缓冲区溢出可能发生的位置 37](#_Toc169969681)

[常见的溢出缓冲区的途径 37](#_Toc169969682)

[防御技术 38](#_Toc169969683)

[黄金规则 38](#_Toc169969684)

[采取防御措施 38](#_Toc169969685)

[防御技术 38](#_Toc169969686)

[哪些编程习惯会导致缓冲区溢出 39](#_Toc169969687)

[Payload 40](#_Toc169969688)

[构造流程 40](#_Toc169969689)

[Unix&Windows 40](#_Toc169969690)

[构造payload注意事项 40](#_Toc169969691)

[如何在内存中找到一个jmp esp指令(二进制码为0xffe4)？ 41](#_Toc169969692)

[如果溢出函数的返回指令有偏移(比如ret 8)，则esp往后移动了 41](#_Toc169969693)

[Windows下编写shellcode 41](#_Toc169969694)

[如何调用系统函数 41](#_Toc169969695)

[消除代码中的null字节 41](#_Toc169969696)

[第六章 程序攻击 42](#_Toc169969697)

[逻辑炸弹攻击 42](#_Toc169969698)

[后门 42](#_Toc169969699)

[后门攻击方法 42](#_Toc169969700)

[可用作后门的系统漏洞 42](#_Toc169969701)

[后门和木马的区别 42](#_Toc169969702)

[后门的隐藏方式 43](#_Toc169969703)

[远程监控技术 43](#_Toc169969704)

[特洛伊木马 44](#_Toc169969705)

[木马的种类 44](#_Toc169969706)

[木马的欺骗手段 44](#_Toc169969707)

[防御木马、后门 44](#_Toc169969708)

[蠕虫 45](#_Toc169969709)

[病毒、蠕虫和特洛伊木马区别 45](#_Toc169969710)

[第七章 欺骗攻击 46](#_Toc169969711)

[DNS欺骗攻击 46](#_Toc169969712)

[DNS欺骗的方法 46](#_Toc169969713)

[DNS欺骗检测和防范思路 46](#_Toc169969714)

[检测方法 46](#_Toc169969715)

[防范思路 46](#_Toc169969716)

[防护方法： 46](#_Toc169969717)

[域名服务器安全DNSSEC 47](#_Toc169969718)

[Email欺骗攻击 47](#_Toc169969719)

[应对方式 47](#_Toc169969720)

[Web欺骗攻击 47](#_Toc169969721)

[欺骗的根源 47](#_Toc169969722)

[欺骗的动机 47](#_Toc169969723)

[欺骗的形式 47](#_Toc169969724)

[防止Web欺骗 48](#_Toc169969725)

[IP欺骗攻击 48](#_Toc169969726)

[动机 49](#_Toc169969727)

[IP欺骗的形式 49](#_Toc169969728)

[IP欺骗成功的要诀 49](#_Toc169969729)

[程序实现IP欺骗 49](#_Toc169969730)

[避免IP欺骗 49](#_Toc169969731)

[TCP会话劫持 50](#_Toc169969732)

[会话 50](#_Toc169969733)

[会话劫持（Session Hijack） 50](#_Toc169969734)

[欺骗和劫持 50](#_Toc169969735)

[动机 50](#_Toc169969736)

[原理 50](#_Toc169969737)

[一般性过程 50](#_Toc169969738)

[会话劫持详细过程 51](#_Toc169969739)

[TCP 会话劫持方法与形式 51](#_Toc169969740)

[不在一个子网中的劫持(欺骗)手法 52](#_Toc169969741)

[TCP ACK风暴（ACK Storm） 52](#_Toc169969742)

[如何到达不同步的状态 52](#_Toc169969743)

[防止会话劫持 52](#_Toc169969744)

[会话劫持工具 53](#_Toc169969745)

[HTTP会话劫持 53](#_Toc169969746)

[一般HTTPS通信过程 53](#_Toc169969747)

[以访问Gmail为例的基本过程如下： 53](#_Toc169969748)

[SSLstrip工作原理： 54](#_Toc169969749)

[采取哪些主要措施来保障WEB服务器安全 54](#_Toc169969750)

[如何理解信息安全领域"三分技术，七分管理”这名话的 54](#_Toc169969751)

[第八章 利用程序错误攻击 55](#_Toc169969752)

[windows 55](#_Toc169969753)

[windows漏洞 55](#_Toc169969754)

[漏洞防护 55](#_Toc169969755)

[ASP (Active Server Page) 55](#_Toc169969756)

[Unix 55](#_Toc169969757)

[Unix漏洞 55](#_Toc169969758)

[黑客的攻击按如下六个步骤进行： 55](#_Toc169969759)

[Web入侵 56](#_Toc169969760)

[Web安全问题的来源 56](#_Toc169969761)

[CGI 56](#_Toc169969762)

[跳板的作用是什么？ 56](#_Toc169969763)

[如何避免多个服务系统之间的连带关系？ 56](#_Toc169969764)

[简述用ASP编写的网站的常见攻击方式有哪些？ 57](#_Toc169969765)

[攻陷一个Windows server + IIS + ASP的网站 57](#_Toc169969766)

[攻陷一个UNIX + CGI + Perl的网站 57](#_Toc169969767)

[第九章 访问控制技术 58](#_Toc169969768)

[主体、客体与访问控制关系 58](#_Toc169969769)

[访问控制的有效性 58](#_Toc169969770)

[访问控制模式 58](#_Toc169969771)

[访问控制功能 58](#_Toc169969772)

[访问控制的类型 58](#_Toc169969773)

[自主访问控制 DAC 58](#_Toc169969774)

[定义 58](#_Toc169969775)

[实现方法 58](#_Toc169969776)

[自主访问控制优缺点 59](#_Toc169969777)

[自主访问控制的类型（控制许可权手段） 59](#_Toc169969778)

[层次型（等级型） 59](#_Toc169969779)

[属主型 60](#_Toc169969780)

[自由型 60](#_Toc169969781)

[强制访问控制 MAC 60](#_Toc169969782)

[实现方法 60](#_Toc169969783)

[支持MAC的措施 61](#_Toc169969784)

[实现MAC的实例 61](#_Toc169969785)

[新型访问控制技术 62](#_Toc169969786)

[木马窃取敏感文件的方法 62](#_Toc169969787)

[DAC和MAC的区别以及结合？ 62](#_Toc169969788)

[身份认证的选择标准 62](#_Toc169969789)

[面向过程的访问控制 63](#_Toc169969790)

[实现DAC的实例 63](#_Toc169969791)

[口令 63](#_Toc169969792)

[口令选择原则 63](#_Toc169969793)

[口令等级 64](#_Toc169969794)

[认证方式 64](#_Toc169969795)

[口令机制缺陷 64](#_Toc169969796)

[一次性口令（OTP One Time Password） 64](#_Toc169969797)

[一次性口令的生成方式 64](#_Toc169969798)

[不确定因子选择方式 64](#_Toc169969799)

[物理隔离 64](#_Toc169969800)

[物理隔离优点 65](#_Toc169969801)

[物理隔离的缺点 65](#_Toc169969802)

[逻辑隔离 65](#_Toc169969803)

[服务器端物理隔离 65](#_Toc169969804)

[隔离网闸 65](#_Toc169969805)

[网闸工作逻辑图 66](#_Toc169969806)

[网闸能够交换的数据类型 66](#_Toc169969807)

[网闸应用例子 66](#_Toc169969808)

[网络隔离卡和网卡的区别 67](#_Toc169969809)

[网闸与防火墙对比 67](#_Toc169969810)

[入网认证 67](#_Toc169969811)

[S/Key一次性口令认证系统 67](#_Toc169969812)

[Kerberos相关 68](#_Toc169969813)

[第十章 防火墙 69](#_Toc169969814)

[防火墙的基本原理 69](#_Toc169969815)

[防火墙分类 69](#_Toc169969816)

[体系结构 69](#_Toc169969817)

[构建 69](#_Toc169969818)

[第十一章 入侵检测 70](#_Toc169969819)

[概述 70](#_Toc169969820)

[分类和评估 70](#_Toc169969821)

[第十二章 VPN 71](#_Toc169969822)

[概述 71](#_Toc169969823)

[协议与实现 71](#_Toc169969824)

[身份验证 71](#_Toc169969825)

[隧道协议 71](#_Toc169969826)

[数据加密 71](#_Toc169969827)

[盲猜一手 72](#_Toc169969828)

[简答题 72](#_Toc169969829)

[攻击步骤 72](#_Toc169969830)

[交换式以太网和共享式以太网 72](#_Toc169969831)

[DoS 72](#_Toc169969832)

[缓冲区溢出 72](#_Toc169969833)

[病毒 蠕虫 木马的区别 72](#_Toc169969834)

[访问控制 72](#_Toc169969835)

[VPN 72](#_Toc169969836)

[综合题 72](#_Toc169969837)

[DDoS实例 72](#_Toc169969838)

[缓冲区溢出 72](#_Toc169969839)

[欺骗攻击 72](#_Toc169969840)

[防火墙 72](#_Toc169969841)

[\*入侵检测 72](#_Toc169969842)

[补充 74](#_Toc169969843)

# 第一章 概述

## 网络安全定义

网络安全指信息系统的硬件、软件及其系统中的数据受到保护，不会遭到偶然的或者恶意的破坏、更改、从而系统能连续、可靠、正常地运行，服务不中断。

网络安全简单的说是在网络环境下能够识别、消除不安全因素的能力。

## 信息安全的三个级别

按照范围和处理方式的不同，通常将信息安全划分为三个级别：

第1级为计算机安全（基础）。设备安全、操作系统安全、数据库安全等。设计漏洞

第2级为网络安全（核心）。处理、存储、传输等。协议漏洞

第3级为信息系统安全（目标）。用户透明

## 安全的几个要素

可用性：授权实体有权访问数据。

机密性：信息不暴露给未授权实体或进程。

完整性：保证数据不被未授权修改。

可控性：控制授权范围内的信息流向及操作方式。

可审查性：对出现的安全问题提供依据与手段。

## 安全威胁的来源

外部渗入：未被授权使用计算机的人。

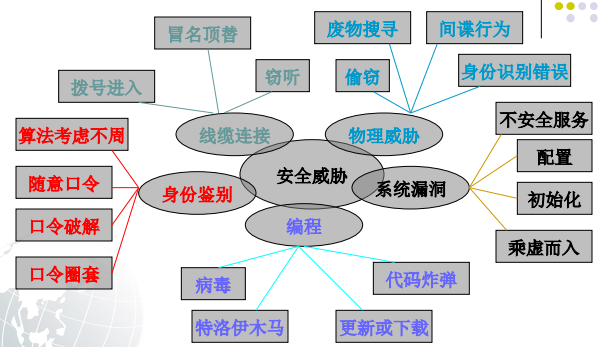
内部渗入者：被授权使用计算机，但不能访问某些数据、程序或资源，它包括：

➢ 冒名顶替: 使用别人的用户名和口令进行操作；

➢ 隐蔽用户: 逃避审计和访问控制的用户；

滥用职权者：被授权使用计算机和访问系统资源，但滥用职权者。

## 安全威胁的类型



## 安全的目标

课本p4

## 安全漏洞简介

漏洞也叫脆弱性（Vulnerability），是计算机系统在硬件、软件、协议的具体实现或系统安全策略上存在的缺陷和不足。

漏洞一旦被发现，就可使用这个漏洞获得计算机系统的额外权限，使攻击者能够在未授权的情况下访问或破坏系统，从而导致危害计算机系统安全。

## 网络安全的根源

### 1.信息系统自身安全的脆弱性。

指信息系统的硬件资源、通信资源、软件及信息资源等，因可预见或不可预见、甚至恶意的原因，而可能导致系统受到破坏、更改、泄露和功能失效，从而使系统处于异常状态，甚至崩溃瘫痪等的根源和起因。

脆弱性与漏洞存在的原因为：支持Internet运行的TCP/IP协议栈原本只考虑互联互通和资源共享问题，并未考虑也无法兼顾解决来自网际的大量安全问题基于TCP/IP的Internet是在可信任网络环境中开发出来的成果，体现在TCP/IP协议上的总体构想和设计本身，基本未考虑安全问题，并不提供人们所需的安全性和保密性。TCP/IP协议最初设计的应用环境是互相信任的

### 2.信息系统面临的安全威胁

**基本威胁**

（1）信息泄漏。信息泄漏指敏感数据在有意或无意被泄漏、丢失或透露给某个未授权的实体。信息泄漏包括：信息在传输中被丢失或泄漏；通过信息流向、流量、通信频度和长度等参数等分析，推测出有用信息。

（2）完整性破坏。以非法手段取得对信息的管理权，通过未授权的创建、修改、删除等操作而使数据的完整性受到破坏。

（3）拒绝服务。信息或信息系统资源等服务能力下降或丧失。

产生服务拒绝的原因：

➢ 受到攻击所致，攻击者通过对系统进行非法的、根本无法成功的访问尝试而产生过量的系统负载，从而导致系统的资源对合法用户的服务能力下降或丧失。

➢ 信息系统或组件在物理上或逻辑上受到破坏而中断服务

（4）未授权访问。未授权实体非法访问信息系统资源

➢ 非法访问：假冒和盗用合法用户身份攻击、非法进入网络系统进行违法操作或授权实体超越权限访问信息系统资源。

➢ 越权访问：合法用户以未授权的方式进行操作等形式。

**威胁信息系统的主要方法**

特洛伊木马；黑客攻击；计算机病毒；信息丢失、篡改、销毁；后门、隐蔽通道；拒绝服务攻击；逻辑炸弹；蠕虫；内部、外部泄密

### 3.安全管理问题

管理策略不够完善，管理人员素质低下，用户安全意识淡薄，有关的法律规定不够健全。

管理上权责不分，对计算机安全不重视，少数管理权过大，而实际工作中又不需要。

管理不严格，缺乏系统的整体培训。

没有保密意识，系统密码随意传播，导致出现问题时相互推卸责任的局面

### 4.黑客攻击

网络黑客的主要攻击手法有：获取口令、放置木马程序、网页的欺骗技术、电子邮件攻击、通过一个节点攻击另一节点、网络监听、寻找系统漏洞、利用帐号进行攻击、窃取特权

### 5.网络犯罪

网络犯罪的类型：网络文化污染、盗版交易、网络欺诈、名誉毁损、侵入他人主页、网站、邮箱、制造传播计算机病毒、网络赌博、网络教唆、煽动各种犯罪

### 6.为什么会感染恶意代码

浏览网页。使用即时通讯工具、浏览邮件、下载文件、远程攻击、局域网攻击、使用移动存储介质

## 硬件与软件漏洞的例子

### 软件：

Log4Shell实现数据盗窃、间谍活动和恶意软件的传播。

EternalBlue可以访问一系列基于Windows的设备，包括运行Windows 7、Windows 8和经常被恶意中伤的Windows Vista的设备

Heartbleed从以前安全的计算机中挤出敏感数据。从本质上讲，攻击者会向系统发送大量请求，希望获得一些有价值的信息。

Double Kill，0day漏洞，使Windows系统面临风险，利用VB脚本缺陷。攻击方法涉及使用恶意 Internet Explorer 网页，其中包含滥用漏洞所需的代码。

BlueKeep漏洞，可用于在目标设备上远程执行代码。Microsoft开发的概念证明表明，目标计算机可能会在一分钟内被攻击者入侵并接管，这凸显了漏洞的严重性。

### 硬件：

（1）CVE-2017-5753 又名边界检查绕过漏洞，可供攻击者利用现代 CPU 的分支预测功能，以 CPU 高速缓存为边信道，从其他进程的内存中抽取信息。利用该漏洞，进程可从另一进程的内存中抽取敏感信息，还能绕过用户/内核内存权限界限。英特尔、IBM 和一些 ARM CPU 均受此漏洞影响。

（2）幽灵变种 2 - CVE-2017-5715

幽灵变种 2 与变种 1 效果类似，但采用了不用的漏洞利用技术——分支目标注入。更新受影响 CPU 微代码可有效缓解该幽灵漏洞变种，通过 BIOS/UEFI 更新或在每次重启时由操作系统更新均可。

（3）熔断变种 3 - CVE-2017-5754

亦称流氓数据缓存加载 (RDCL) 或 CPU 预测执行漏洞变种 3，利用的是现代英特尔 CPU 的乱序执行功能。该漏洞可使用户进程跨越安全边界读取受保护的内核内存。只需操作系统更新和实现更严格的内核内存隔离即可修复此漏洞。内核内存中通常存有敏感秘密，严格隔离可通过 Linux 内核页表隔离 (KPTI) 等机制实现。

（4）熔断-GP - CVE-2018-3640

属于熔断漏洞变种，亦称变种 3a，利用系统寄存器预测读取功能实现信息边信道泄露。因此，该漏洞变种还有个名字叫流氓系统寄存器读 (RSRE)。缓解该漏洞需进行微代码更新。

（5）熔断-NM - CVE-2018-3665

与熔断漏洞相关的预测执行漏洞，又名 LazyFP，可用于泄露浮点运算单元 (FPU) 状态。FPU 是英特尔现代 CPU 中的专用数学协处理器，用于加速浮点数算术运算的。FPU 状态可能含有密码运算敏感信息。在操作系统层级将 FPU 上下文切换模式由“lazy”改为“eager”即可缓解该漏洞。

## 管理安全等级划分

第一级：用户自主保护级。实施计划管理

第二级：系统审计保护级。实施操作规程管理

第三级：安全标记保护级。实施标准化过程管理

第四级：结构化保护级。实施安全生态管理

第五级：访问验证保护级。实施安全文化管理

## 信息保障（PDRR）

保护（Protect）

采用的手段保障信息的保密性、完整性、可用性、可控性和不可否认性。

检测（Detect）

利用高级术提供的工具检查系统存在的可能提供黑客攻击、白领犯罪、病毒泛滥脆弱性。

反应（React）

对危及安全的事件、行为、过程及时作出响应处理，杜绝危害的进一步蔓延扩大，力求系统尚能提供正常服务。

恢复（Restore）

一旦系统遭到破坏，尽快恢复系统功能，尽早提供正常的服务

## 信息保障常用措施

完善安全管理制度、采用访问控制措施、运行数据加密措施、数据备份与恢复、建立敏感的安全意识

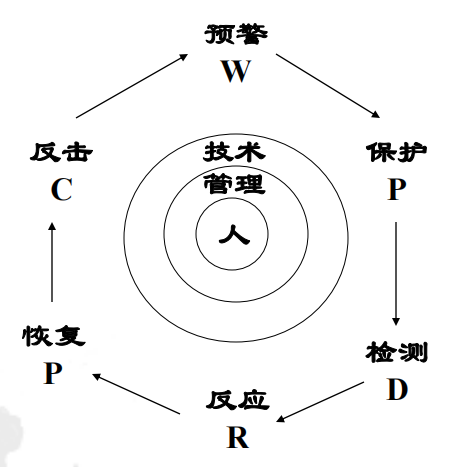
## 信息保障体系

法律与政策体系、标准与规范体系、人才培养体系、产业支撑体系、技术保障体系、组织管理体系

## 信息系统安全管理准则

管理策略、组织与人员、资产分类与安全控制、配置与运行、网络信息安全与通信安全、异常事件与审计、信息标记与文档、物理环境、开发与维护、作业连续性保障、符合性

## 信息安全管理的地位



## 信息安全管理的层次与内容

宏观管理（政府）

方针、政策、法规、标准

微观管理（机构）

规章、制度、策略、措施

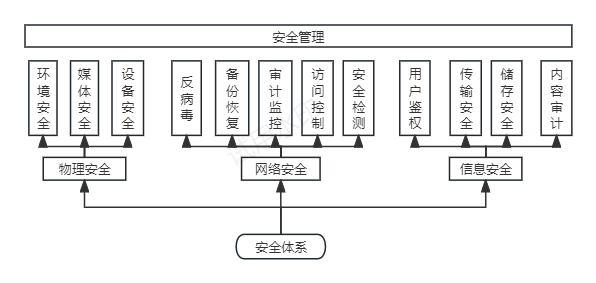
## 信息安全管理的发展

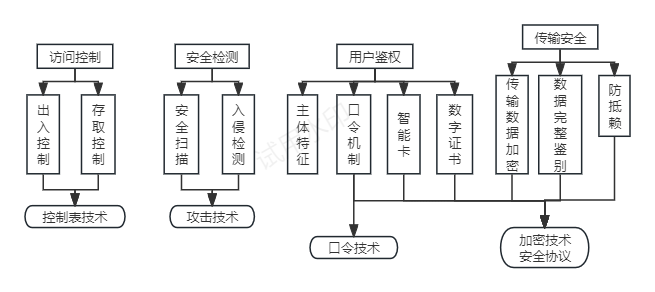
管人-管密码-管密钥-管口令-管配置-管产品测评-管产品采购-管系统安全-管等级划分

## 个人用户防护措施

防火墙、加密重要文件、杀毒软件定期升级和杀毒、定期备份系统或重要文件、定期升级补丁、关闭不常用端口、关闭不常用程序和服务、发现系统异常立刻检查

## 信息安全体系





## 三个目标

准备和防范

检测和响应

建立牢固的根基

## 十个步骤

准备和防范

步骤1：确认关键基础设施资产以及相互依赖性，发现其脆弱性

检测与响应

步骤2：检测攻击和非法入侵

步骤3：开发稳健的情报和执法功能，保持法律的一致

步骤4：以实时的方式共享攻击警告和信息

步骤5：建立响应、重建和回复能力

建立牢固的根基

步骤6：为支持程序1－5，加强研究和开发

步骤7：培训和雇用足够数量的安全专家

步骤8：进行拓展，使公知晓提高计算机安全的必要性

步骤9：通过立法和拨款，支持程序1－8

步骤10：在计划的每一步骤和部分中，要完全保护公民的自由权、隐私权以及私有数据

## 网络安全策略

安全策略，是针对那些被允许进入某一组织、可以访问网络技术资源和信息资源的人所规定的、必须遵守的规则。

即：网络管理部门根据整个计算机网络所提供的服务内容、网络运行状况、计算机安全状况、安全性需求、易用性、技术实现所付出的代价和风险、社会因素等许多方面因素，所制定的关于计算机安全总体目标、计算机安全操作、计算机安全工具、人事管理等方面的规定。

## 制定安全策略的目的

目的1－让所有用户、操作人员和管理员清楚，为了保护技术和信息资源所必须遵守的原则。

目的2－提供一个可以获得、能够配置和检查的用于确定是否与计算机和网络系统的策略一致的基准。

## 安全策略的必要性

网络管理员在作安全策略时的依据，在很大程度上取决于网络运行过程中的安全状况，网络所提供的功能以及网络的易用程度。

安全策略应以要实现目标为基础，而不能简单地规定要检验什么和施加什么限制。

在确定的安全目标下，应该制定如何有效地利用所有安全工具的策略。

P2DR模型，强调了策略的核心作用，强调了检测、响应、防护的动态性，检测、响应、防护必须遵循安全策略进行

## PPDR与PDRR区别

PDRR安全模型强调网络保护不再是简单的被动保护，而是保护、检测、响应和恢复的有机结合。因此，PDRR模型不仅包含了安全防护的概念，而且还包含了主动恢复概念。

在PDRR安全模型中检测显得非常重要的一步。检测的目的是检测网络攻击，检测本地网络中的非法信息流，检测本地网络中的安全漏洞，有效防范网络攻击。通信部分检测技术包括入侵检测技术和网络安全扫描技术。

PPDR中，所谓管理和策略的部分，仅仅列上Policy，显然不全面。比如，策略难于包含组织、运作等管理内涵。

## 网络安全策略的基本原则

**适用性原则**

网络的安全管理是一个系统化的工作，因此在制定安全管理策略时，应全面考虑网络上各类用户，各种设备，各种情况，有计划有准备地采取相应的策略，任何一点疏忽都会造成整个计算机安全性的降低。

**可行性原则**

安全管理策略的制定还要考虑资金的投入量，因为安全产品的性能一般是与其价格成正比的，所以要适合划分系统中信息的安全级别，并作为选择安全产品的重要依据，使制定的安全管理策略达到成本和效益的平衡

**动态性原则**

安全管理策略有一定的时限性，不能是一成不变的。由于网络用户在不断地变化，网络规模在不断扩大，网络技术本身的发展变化也很快，而安全措施是防范性的。所以安全措施也必须随着网络发展和环境的变化而变化。

**简单性原则**

网络用户越多，网络管理人员越多，网络拓扑越复杂，采用网络设备种类和软件种类越多，网络提供的服务和捆绑越多，出现安全漏洞的可能性就越大。因此制定的安全管理策略越简单越好，如简化授权用户的注册过程等

**系统性原则**

网络的安全管理是一个系统化的工作，因此在制定安全管理策略时，应全面考虑网络上各类用户，各种设备，各种情况，有计划有准备地采取相应的策略。任何一点疏忽都会造成整个计算机安全性的降低。

## 安全策略的特点

有效的安全策略都是起码应该具有以下特点：

**发布**：必须通过系统正常管理程序，采用合适的标准出版物或其他适当的方式来发布。

**强制执行**：在适当的情况下，必须能够通过安全工具来实现其强制实施，并在技术确定不能满足要求的情况下强迫执行。

**人员责任规定**：必须明确规定用户、系统管理员和公司管理人员等各类人员的职责范围和权限

## 网络安全技术

安全内核技术

安全等级制

身份鉴别技术

Kerberos

Web安全技术

SSL

SHTTP

SOCKS协议

网络反病毒技术

防火墙技术

动态IP过滤技术

IP分片过滤技术

IP欺骗保护

地址转换

访问控制

保密网关技术

面向信息与面向客户

综合安全与保密策略实现

# 第二章 网络攻击行径分析

## 攻击事件分类

破坏型攻击

利用型攻击

信息收集型攻击

网络欺骗攻击

垃圾信息攻击

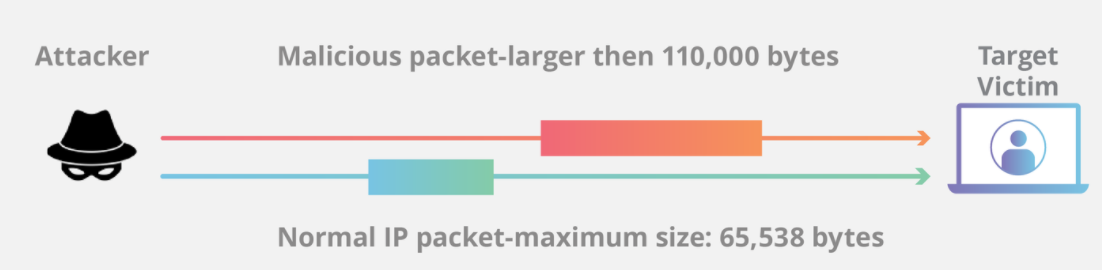
## 破坏型攻击

P9 定义，常见类型（病毒攻击，Dos）

**Dos常见类型和手段**

（1）Ping of Death

PPT+书上都有原理 *ping -s (设置数据包大小) IP\_Addr*



**防御方法**：对重新组装过程添加检查，以确保在分组重组后不会超过最大数据包大小约束;创建一个具有足够空间的内存缓冲区来处理超过最大准则的数据包。

（2）IGMP/ICMP Flood

**防御方法**：被攻击目标可以在其网络边界直接过滤并丢弃ICMP/IGMP数据包使攻击无效化。

（3）Teardrop

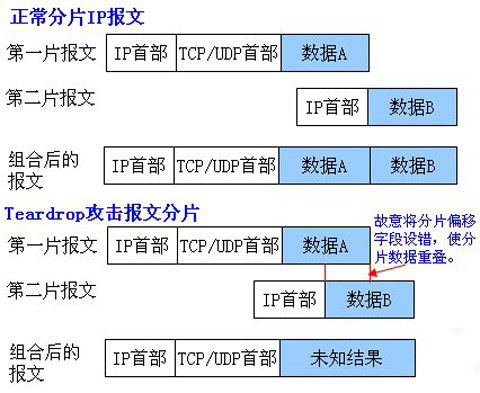
是一种畸形报文攻击。

是基于UDP的病态分片数据包的攻击方法

其工作原理是向被攻击者发送多个分片的IP包（IP分片数据包中包括该分片数据包属于哪个数据包以及在数据包中的位置等信息）

某些操作系统收到含有重叠偏移的伪造分片数据包时将会出现系统崩溃、重启等现象

**原理**：第一个包的偏移量为0，长度为N，第二个包的偏移量小于N。为了合并这些数据段，TCP/IP堆栈会分配超乎寻常的巨大资源，从而造成系统资源的缺乏，甚至机器重新启动。



**防御方法：**

1. 网络安全设备将接收到的分片报文先放入缓存中，并根据源IP地址和目的IP地址对报文进行分组，源IP地址和目的IP地址均相同的报文归入同一组，然后对每组IP报文的相关分片信息进行检查，丢弃分片信息存在错误的报文。

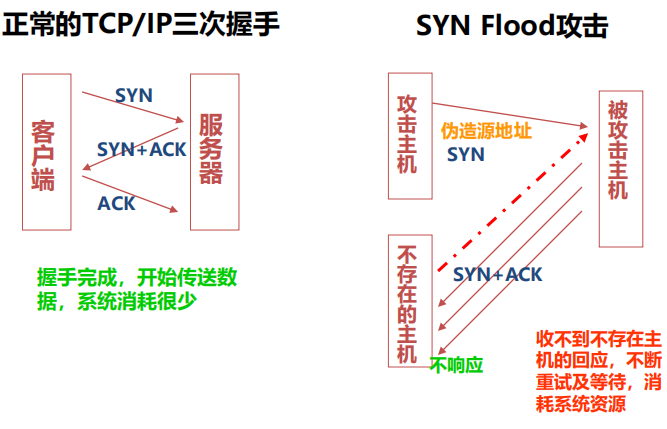
2.为了防止缓存溢出，当缓存快要存满时，直接丢弃后续分片报文。

（4）UDP Flood

攻击端口为业务端口：根据该业务UDP最大包长设置UDP最大包大小以过滤异常流量。

攻击端口为非业务端口：丢弃所有UDP包；建立UDP连接规则，要求所有去往该端口的UDP包，必须首先与TCP端口建立TCP连接。这种方法需要专业的防火墙或其他防护设备支持。

1. SYN Flood



（6）ICMP Flood

攻击者向一个子网的广播地址发送多个ICMP Echo请求数据包。并将源地址伪装成想要攻击的目标主机的地址。

然后该子网上的所有主机均会对此ICMP Echo请求包作出答复，向被攻击的目标主机发送数据包，使此主机受到攻击，导致网络阻塞。

**ICMP实质**是控制信息协议，攻击者利用ICMP获取主机信息，时间、路由信息等。

## 利用型攻击

P11

（1）口令猜测

首先获得账号（使用Finger命令查询时会保存用户名；社工-email等），然后破译口令。

**网络监听**：很多协议没有采用任何加密或身份认证技术（Telnet、FTP、HTTP、SMTP等传），用户帐户和密码信息都是以明文格式传输的，此时若攻击者利用数据包截取工具便可很容易收集到帐户和密码。

**缓冲区溢出**：编制有缓冲区溢出错误的SUID程序来获得超级用户权限。

（2）特洛伊木马

基于C/S结构的远程控制程序，是一类隐藏在合法程序中的恶意代码，这些代码或者执行恶意行为，或者为非授权访问系统的特权功能而提供后门。

**传染方式**:通过电子邮件附件发出，捆绑在其他的程序中。

**病毒特性**:会修改注册表、驻留内存、在系统中安装后门程序、开机加载附带的木马。

**防范措施**:不下载和运行来历不明的程序，不打开不明来历的邮件附件。

（3）缓冲区溢出

## 信息收集型攻击—p12

**信息收集方法：**

1. 从一些社会信息入手：通过社工的方式，查找到目标的一些信息。可以参考CYPPT，上面有很多信息收集的内容。

2. 找到网络地址范围：比如通过社交软件获得目标所在地，获得大致ip范围。通过扫描技术，地址扫描

3. 找到关键的机器地址：通过扫描技术，地址扫描

4. 找到开放端口和入口点：端口扫描技术（以上都在课本p12）

5. 找到系统的制造商和版本：体系结构探测（课本p13）

**扫描技术:**

Nmap: P35

进行ping扫描，打印出对扫描做出响应的主机,不做进一步测试

nmap -sP 192.168.1.0/24

仅列出指定网络上的每台主机，不发送任何报文到目标主机：

nmap -sL 192.168.1.0/24

探测目标主机开放的端口，可以指定一个以逗号分隔的端口列表(如-PS22，23，25，80)：

nmap -PS 192.168.1.234

使用UDP ping探测主机：

nmap -PU 192.168.1.0/24

使用频率最高的扫描选项：SYN扫描,又称为半开放扫描，它不打开一个完全的TCP连接，执行得很快：

nmap -sS 192.168.1.0/24

当SYN扫描不能用时，TCP Connect()扫描就是默认的TCP扫描：

nmap -sT 192.168.1.0/24

UDP扫描用-sU选项,UDP扫描发送空的(没有数据)UDP报头到每个目标端口:

nmap -sU 192.168.1.0/24

确定目标机支持哪些IP协议 (TCP，ICMP，IGMP等):

nmap -sO 192.168.1.19

探测目标主机的操作系统：

nmap -O 192.168.1.19

nmap -A 192.168.1.19

* 地址扫描
* 端口扫描
* 反响映射
* 慢速扫描
* 漏洞扫描

## 网络欺骗攻击

P13 （扩展答）第七章 欺骗攻击p156

* DNS欺骗攻击
* 电子邮件欺骗
* Web欺骗
* IP欺骗

## 攻击步骤

P17

（1）准备

确定攻击目的

准备攻击工具

网络侦查工具: superscan ，Nmap

拒绝服务攻击工具:DDoS攻击者, sqldos , Trinoo

木马:BackOrifice2000 ，冰河 ，NetSpy

安全漏洞攻击:Outlook，IIS，Serv-U

对防火墙的攻击:Firewalking、Hping

收集目标信息 结合扫描技术

（2）实施

隐藏自己位置（使用他人IP）

登录主机 结合利用型攻击三点答（口令猜测/特洛伊木马/缓冲区溢出）

窃取网络资源和特权 （根据具体问题的攻击目的答）

（3）善后

禁止日志审计，清除事件日志，清除IIS服务日志。为了下次攻击的方便，攻击者都会留下一个后门，充当后门的工具种类非常多，最典型的是木马程序

**为什么要进行善后工作？--p20**

## 攻击目的

动机：恶作剧、恶意破坏、商业目的、政治军事、偷取国家机密、商业竞争行为、内部员工对单位的不满、对企业核心机密的企望、网络接入帐号、信用卡号等金钱利益的诱惑、利用攻击网络站点而出名、对网络安全技术的挑战、对网络的好奇心

性质：破坏、入侵

攻击目的：破坏目标工作、窃取目标信息、控制目标机器、利用假消息欺骗对方

# 第三章 网络侦查技术

## 信息收集工具

(1)Whois

为Internet提供目录服务，包括名字、通讯地址、电话号码、电子邮箱、IP地址等信息

(2)Client/Server结构

Client端

发出请求，接受结果，并按格式显示到客户屏幕上

Server端

1. 建立数据库，接受注册请求
2. 提供在线查询服务

(3)客户程序

UNIX系统自带whois程序

Windows也有一些工具

直接通过Web查询

(4)nslookup、

可以把DNS数据库中的信息挖掘出来

分两种运行模式

1. 非交互式：通过命令行提交命令
2. 交互式：可以访问DNS数据库中所有开放的信息

通过nslookup可以做什么？

1. 区域传送：可以列出DNS节点中所有的配置信息。这是为了主DNS和辅DNS之间同步复制才使用的
2. 查看一个域名，根据域名找到该域的域名服务器
3. 反向解析，根据IP地址得到域名名称

DNS & nslookup的注意点

1. 关闭未授权区域传送功能
2. 或者，在防火墙上禁止53号TCP端口，DNS查询请求使用53号UDP端口
3. 区分内部DNS和外部DNS
4. 内部信息不出现在外部DNS中
5. DNS中该公开的信息总是要公开的，否则，域名解析的功能就无效了，没有MX记录就不能支持邮件系统

## 扫描器

定义，功能P30

### 重要性

扫描器能够暴露网络上潜在的脆弱性

无论扫描器被管理员利用，或者被黑客利用，都有助于加强系统的安全性

➢ 它能使得漏洞被及早发现，而漏洞迟早会被发现的

扫描器除了能扫描端口，往往还能够

➢ 发现系统存活情况，以及哪些服务在运行

➢ 用已知的漏洞测试这些系统

➢ 对一批机器进行测试，简单的迭代过程

➢ 有进一步的功能，包括操作系统辨识、应用系统识别

扫描器可以满足很多人的好奇心

**常用的网络扫描器**

p35

最近的扫描器

地址端口扫描

➢ ScanPort，Ghost Port Scanner，nbtscan，Dr. Morena，IP Restrictions Scanner，Arp-scan，Cheops-ng

特征扫描

➢ Xprobe，LDistFP，TelnetFP

系统扫描

➢ Security Administrators Integrated Network Tool ，VLAD ，NetPing，HostScan

无线扫描

➢ NetStumbler, WirelessMon，WiFi Hopper，Vistumbler

其他

SATAN、strobe、Pinger、Portscan、Superscan

## 操作系统辨识

### 操作系统辨识的动机

➢ 许多漏洞是系统相关的，而且往往与相应的版本对应

➢ 从操作系统或者应用系统的具体实现中发掘出来的攻击手段都需要辨识系统

➢ 操作系统的信息还可以与其他信息结合起来，比如漏洞库，或者社会诈骗(社会工程，social engineering)

### 如何辨识一个操作系统

➢ 一些端口服务的提示信息，例如，telnet、http、ftp等服务的提示信息

➢ TCP/IP栈指纹

➢ DNS泄漏出OS系统

### 栈指纹技术

定义：利用TCP/IP协议栈实现上的特点来辨识一个操作系统

技术导向

1. 可辨识的OS的种类，包括哪些操作系统
2. 结论的精确度，细微的版本差异是否能识别

工具

Checkos, by Shok

Queso, by Savage

Nmap, by Fyodor

做法

寻找不同操作系统之间在处理网络数据包上的差异，并且把足够多的差异组合起来,以便精确地识别出一个系统的OS版本

配置能力

扩展性，新的OS，版本不断推出

定义一种配置语言或者格式

**栈指纹识别方法**

ICMP协议

ICMP错误消息的限制

发送一批UDP包给高端关闭的端口，然后计算返回来的不可达错误消息

ICMP端口不可达消息的大小

通常情况下送回IP头+8个字节，但是个别系统送回的数据更多一些

ICMP回应消息中对于校验和的处理方法不同

ICMP回应消息中，TOS域的值

TCP选项

这里充满了各种组合的可能性

应答方式“Query-Reply”，可以把多个选项放到一个包中

SYN Flooding对抗测试

先发送8个SYN包，看还能不能建立连接，确认它是否受此攻击

## 扫描的类型

### 地址扫描

### Ping: Packet InterNet Groper

用来判断远程设备可访问性最常用的方法

**原理**：发送ICMP Echo消息，然后等待ICMP Reply消息

可以确定网络和外部主机的状态，可以用来调试网络的软件和硬件，可以用来发现一台主机是否active

**为什么不能ping成功？**

➢ 没有路由，网关设置？

➢ 网卡没有配置正确

➢ Timeout值

➢ 防火墙阻止掉了

“Ping of death”

➢ 发送特大ping数据包(>65535字节)导致机器崩溃

缺点：针对单台主机，面对大量主机，效率低。需要借助一些现成工具。

### Traceroute

用来发现实际的路由路径

**原理**：给目标的一个无效端口发送一系列UDP，其TTL依次增一，中间路由器返回一个ICMP Time Exceeded消息通过向目标发送不同 IP 生存时间 (TTL) 值的“ (ICMP)”回应数据包，确定到目标所采取的路由。要求路径上的每个路由器在转发数据包之前至少将数据包上的 TTL 递减 1。数据包上的 TTL 减为 0 时，路由器应该将“ICMP 已超时”的消息发回源系统

Traceroute允许指定宽松的源路由选项。

### 端口扫描

基于TCP/IP协议，对各种网络服务，无论是主机或者防火墙、路由器都适用

端口扫描可以确认各种配置的正确性，避免遭受不必要的攻击

**用途**

➢ 管理员可以用来确保自己系统的安全性

➢ 黑客用来探查系统的入侵点

熟知端口 ( 0～1023 )

注册端口 ( 1024～49151 )

专用端口 ( 49152～65535 )

**扫描方法**：-p32

**TCP connect()扫描**

**做法**

扫描器调用socket的connect()函数发起一个正常的连接如果端口是打开的，则连接成功否则，连接失败

**优点**

简单，不需要特殊的权限

**缺点**

服务器可以记录下客户的连接行为，如果同一个客户轮流对每一个端口发起连接，则一定是在扫描

**TCP SYN扫描**

**做法**

向目标主机的特定端口发送一个SYN包

如果应答包为RST包，则说明该端口是关闭的

否则，会收到一个SYN|ACK包。于是，发送一个RST，停止建立连接

由于连接没有完全建立，所以称为“半开连接扫描”

**优点**

很少有系统会记录这样的行为，更隐蔽

**缺点**

在UNIX平台上，需要root权限才可以建立这样的SYN数据包

**TCP Fin扫描**

**做法**

扫描器发送一个FIN数据包

如果端口关闭的，则远程主机丢弃该包，并送回一个RST包

否则的话，远程主机丢弃该包，不回送

变种，组合其他的标记

**优点**

不是TCP建立连接的过程，所以比较隐蔽

**缺点**

与SYN扫描类似，也需要构造专门的数据包

在Windows平台无效，总是发送RST包

分片扫描

它本身并不是一种新的扫描方法，而是其他扫描技术的变种，特别是SYN扫描和FIN扫描思想是，把TCP包分成很小的分片，从而让它们能够通过包过滤防火墙

注意，有些防火墙会丢弃太小的包

而有些服务程序在处理这样的包的时候会出现异常，或者性能下降，或者出现错误

**Reverse-ident扫描**

Ident协议使得攻击者可以发现任何一个通过TCP连接的进程的所有者的用户名，即使该进程并没有发起该连接

只有在TCP全连接之后才有效

TCP端口113

例如

可以先连接到80端口，然后通过identd来发现服务器是否在root下运行

建议关闭ident服务，或者在防火墙上禁止，除非是为了审计的目的

**TCP ftp proxy扫描（FTP bounce attack）**

**做法**

在ftp协议中，数据连接可以与控制连接位于不同的机器上

让ftp server与目标主机建立连接，而且目标主机的端口可以指定

如果端口打开，则可以传输否则，返回"425 Can't build data connection: Connection refused."

Ftp这个缺陷还可以被用来向目标(邮件,新闻)传送匿名信息

**优点**：这种技术可以用来穿透防火墙

**缺点**：慢，且有些ftp server禁止这种特性

**UDP ICMP端口不可达扫描**

利用UDP协议

**做法**

开放的UDP端口并不需要送回ACK包，而关闭的端口也不要求送回错误包，所以利用UDP包进行扫描非常困难

有些协议栈实现的时候，对于关闭的UDP端口，会送回一个ICMP Port Unreach错误

**缺点**

速度慢，而且UDP包和ICMP包都不是可靠的

需要root权限，才能读取ICMP Port Unreach消息

一个应用例子

Solaris的rpcbind端口(UDP)位于32770之上，这时可以通过这种技术来探测

**UDP recvfrom() & write()扫描**

非root用户不能直接读取ICMP Port Unreach消息，但是Linux提供了一种方法

可以间接通知到

**做法**

第二次对一个关闭的UDP端口调用write()总是会失败

**经验**：在ICMP错误到达之前，在UDP端口上调用recvfrom()会返回EAGAIN(重试)，否则会返回ECONNREFUSED(连接拒绝)

### 漏洞扫描

## ICMP协议

### 用途

网关或者目标机器利用ICMP与源通讯，当出现问题时，提供反馈信息

用于报告错误

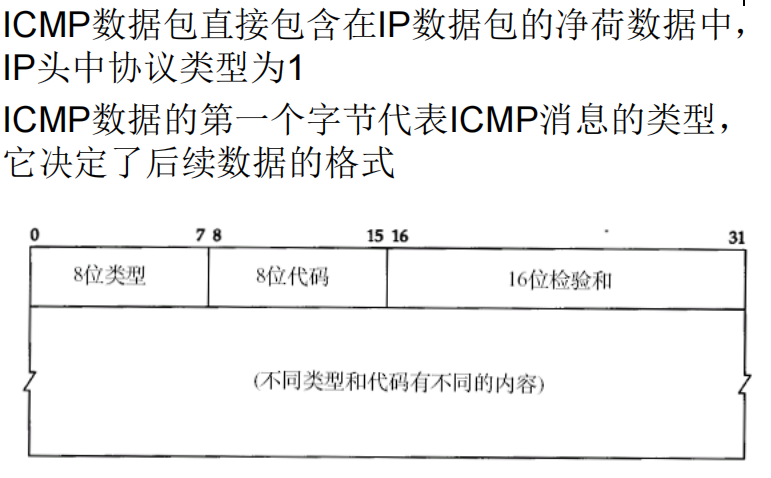
### 特点

其控制能力并不用于保证传输的可靠性

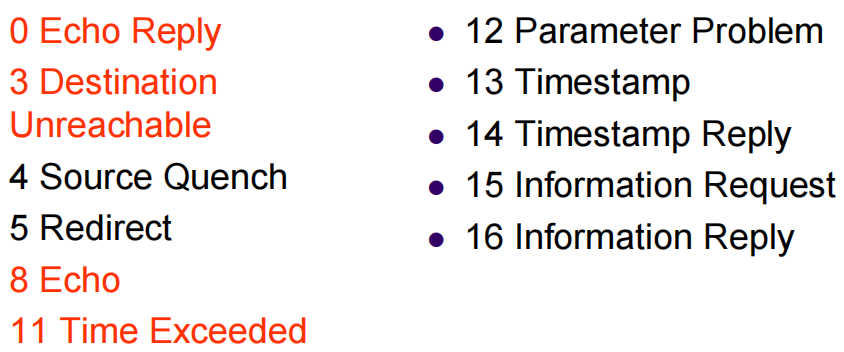
它本身也不是可靠传输的

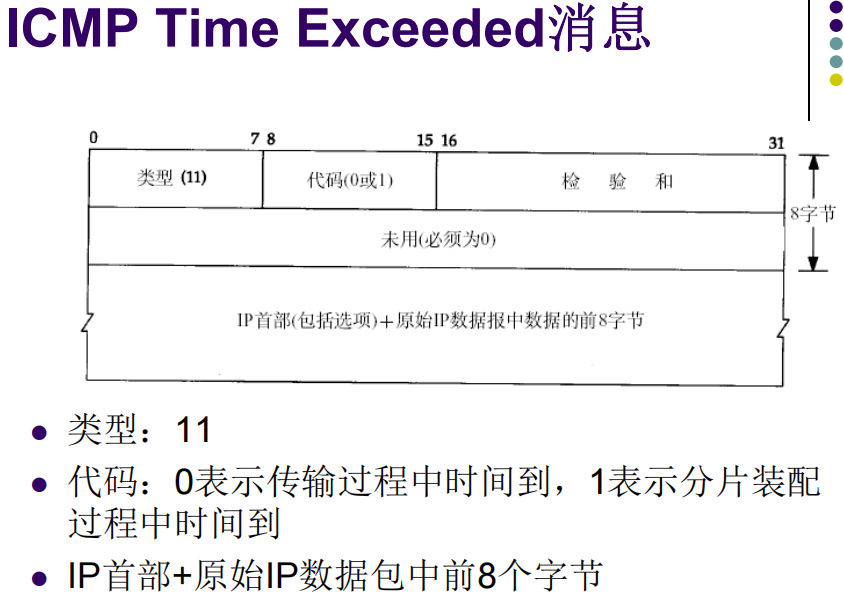
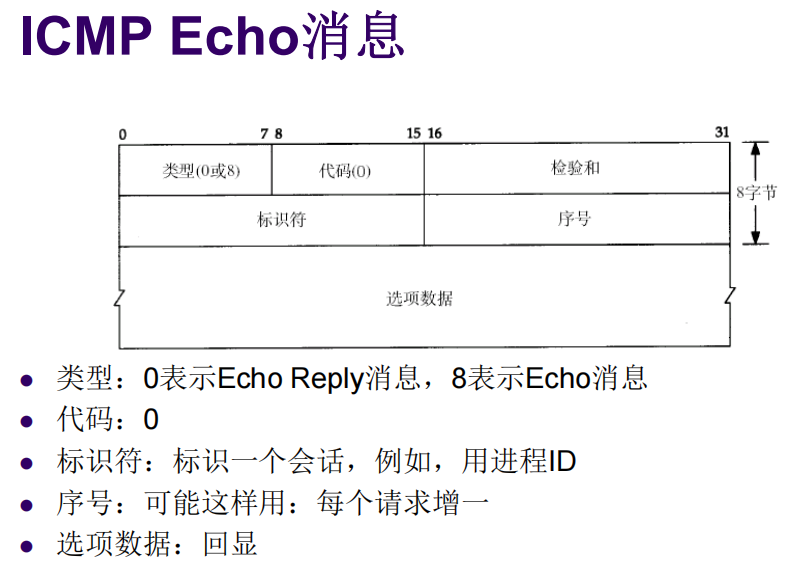
并不用来反映ICMP消息的传输情况

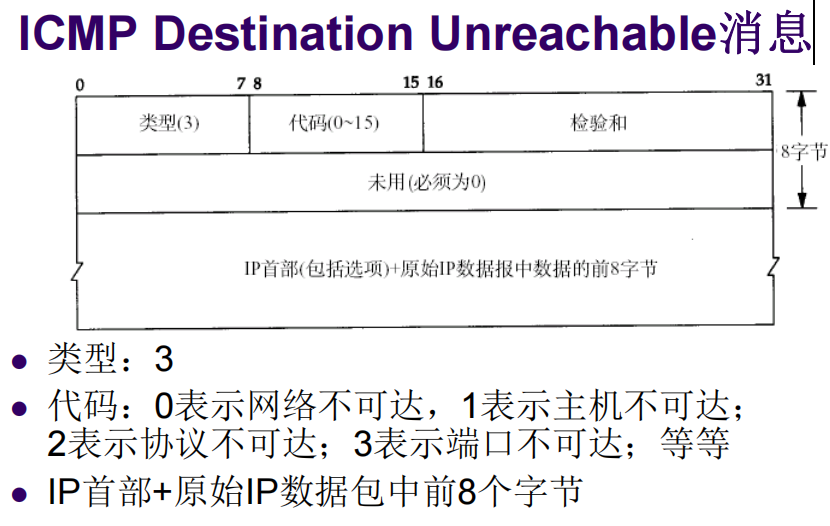
### 数据包



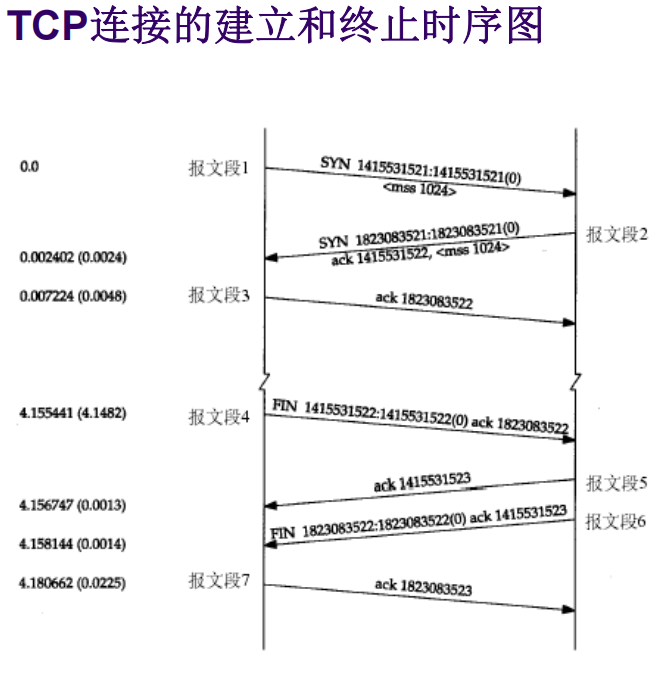
### 消息类型







## TCP协议



## 网络监听

p39

### 防范方法

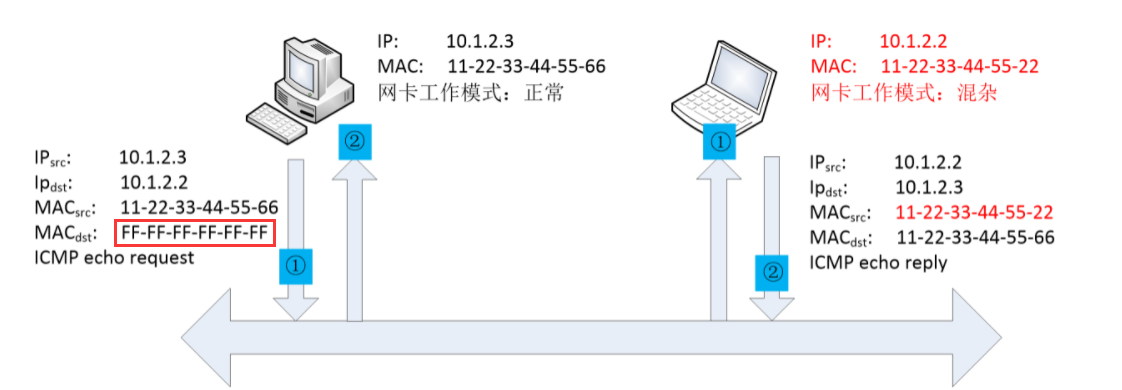
P45

1. 确保以太网整体安全性
2. 采用加密技术
3. 如使用RSA加密算法等。缺点是影响数据传输速度以及使用一个弱加密术比较容易被攻破。系统管理员和用户需要在网络速度和安全性上进行折中。
4. 交换机（原理？例子？）
5. 启用并正确配置交换机的安全机制。将交换机物理端口和MAC地址静态绑定，限制交换机单个物理端口可以动态绑定的MAC地址数量。以交换式集线器代替共享式集线器对局域网的中心交换机进行网络分段后，局域网监听的危险仍然存在。这是因为网络最终用户的接入往往是通过分支集线器而不是中心交换机，而使用最广泛的分支集线器通常是共享式集线器。这样，当用户与主机进行数据通信时，两台机器之间的数据包（称为单播包Unicast Packet）还是会被同一台集线器上的其他用户所监听。因此，应该以交换式集线器代替共享式集线器，使单播包仅在两个节点之间传送，从而防止非法监听。当然，交换式集线器只能控制单播包而无法控制广播包（Broadcast Packet）和多播包（Multicast Packet）。但广播包和多播包内的关键信息，要远远少于单播包。
6. 划分VLAN
7. 划分VLAN虚拟局域网，可以将以太网通信变为点到点通信，从而防止大部分基于网络监听的入侵。
8. 一次性口令
9. Kerberos等第三方服务提供认证机制

### 检测手段

P46

1. 反应时间
2. DNS测试
3. 地址反向解析：通过IP查域名
4. Ping
5. 由于在正常模式下，主机只接收目的 MAC 地址是自己的数据包，其他将一律丢弃；而在混杂模式下，网卡并不检查目的 MAC 地址，对所有的数据包都来者不拒。所以，我们只需要构造并发送一个：疑似待排查为混杂模式目标主机的 IP 目的地址和与其真实 MAC 地址不同的虚假 MAC 目的地址 ICMP 回显请求（echo request）数据包，如果收到应答，则说明该被排查的目标主机网卡当前正工作于混杂模式，即当前网络中可能存在监听者。



1. ARP数据包 （原理）

首先发送一个 ARP 广播包，这样 ARP 缓存表中便包括了所有在线的主机 IP 到 MAC 地址的映射，随后再发送一个非广播 ARP，接着查看当前 ARP 缓存表，检查两表是否一致，若有异常则说明存在监听。其中arp -a命令用来查看当前 ARP 缓存表。（ARP举例别用FF-FF-FF-FF-FF-FF，那是ARP广播包。原理也是混杂模式不看MAC\_addr）



1. 检测处于混杂模式的节点

网卡和操作系统对于是否处于混杂模式会有一些不同的行为，利用这些特征可以判断一个机器是否运行在混杂模式下

1. 一些检测手段
2. 根据操作系统的特征

Linux内核的特性：正常情况下，只处理本机MAC地址或者以太广播地址的包。在混杂模式下，许多版本的Linux内核只检查数据包中的IP地址以确定是否送到IP堆栈。因此，可以构造无效以太地址而IP地址有效的ICMP ECHO请求，看机器是否返回应答包(混杂模式)，或忽略(非混杂模式)。

Windows 9x/NT：在混杂模式下，检查一个包是否为以太广播包时，只看MAC地址前八位是否为0xff。

1. 根据网络和主机的性能

根据响应时间：向本地网络发送大量的伪造数据包，然后，看目标主机的响应时间，首先要测得一个响应时间基准和平均值

## 以太网络的工作原理

载波侦听/冲突检测(CSMA/CD, carrier sense multiple access with collision detection)技术

载波侦听：是指在网络中的每个站点都具有同等的权利，在传输自己的数据时，首先监听信道是否空闲

如果空闲，就传输自己的数据

如果信道被占用，就等待信道空闲

而冲突检测则是为了防止发生两个站点同时监测到网络没有被使用时而产生冲突

以太网采用了CSMA/CD技术，由于使用了广播机制，所以，所有与网络连接的工作站都可以看到网络上传递的数据

## 以太网卡的工作模式

网卡的MAC地址(48位)

通过ARP来解析MAC与IP地址的转换

用ipconfig/ifconfig可以查看MAC地址

正常情况下，网卡应该只接收这样的包

MAC地址与自己相匹配的数据帧

广播包

网卡完成收发数据包的工作，两种接收模式

**混杂**模式：不管数据帧中的目的地址是否与自己的地址匹配，都接收下来

**非混杂**模式：只接收目的地址相匹配的数据帧，以及广播数据包(和组播数据包)

为了监听网络上的流量，必须设置为混杂模式

## 嗅探器

P43

## 交换式以太网

**特点**

通过交换机连接网络

由交换机构造一个“MAC地址-端口”映射表

发送包的时候，只发到特定的端口上

在交换环境中，Sniffer的简单的做法就是伪装成为网关

**ARP欺骗**

**利用**：冒充网关，窃取消息

**防御**：

建立正确的 ARP 映射关系、检测并过滤伪造的 ARP 报文，保证经过其转发的 ARP报文正确合法；

抑制短时间内大量 ARP报文的冲击。

在无线网络中，防御ARP欺骗的有效方法是网络管理员分别在主机或接入点上和路由器或控制器上对IP和MAC地址进行静态映射绑定。（ARP欺骗是通过重复应答实现的，那么只需要在主机添加一条静态的ARP映射，这样就不需要询问网关MAC地址了；对于网关欺骗还需要在网关中也添加一条到主机的静态ARP映射。）

**交换技术：**

二层交换技术

二层交换技术是发展比较成熟，二层交换机属数据链路层设备，可以识别数据包中的MAC地址信息，根据MAC地址进行转发，并将这些MAC地址与对应的端口记录在自己内部的一个地址表中。

三层交换技术

二层交换技术＋三层转发技术

四层交换技术

决定传输不仅仅依据MAC地址(第二层网桥)或源/目标IP地址(第三层路由),而且依据

TCP/UDP(第四层) 应用端口号。

1. 包过滤/安全控制
2. 服务质量
3. 服务器负载均衡 :
4. 主机备用连接
5. 统计

## 共享式以太网

通过网络的所有数据包发往每一个主机，最常见的是通过HUB连接起来的子网

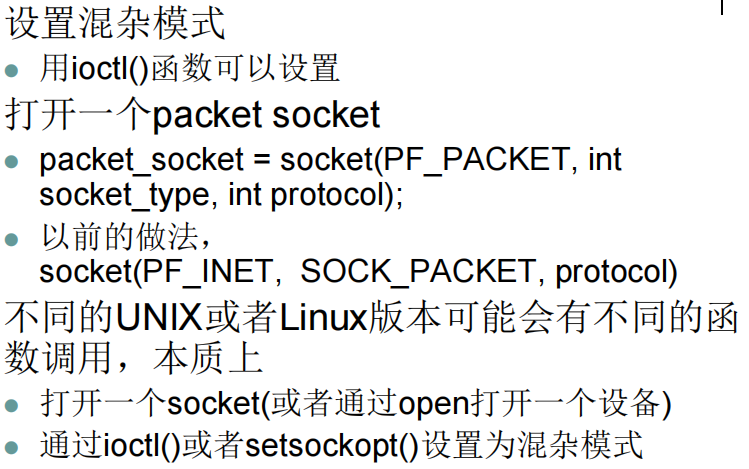
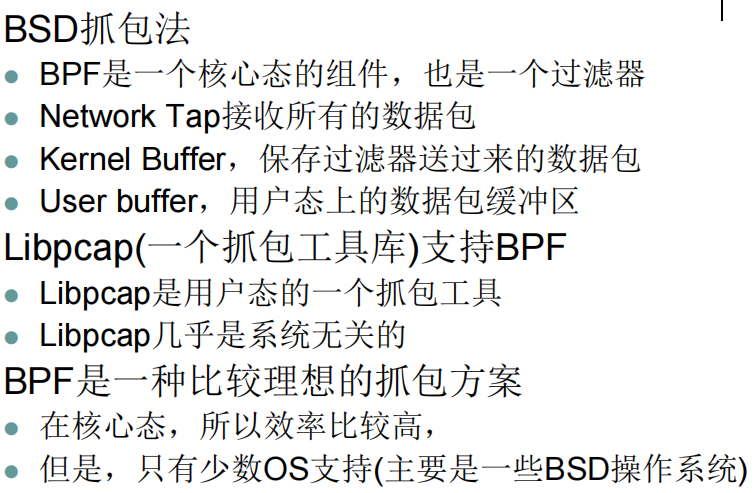
P39 共享以太网的四种工作模式：混杂模式下可以收听所有报文。

P40 设置为混杂模式的方法

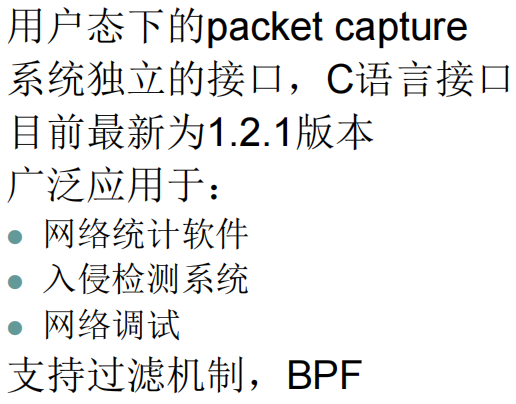
## 应用程序抓包的技术

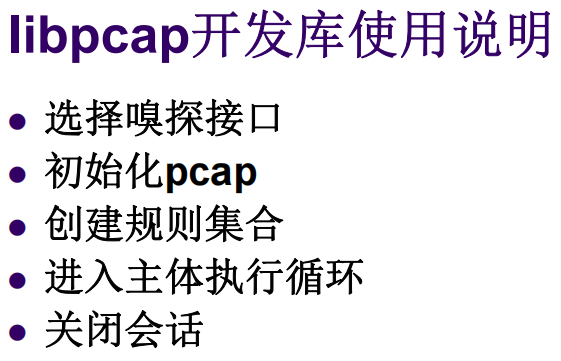
UNIX系统提供了标准的API支持

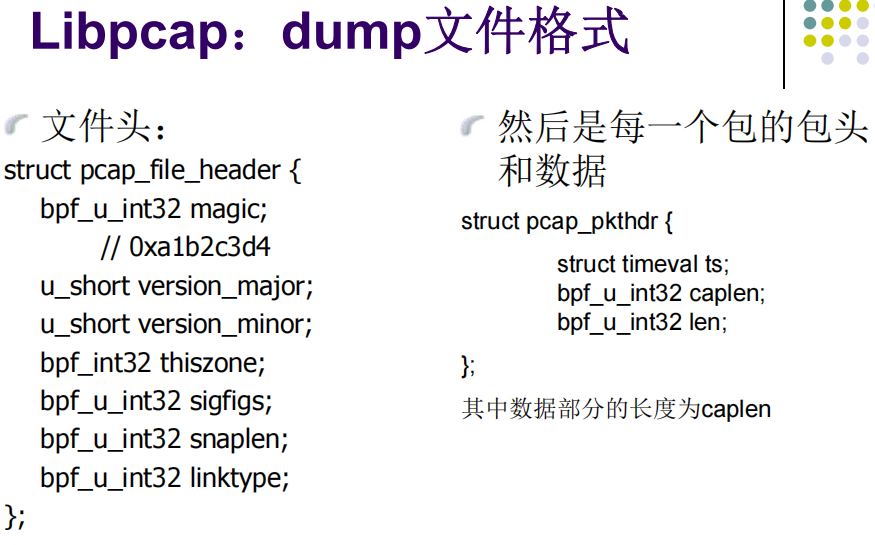
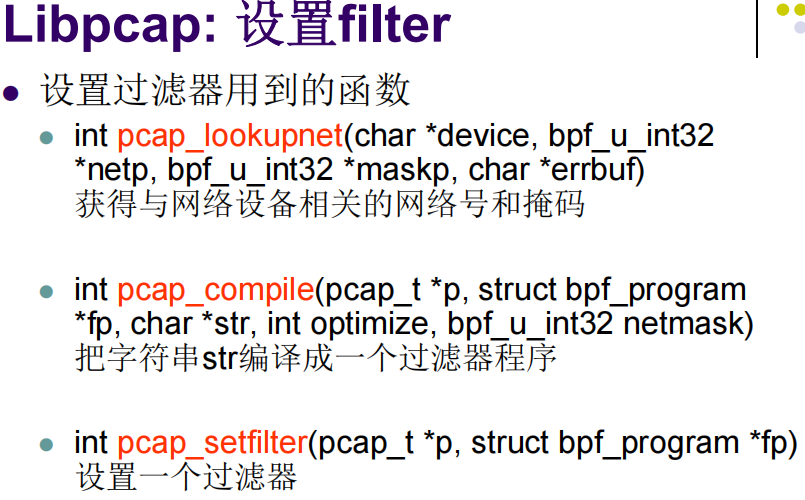
Packet socket BPF

### libpcap



Windows平台上通过驱动程序来获取数据包

内核本身没有提供标准的接口

通过增加一个驱动程序或者网络组件来访问内核网卡驱动提供的数据包

在Windows不同操作系统平台下有所不同

不同sniffer采用的技术不同

WinPcap是一个重要的抓包工具，它是libpcap的Windows版本

### WinPcap

WinPcap包括三个部分

第一个模块NPF(Netgroup Packet Filter)，是一个虚拟设备驱动程序文件。它的功能是过滤数据包，并把这些数据包原封不动地传给用户态模块，这个过程中包括了一些操作系统特有的代码

第二个模块packet.dll为win32平台提供了一个公共的接口。不同版本的Windows系统都有自己的内核模块和用户层模块。Packet.dll用于解决这些不同。调用Packet.dll的程序可以运行在不同版本的Windows平台上，而无需重新编译

第三个模块 Wpcap.dll是不依赖于操作系统的。它提供了更加高层、抽象的函数。

packet.dll和Wpcap.dll

packet.dll直接映射了内核的调用

Wpcap.dll提供了更加友好、功能更加强大的函数调用

windows网络结构

NDIS(Network Driver Interface Specification，网络驱动接口规范)描述了网络驱动与底层网卡之间的接口规范，以及它与上层协议之间的规范

NDIS支持三种类型的网络驱动程序：

网卡驱动程序(NICdrivers)

中间驱动程序 (InterMediateProtocolDrivers)

协议驱动程序 (UpperLevelProtocolDrivers)：

winpcap优势：

* 1. 提供了一套标准的抓包接口
  2. 与libpcap兼容，可使得原来许多UNIX平台下的网络分析工具快速移植过来
  3. 便于开发各种网络分析工具
  4. 除了与libpcap兼容的功能之外，还有
  5. 充分考虑了各种性能和效率的优化，包括对于NPF内核层次上的过滤器支持
  6. 支持内核态的统计模式
  7. 提供了发送数据包的能力

## 口令破解

P46

字典文件通常包含数据内容、类型知道即可。

# 第四章 拒绝服务攻击

定义、分类、攻击思想和方法、发起方式P56

**防御**：加强管理、机房管理、设备分离、定时检查各种配置、定时检查关键资源的使用情况、定时检查升级包

## 资源消耗破坏

**消耗网络资源实例：**

* **Smurf广播风暴攻击**

P58+p67

防御

1. 阻断虚假源地址（对应书上3）

2. 阻断WAN广播包（对应书上1）

3. 求助ISP，阻断某IP网段（对应书上2）

* **TCP SYN攻击**

p61

* **利用处理程序错误**

p69 ping of death、Teardrop、winnuke、Land

Teardrop防御方法：网络安全设备将接收到的分片报文先放入缓存中，并根据源IP地址和目的IP地址对报文进行分组，源IP地址和目的IP地址均相同的报文归入同一组，然后对每组IP报文的相关分片信息进行检查，丢弃分片信息存在错误的报文。为了防止缓存益处，当缓存快要存满是，直接丢弃后续分片报文。

* **ACK Flood攻击**

**原理：**

ack flood攻击是TCP连接建立之后，所有传输的TCP报文都是带有ACK标志位的数据包。

接收端在收到一个带有ACK标志位的数据包的时候，需要检查该数据包所表示的连接四元组是否存在，如果存在则检查该数据包所表示的状态是否合法，然后再向应用层传递该数据包。

如在检查中发现数据包不合法，如所指向的目的端口未开放，则操作系统协议栈会回应RST包告诉对方此端口不存在。

此时服务器要做两个动作，查表和回应ack/rst。

此类攻击一定要用大流量ack小包冲击才会对服务器造成影响。

根据tcp协议栈原理，随机源IP的ack小包应该会被server很快丢弃，因为在服务器的tcp堆栈中没有这些ack包的状态信息。

在实际测试中发现有一些tcp服务对ack flood比较敏感。

对于Apache或者IIS来说，几十kpps的ack flood不会构成威胁，但更高数量的ack flood冲击会造成网卡中断频率过高负载过重而停止响应。

jsp server在数量不多的ack小包冲击下jsp server很难处理正常的连接请求。

所以ack flood不仅危害路由器等网络设备，并且对服务器上的应用也有很大的影响。

**危害**：

带有超大载荷的ack flood攻击，会导致链路拥塞。

攻击报文到达服务器导致处理性能耗尽，从而拒绝正常服务。

极高速率的变源变端口ack flood攻击，很容易导致依靠会话转发的设备转发性能降低甚至成网络瘫痪。

**防御：**

攻击流量达到阈值后启动ack防护。

真实的报文经过ack重传之后，由客户端重新发起连接，此时会通过syn验证算法通过后加入白名单信任。

伪造的ack报文通过查询会话表直接丢弃。

过syn验证算法通过后加入白名单信任。

伪造的ack报文通过查询会话表直接丢弃

僵尸主机多的话怎么办？可以用防范DDoS的思路答。

**消耗存储资源实例：**

Teardrop

FORK

## 电子邮件轰炸

危害 p72

邮箱列表炸弹KaBoom、病毒发送电子邮件炸弹、应对方法 p73

常用攻击工具：upyours4、KaBoom3、HakTek、Avalanche

## 分布式拒绝服务攻击DDoS

攻击者首先侵入并控制一些计算机，然后控制这些计算机同时向一个特定的目标发起拒绝服务攻击

### DDoS和DoS区别

P74

### DDoS特点

由于集中了成百上千台机器同时进行攻击，其攻击力是十分巨大的。即使像Yahoo，Sina等应用了可以将负荷分摊到每个服务器的集群服务器技术，也难以抵挡这种攻击。

多层攻击网络结构使被攻击主机很难发现攻击者

而且大部分装有主控进程和守护进程的机器的合法用户并不知道自己是整个拒绝服务攻击网络中的一部分，即使被攻击主机监测到也无济于事。

### 被DDoS攻击时的现象

p74

### DDoS工具

Trinoo

UDP

TFN（Tribe Flooding Network）

Stacheldraht

TFN2K（Tribe Flooding Network 2000）

多点攻击、加密传输、完整性检查、随机选择底层协议和攻击手段、IP地址欺骗、哑代理、隐藏身份等特点

Trinity v3

### DDoS攻击过程

**主要步骤：攻占代理主机 + 向目标发起攻击**

1.探测扫描大量主机以寻找可入侵主机；

2.入侵有安全漏洞的主机并获取控制权；

3.在每台被入侵主机中安装攻击所用的客户进程或守护进程；

4.向安装有客户进程的主控端主机发出命令，由它们来控制代理主机上的守护进程进行协同入侵。

**详细步骤：**

黑客利用工具扫描Internet，发现存在漏洞的主机

黑客在非安全主机上安装类似“后门”的代理程序

黑客选择主控主机，用来向“僵尸”发送命令

通过客户端程序，黑客发送命令给主控端，并通过主控主机启动“僵尸”程序对目标系统发动攻击

主控端向“僵尸”发送攻击信号，对目标发动攻击

目标主机被“淹没”，无法提供正常服务，甚至系统崩溃

### DDoS应对方式

P76

在数据流中搜寻特征字符串，尽管攻击包加入伪装，通过字符串特征提取，确定攻击者位置

利用攻击数据包的特征

设置防火墙

1. 监视本地主机端口的使用情况

2. 在防火墙服务器上放置一份 ACL（访问控制列表) 来阻断这些来自一些IP的访问。

对通信数据量进行统计

### DDoS常用方式

p76

**Trinoo（Tribe Flood Network）攻击**

用UDP包进行攻击的工具软件

与针对某特定端口的一般UDP flood攻击相比，Trinoo攻击随机指向目标端的各个UDP端口，产生大量ICMP不可到达报文，严重增加目标主机负担并占用带宽，使对目标主机的正常访问无法进行

**TFN攻击**

用ICMP给主控端或分布端下命令 ，其来源可以做假

发动SYN flood 、UDP flood 、ICMP flood及Smurf(利用多台服务器发出海量数据包，实施DoS攻击 )等攻击

**TFN2K攻击**

TFN2K是TFN的增强版，它增加了许多诸如加密新功能

**Stacheldraht攻击**

结合了Trinoo和TFN的特点

SHAFT是一种独立发展起来的DDoS攻击方法，独特之处在于：

首先，在攻击过程中，受控主机之间可以交换对分布端的控制和端口，这使得入侵检测工具难以奏效

其次，SHAFT采用了“ticket”机制进行攻击，使其攻击命令有一定秘密性

第三，SHAFT采用了独特的包统计方法使其攻击得以顺利完成

## 反弹技术

利用反弹服务器实现攻击的技术。

反弹服务器（Reflector）是指当收到一个请求数据报后就会产生一个回应数据报的主机

例如，所有的Web服务器、DNS服务器和路由服务器都是反弹服务器。攻击者可以利用这些回应的数据报对目标机器发动DDoS攻击

### 攻击步骤

传统DDoS第4步改为：攻击者锁定大量的可以作为反弹服务器的服务器群，攻击命令发出后，代理守护进程向已锁定的反弹服务器群发送大量的欺骗请求数据包，其原地址为受害服务器或目标服务器

### 反弹技术与传统DDoS区别

多了第四层——被锁定的反弹服务器层

反弹服务器的数量可以远比驻有守护进程的代理服务器多

使攻击时的洪水流量变弱，最终才在目标机汇合为大量的洪水

目标机更难追查到攻击来源

目标机接收到的攻击数据报的源IP是真实的，反弹服务器追查到的数据报源IP是假的。

# 第五章 缓冲区溢出攻击

## 缓冲区定义

连续的一段存储空间，动态变量在程序运行时被分配缓冲区

## 缓冲区溢出的定义

指写入缓冲区的数据量超过该缓冲区能容纳的最大限度，造成溢出的数据改写了与该缓冲区相邻的原始数据的情形。

## 造成缓冲区溢出的根本原因

代码在操作缓冲区时，没有对缓冲区边界进行检查

使得写入缓冲区的数据量超过缓冲区能够容纳的范围，从而导致溢出的数据改写了与该缓冲区相邻存储单元的内容。

C语言中许多字符串处理函数如：Strcpy、Strcat、Gets、Sprintf等都没有对数组越界加以检测和限制。

## 缓冲区溢出的危害

利用缓冲区溢出实现在本地或者远程系统上，实现任意执行代码的目的，从而进一步达到对被攻击系统的完全掌控；

利用缓冲区溢出进行DoS(Denial of Service)攻击；

利用缓冲区溢出破坏关键数据，使系统的稳定性和有效性受到不同程度的影响；

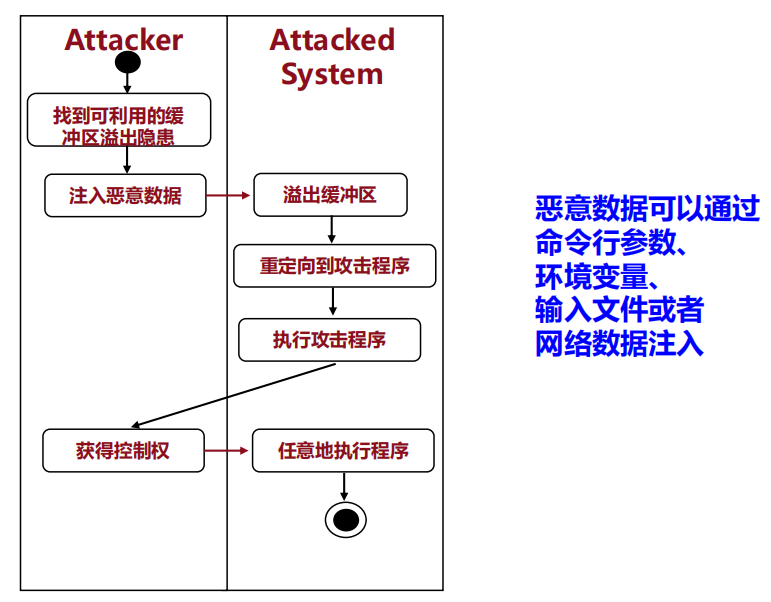
实现蠕虫程序

## 缓冲区溢出的目标

缓冲区溢出可以成为攻击者实现攻击目标的手段，但是单纯地溢出缓冲区并不能达到攻击的目的

在绝大多数情况下，一旦程序中发生缓冲区溢出，系统会立即中止程序并报告“fault segment”。例如缓冲区溢出，将使返回地址改写为一个非法的、不存在的地址，从而出现core dump错误，不能达到攻击目的。只有对缓冲区溢出“适当地”加以利用，攻击者才能通过其实现攻击目标。超出缓冲空间，覆盖正常程序或数据，使计算机转向预测程序

## 缓冲区溢出工作原理



## 进程在内存中的布局



## 缓冲区溢出可能发生的位置

堆栈(stack)

潜在危害：改写返回地址、改写调用函数栈的栈帧地址、改写函数指针、改写虚函数指针、改写异常处理指针、改写数据指针

堆(heap)

数据段(data)

BSS段

## 常见的溢出缓冲区的途径

利用C的标准函数库，常见的有strcpy、strcat、sprintf、gets

利用数组下标的越界操作。Off-by-one缓冲区溢出：利用数组的最大下标与数组长度的差异产生。

利用有符号整数与无符号整数的转换

恶意代码（注入）。利用已在内存中的代码作为攻击代码，但是需要提供调用这些现成代码产生攻击的参数。

## 防御技术

### 黄金规则

如果一个函数从某个不可信赖的来源接收到缓冲区，请遵守以下这些规则：

要求代码传递缓冲区长度

检查内存

采取防御措施

### 采取防御措施

在操作或复制不可信赖的数据，时如果误用了某些函数，这些函数就有可能存在严重的安全问题。

当检查代码中的缓冲区溢出错误时，应该随着数据在代码内的流动而对其进行跟踪，并质疑该数据的有关假设。

如果编写 C++ 代码，请考虑使用自定义字符串操作类来对字符串进行操作，而不要直接对各字节进行操作。唯一潜在的缺点是可能会导致性能下降

### 防御技术

基于软件的防御技术

类型安全的编程语言。

类型安全近似于所谓的内存安全（就是限制从内存的某处，将任意的字节复制到另一处的能力）Java, C#, Visual Basic等属于类型安全的编程语言缺点：性能代价、类型安全的编程语言自身的实现可能存在缓冲区溢出问题

相对安全的函数库

例如在使用C的标准库函数时，做如下替换

strcpy -> strncpy

strcat -> strncat

使用不当仍然会造成缓冲区溢出问题。

修改的编译器

增强边界检查能力的C/C++编译器

提供动态检测缓冲区溢出的能力

返回地址的完整性保护

将堆栈上的返回地址备份到另一个内存空间；

在函数执行返回指令前，将备份的返回地址

重新写回堆栈。

缺点：性能代价、检查方法仍不完善

内核补丁

将堆栈标志为不可执行来阻止缓冲区溢出攻击

将堆或者数据段标志为不可执行

例 如 Linux 的 内 核 补 丁 Openwall 、 RSX 、 kNoX 、ExecShield和PaX等实现了不可执行堆栈，并且RSX、 kNoX、ExecShield、PaX还支持不可执行堆。

缺点：对于一些需要堆栈/堆/数据段为可执行状态的应用程序不合适；需要重新编译原来的程序，如果没有源代码，就不能获得这种保护。

静态分析方法

**字典检查法**

➢ 遍历源程序查找其中使用到的不安全的库函数和系统调用。

➢ 缺点：误报率很高，需要配合大量的人工检查工作。

**程序注解法**

➢ 包括缓冲区的大小，指针是否可以为空，输入的有效约定等等。

➢ 缺点：依赖注释的质量

**整数分析法**

➢ 将字符串形式化为一对整数，表明字符串长度(以字节数为单位)以及目前已经使用缓冲区的字节数。通过这样的形式化处理，将缓冲区溢出的检测转化为整数计算

➢ 缺点：仅检查C中进行字符串操作的标准库函数。检查范围很有限

**控制流程分析法**

➢ 将源程序中的每个函数抽象成语法树，然后再把语法树转换为调用图/控制流程图来检查函数参数和缓冲区的范围。

➢ 缺点：对于运行时才会显露的问题无法进行分析 、存在误报的可能

动态检测方法

Canary-based 检测方法

➢ 将canary(一个检测值)放在缓冲区和需要保护的数据之间，并且假设如果从缓冲区溢出的数据改写了被保护数据，检测值也必定被改写。

➢ 缺点：很多工具通过修改编译器实现检测功能，需要重新编译程序；这种方法无法检测能过绕过检测值的缓冲区溢出攻击。

输入检测方法

➢ 向运行程序提供不同的输入，检查在这些输入条件下程序是否出现缓冲区溢出问题。不仅能检测缓冲区溢出问题，还可以检测其它内存越界问题。

➢ 缺点：系统性能明显降低。输入检测方法的检测效果取决于输入能否激发缓冲区溢出等安全问题的出现。

基于硬件的防御技术

处理器结构方面的改进

64位处理器的支持

➢ Intel/AMD 64位处理器引入称为NX(No Execute)或者AVP(Advanced Virus Protection)的新特性，将以前的CPU合为一个状态存在的“数据页只读”和“数据页可执行”分成两个独立的状态。

➢ ELF64 SystemV ABI通过寄存器传递函数参数而不再放置在堆栈上，使得64位处理器不仅可以抵制需要注入攻击代码的缓冲区溢出攻击还可以抵制returnto-libc这类的攻击。

➢ 缺点：无法抵制“borrowed code chunks”这类的攻击

## 哪些编程习惯会导致缓冲区溢出

* 未进行缓冲区越界检查

不知道内存缓冲区的总长度

不能信任来自用户的数据，调用字符串处理函数等库函数时，对数组越界加以检查和限制。（比如下列例子就不知道szName长度）也可以见project1-vul1

　　void Function(char \*szName) {

　　char szBuff[MAX\_NAME];

　　// 复制并使用 szName

　　strcpy(szBuff,szName);

　　 }

由于strcpy不检查内容长度，所以有溢出风险。因此Vul1中需要利用地点就是strcpy()。

* 不安全的库函数调用

PPT strcpy->strncpy

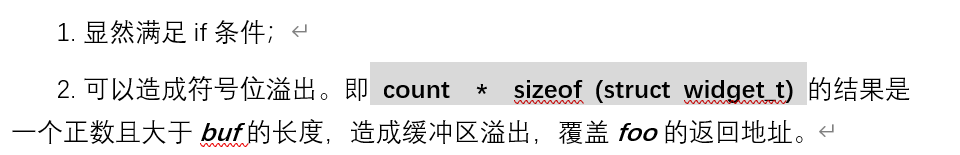
见project1-vul5 格式化字符串

漏洞在snprintf()函数中，因为是将arg作为格式化字符串拷贝（遇到格式化参数之前会先将普通字符拷贝给目标），所以可以从这里下手。除了常见的格式化字符串‘%d’、‘%s’等，%$(如：%20$x)格式输出，可泄漏栈内的数据；%n格式输出，可修改栈中指针指向的任意可写地址。

* 有无符号类型转换

见project1-vul3





* \*多次free() 结合UAF进行攻击

eg. 假设一个对象A前4字节(0x400000)为函数printf，释放后，恶意用户重用这个空间（使用缓冲区溢出等方式），将前4字节改为(0x4a0000)system函数，然后来到漏洞的触发点，程序由于疏忽再次调用A的printf，却调用了system函数

## Payload

### 构造流程

P82

* 准备shellcode

### Unix&Windows

P99 地址有/x00

P101 无系统调用

### 构造payload注意事项

* 真实内存地址未知

P94 使用相对IP的偏移地址

* Shellcode中不能存在\x00

P95 使用xor代替mov

* 存在阻断函数\_exit(0)

见project1-vul6，观察\_exit(0)跳转地址，直接修改其为&shellcode

## 如何在内存中找到一个jmp esp指令(二进制码为0xffe4)？

可以在目标exe或者dll中寻找，也可以在系统dll中寻找，但是系统dll有版本相依性

## 如果溢出函数的返回指令有偏移(比如ret 8)，则esp往后移动了

解决的办法是在shellcode的前面加上多个nop(二进制码为0x90)

## Windows下编写shellcode

### 如何调用系统函数

汇编代码是使用call指令，如何确定call后面的指令？我们不能使用import table

我们使用GetProcAddress和LoadLibraryA来解决

这两个函数的地址可以从exe的import table中找到

做法：把所有需要用到的系统函数都放到一张数据表中，然后通过GetProcAddress找到这些函数的地址

### 消除代码中的null字节

在数据区，加减一个常量来避免出现null，这样在代码刚开始执行的时候，要先恢复数据区

在指令区，通过一些技巧来获得0，比如

xor eax, eax

要赋值0x00441110给ebx，则可以

mov ebx, 44111099

shr ebx, 08

跳转的时候使用相对地址，如果出现0，则可以用nop来避免

# 第六章 程序攻击

## 逻辑炸弹攻击

P109

## 后门

p114

## 后门攻击方法

获取尽可能多的用户口令，并不会被管理员察觉或查封

更改配置

例如：rhosts

替换程序（包括源代码，函数库，内核）

要点：时间、校验和

开设新的服务，定时开启服务

## 可用作后门的系统漏洞

登录程序后门

其身份验证过程可能存在漏洞, 使用这类后门可以方便地登录系统, 并且不容易被发现

网络服务后门

一些系统服务程序或应用服务程序中存在着漏洞, 如系统服务Telnet,Ftp, E-mail, Rlogin 等和应用服务IRC, OICQ 等

系统库后门

系统库用于函数的重用而减少代码长度

利用系统库来安装后门可以做 到很隐秘, 成功率也很高

内核后门

内核是操作系统工作的核心, 但内核模块的处理过程存在一些漏洞

### 后门和木马的区别

在PPT“蠕虫和特洛伊木马”最后

所谓“后门”，就是程序开发人员为了改进自己设计的程序而打开的专用接口（通道），方便自己修改程序，一般具有最高权限，特征在于隐蔽性，没有**欺骗性**。

后门是一个允许攻击者绕过系统中常规安全控制机制的程序，他按照攻击者自己的意图提供通道。

后门的重点在于为攻击者提供进入目标计算机的通道

木马是指利用后门或漏洞被发现非法入侵用户电脑，从事侵害用户利益的活动，不一定具有最高权限。

如果一个程序仅仅提供远程访问，那么它只是一个后门。

如果攻击者将这些后门伪装成某些其他良性程序，那么那就变成真正的特洛伊木马。

联系在于:都是隐藏在用户系统中向外发送信息,而且本身具有一定权限,以便远程机器对本机的控。

区别在于:木马是一个完整的软件,而后门则体积较小且功能都很单一。

### 后门的隐藏方式

**隐藏**

代码：坏扇区，Boot

通讯：TCP，UDP和ICMP

Shell后门：TCP/UDP/ICMP

**隐藏执行**

DLL

Rundll32.exe

Dllcache

动态嵌入，挂接API，全局钩子（HOOK），远程线程

* 应用级隐藏：通过修改、捆绑或替代系统合法的应用程序来实现隐藏。早期的后门一般是在应用级上实现隐藏

1. 后门程序和进程隐藏 选择一个隐秘的目录存放后门程序文件

2. 后门程序设置隐藏标记。

3. 定制程序名、修改图标。将自己的服务 端程序图标改成 HTML, TXT, ZIP 等各种文件的图

4. 附着、捆绑或替换合法文件。

5. 修改或替换管理命令。如替换“ ls”“find”“du”可以实现文件和目录隐藏; 替换 “ ps”“top”“ pidof”可以实现进程隐藏; 替换 Netstat, Ifconfig 可 实现网络的连接状态隐藏; 替换 Kill, Killall 可防止删除攻击者 的进程, 替换 Crontab 能隐 藏攻击者的定 时启动信息, 替 换 Tcpd, Syslogd 隐藏攻击者的连接信息等。

6. 进程动态注入。将后门程序可执行代码注入正在运 行的进程中, 可以极大地提高后门的生存能力。此项技术可以 很好地隐藏后门程序, 并且植入后门代理程序后

* 通信状态隐藏

1. 通信端口隐藏

2. 将后门隐藏在合法端口,

3. 通信连接隐藏

检测方式：通信端口检查、通信特征匹配、植入痕迹查找和完整性检查

* 内核级隐藏

LKM( Loadable Kernel Modules) 内核加载模块隐藏，另一种是通过系统库来实现隐藏

检测方式：内核级后门是在内核级隐藏目录、文件、进程和通信连接等, 它不修改程序二进制文件, 因此 MD5 校验无法检测。内核级隐藏的一种方法是通过修改系统调用表来实现系统调用重定向从而隐藏文件、进程和通信连接, 所以通过检查系统调用的内存地址就可能发现是否被植入了后门

常用查杀工具：rkhunter，chkrookit

## 远程监控技术

对计算机的监视包括对主机的鼠标、键盘以及屏幕显示甚至网络通讯流量流向等的监视，也包括对对方计算机系统信息（包括磁盘信息、操作系统信息及硬件信息）的搜集

远程控制则是攻击者控制目标机，按照自己的意愿在被攻击计算机上运行程序或者关闭对方的功能，包括控制对方的鼠标、键盘、操作系统，在对方计算机上启动服务，或者关闭对方计算机等

## 特洛伊木马

p129

反弹型木马、嵌入式木马

## 木马的种类

网络游戏木马

网银木马

即时通讯软件木马

发送消息型、盗号型、传播自身型

网页点击类木马

下载类木马

代理类木马

## 木马的欺骗手段

名字欺骗

修改文件的文件名以欺骗用户

与Windows扩展名放在一起

Beauty.jpg .exe

模仿其他文件名

httpd,iexplore,notepad,ups,svchost…

不能用“任务管理器”删除的进程名

Csrss.exe,services.exe,smss.exe,winlogon.exe,system,system idle process

路径威胁

如将木马命名为explorer.exe放在C:\下。

防御：确定指定进程属于哪个程序Fport,icesword,tcpview…

使用杀进程工具进行查杀Pskill,icesword…

文件捆绑

恶意程序与正常程序捆绑

捆绑工具

EXE捆绑机,Wrappers,binders,EXE binders……

压缩软件

WinRMSetup30.exe

防御

使用反病毒软件进行检测，并及时更新病毒库

## 防御木马、后门

培养良好的安全意识和习惯。

使用网络防火墙封锁与端口的连接。

仅允许最少数量的端口通信通过防火墙

天网个人防火墙，瑞星防火墙，江民黑客防火墙，Zone Alarm，Norton Personal Firewall……

经常利用端口扫描器扫描主机或端口查看工具查找本地端口监听程序。

Nmap，Xscan，NC，Fport，TcpView，IceSword……

## 蠕虫

技术等级相当高，利用提供网络服务的软件漏洞来达到在远程计算机上的自我复制

步骤

1. 入侵计算机，传入网络

2. 获得用户账号

3. 找出提供直接访问而无需身份鉴定的主机，实现自我复制

## 病毒、蠕虫和特洛伊木马区别

蠕虫的其他方法 P125

**病毒 蠕虫 木马的区别**

普通病毒需要传播受感染的驻留文件来进行复制

蠕虫不使用驻留文件即可在系统之间进行自我复制

木马表面上是有用的软件、实际目的却是危害计算机安全并导致严重破坏的计算机程序。

它是具有欺骗性的文件(宣称是良性的，但事实上是恶意的) ，是一种基于远程控制的黑客工具，具有隐蔽性和非授权性的特点。

首先病毒，木马，蠕虫统称为电脑病毒。病毒（包含蠕虫）的共同特征是**自我复制、传播、破坏电脑文件，对电脑造成数据上不可逆转的损坏**。而木马独有特征是**伪装成正常应用骗取用户信任而入侵，潜伏在电脑中盗取用户资料与信息**。

病毒：编制者在计算机程序中插入的破坏计算机功能或者数据的代码，能影响计算机使用，能**自我复制**的一组计算机指令或者程序代码。

1、它必须能自行执行。它通常将自己的代码置于**另一个程序的执行路径中**。

2、它必须能**自我复制**。例如，它可能用受病毒感染的文件副本替换其他可执行文件。病毒既可以感染桌面计算机也可以感染网络 服务器。

木马：木马也称木马病毒，是指通过特定的程序来控制另一台计算机。与一般的病毒不同，**它不会自我繁殖**，也专并不“刻意”地去感染其他文件，它通过将自身伪装**吸引用户下载执行**，向施种木马者提供打开被种主机的门户，使施属种者可以任意毁坏、窃取被种者的文件，甚至远程操控被种主机。

蠕虫病毒：蠕虫病毒一种能够**利用系统漏洞通过网络进行自我传播的恶意程序**。它**不需要附着在其他程序上**，而是独立存在的。当形成规模、传播速度过快时会极大地消耗网络资源导致大面积网络拥塞甚至瘫痪。通过网络传播的恶性病毒，它具有病毒的一些共性，如传播性、隐蔽性、破坏性等等，同时具有自己的一些特征，如**不利用文件寄生(有的只存在于内存中)， 对网络造成拒绝服务，以及和黑客技术相结合**。普通病毒需要传播受感染的驻留文件来进行复制，而蠕虫**不使用驻留文件即可在系统之间进行自我复制**，普通病毒的传染能力主要是针对计算机内的文件系统而言，而蠕虫病毒的传染目标是互联网内的所有计算机。

按照传输方式分类 P130

# 第七章 欺骗攻击

## DNS欺骗攻击

基本原理、前提条件 P158

## DNS欺骗的方法

中间人（MITM） - 拦截用户和DNS服务器之间的通信，以便将用户路由到不同的/恶意的IP地址。

DNS服务器危害 - 直接劫持DNS服务器，该服务器配置为返回恶意IP地址。

## DNS欺骗检测和防范思路

### 检测方法

发生DNS欺骗时，Client最少会接收到两个以上的应答数据报文，报文中都含有相同的ID序列号，一个是合法的，另一个是伪装的。据此特点，有以下两种检测办法：

(1)被动监听检测。即监听、检测所有DNS的请求和应答报文。通常DNS Server对一个请求查询仅仅发送一个应答数据报文(即使一个域名和多个IP有映射关系，此时多个关系在一个报文中回答)。因此在限定的时间段内一个请求如果会收到两个或以上的响应数据报文，则被怀疑遭受了DNS欺骗。

(2)主动试探检测。即主动发送验证包去检查是否有DNS欺骗存在。通常发送验证数据包接收不到应答，然而黑客为了在合法应答包抵达客户机之前就将欺骗信息发送给客户，所以不会对DNS Server的IP合法性校验，继续实施欺骗。若收到应答包，则说明受到了欺骗攻击。

### 防范思路

在侦测到网络中可能有DNS欺骗攻击后，防范措施有：①在客户端直接使用IP Address访问重要的站点，从而避免DNS欺骗; ②对DNS Server和Client的数据流进行加密，Server端可以使用SSH加密协议，Client端使用PGP软件实施数据加密。

对于常见的ID序列号欺骗攻击，采用专业软件在网络中进行监听检查，在较短时间内，客户端如果接收到两个以上的应答数据包，则说明可能存在DNS欺骗攻击，将后到的合法包发送到DNS Server并对DNS数据进行修改，这样下次查询申请时就会得到正确结果。

### 防护方法：

1 进行IP地址和MAC地址的绑定

 (1)预防ARP欺骗攻击。

 (2)DNS信息绑定。

2 使用Digital Password进行辨别

3.优化DNS SERVER的相关项目设置

4.直接使用IP地址访问

5.对DNS数据包进行监测

## 域名服务器安全DNSSEC

是一种通过添加其他验证方法来保护您的DNS的协议。该协议创建一个存储在DNS记录旁边的唯一密码签名，例如A记录和CNAME。然后，DNS解析器使用此签名来验证DNS响应，确保记录未被篡改。

## Email欺骗攻击

步骤、Email攻击方法 p161

### 应对方式

查看邮件原文，检查真正的发件服务器地址

通过邮件链接网页的时候，注意真正的网站地址

在不同的应用中，尽可能使用不相同的、无关的密码

## Web欺骗攻击

原理p163

### 欺骗的根源

由于Internet的开放性，任何人都可以建立自己的Web站点

Web站点名字(DNS域名)可以自由注册，按先后顺序

并不是每个用户都清楚Web的运行规则

### 欺骗的动机

商业利益、商业竞争、政治目的

### 欺骗的形式

使用相似的域名

注册与目标公司或组织相似的域名，建立一个欺骗网站，骗取用户信任，以便得到用户信息

例如，针对ABC公司，用abc.net来混淆abc.com

如果客户提供了敏感信息，那么这种欺骗可能会造成进一步的危害，例如：

用户在假冒的网站上订购了一些商品，然后出示支付信息，假冒的网站把这些信息记录下来(并分配一个cookie)

然后提示：现在网站出现故障，请重试一次。

当用户重试的时候，假冒网站发现这个用户带有cookie，就把它的请求转到真正的网站上。

用这种方法，假冒网站可以收集到用户的敏感信息。

改写URL

一个HTTP页面从Web服务器到浏览器的传输过程中，如果其中的内容被修改了的话，则欺骗就会发生，其中最重要的是URL改写

URL改写可以把用户带到不该去的地方，例如：<a href=www.hackersite> Welcom to Hollywood-Movie site.</a>

http://3525999368 /

有一些更为隐蔽的做法

直接指向一些恶意的代码

把url定向放到script代码中，难以发现

改写页面的做法

入侵Web服务器，修改页面

设置中间http代理

在传输路径上截获页面并改写

在客户端装载后门程序

劫持Web会话

HTTP协议不支持会话(无状态)，Web会话如何实现？

Cookie

认证

用url记录会话

用表单中的隐藏元素记录会话

Web会话劫持的要点在于，如何获得或者猜测出会话ID

### 防止Web欺骗

使用类似的域名

注意观察URL地址栏的变化

不要信任不可靠的URL信息

改写URL

查看页面的源文本可以发现

使用SSL

Web会话劫持

养成显式注销的习惯

使用长的会话ID

Web的安全问题很多，我们需要更多的手段来保证Web安全

短期方案

禁止浏览器JavaScript，各类改写信息原形毕露

确保浏览器连接状态可见，提供当前位置各类信息

不信任不可靠的URL信息

长期方案

改变浏览器，使之具有反应真是URL信息的功能

对于通过安全链接建立的Web，关注另一端身份

## IP欺骗攻击

IP欺骗过程p172

### 动机

隐藏自己的IP地址，防止被跟踪

以IP地址作为授权依据

穿越防火墙

### IP欺骗的形式

单向IP欺骗：不考虑回传的数据包

双向IP欺骗：要求看到回传的数据包

更高级的欺骗：TCP会话劫持

### IP欺骗成功的要诀

IP数据包路由原则：根据目标地址进行路由

### 程序实现IP欺骗

发送IP包，IP包头填上假冒的源IP地址

在Unix/Linux平台上，直接用socket就可以发送，但是需要root权限

在Windows平台上，不能使用Winsock

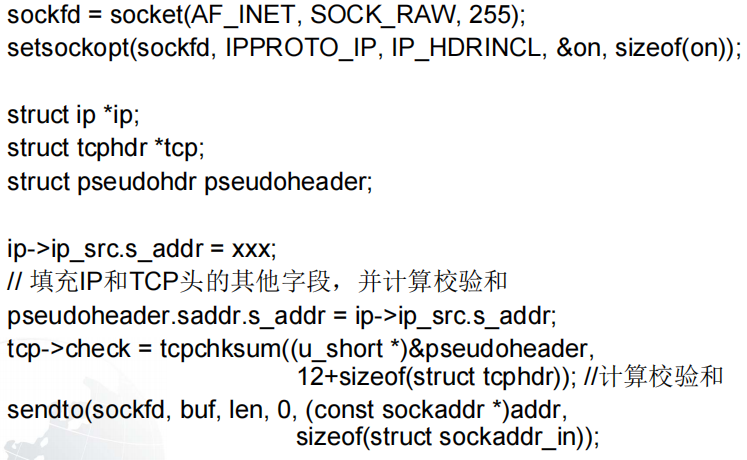
可以使用winpcap

可以用libnet构造IP包

代码示例

在Linux平台上，打开一个raw socket，自己填

写IP头和传输层数据，然后发送出去



### 避免IP欺骗

主机保护，两种考虑

保护自己的机器不被用来实施IP欺骗

物理防护、登录口令

权限控制，不允许修改配置信息

保护自己的机器不被成为假冒的对象

网络防护

路由器上设置欺骗过滤器

入口过滤，外来的包带有内部IP地址

出口过滤，内部的包带有外部IP地址

保护免受源路由攻击

路由器上禁止这样的数据包

## TCP会话劫持

### 会话

就是两台主机之间的一次通讯。例如你Telnet到某台主机，这就是一次Telnet会话；你浏览某个网站，这就是一次HTTP会话。

### 会话劫持（Session Hijack）

就是结合了嗅探以 及欺骗技术在内的攻击手段。例如，在一次正常的会话过程当中，攻击者作为第三方参与到其中，他可以在正常数据包中插入恶意数据，也可以在双方的会话当中进行监听，甚至可以是代替某一方主机接管会话。

### 欺骗和劫持

欺骗是伪装成合法用户，以获得一定的利益

劫持是积极主动地使一个在线的用户下线，或者冒充这个用户发送消息，以便达到自己的目的

### 动机

Sniffer对于一次性密钥并没有用

认证协议使得口令不在网络上传输

### 原理

TCP协议

三次握手建立TCP连接(即一个TCP会话)

终止一个会话，正常情况需要4条消息

如何标识一个会话：状态：源IP:端口+SN <> 目标IP:端口+SN

从TCP会话的状态入手

要了解每一个方向上的SN(数据 序列号)

两个方向上的序列号是相互独立的TCP数据包，除了第一个SYN包之外，都有一个ack标志，给出了期待对方发送数据的序列号，所以，猜测序列号是成功劫持TCP会话的关键

根据TCP/IP中的规定，使用TCP协议进行通讯需要提供两段序列号，TCP协议使用这两段序列号确保连接同步以及安全通讯，系统的TCP/IP协议栈依据时间或线性的产生这些值。

在通讯过程中，双方的序列号是相互依赖的，如果攻击者直接进行会话劫持，结果肯定是失败的。因为会话双方“不认识”攻击者，攻击者不能提供合法的序列号;所以，会话劫持的关键是预测正确的序列号，攻击者可以采取嗅探技术获得这些信息。

### 一般性过程

发现目标

找到什么样的目标，以及可以有什么样的探查手段，取决于劫 持的动机和环境

探查远程机器的ISN(初始序列号)规律

可以用nmap，或者手工发起多个连接

等待或者监听会话

最好在流量高峰期间进行，不容易被发现，而且可以有比较多 可供选择的会话

猜测序列号

这是最为关键的一步，如果不在一个子网中，难度将非常大

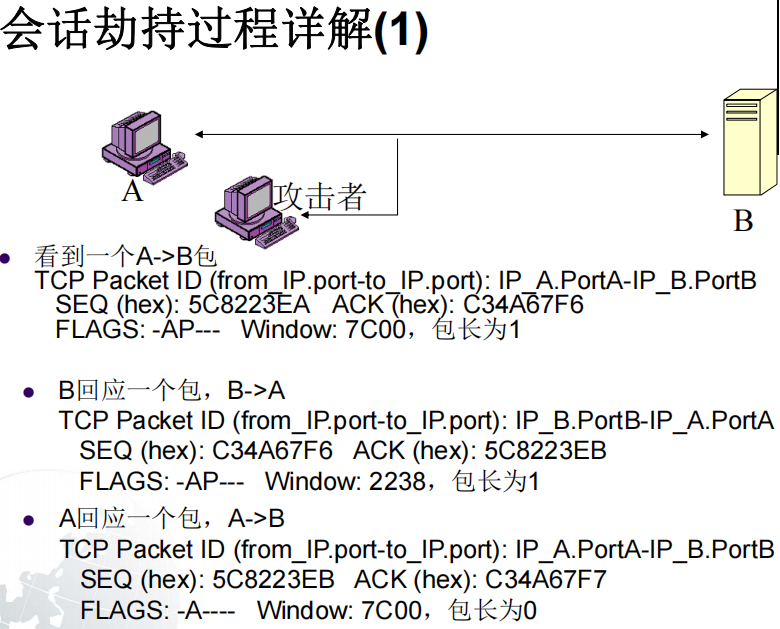
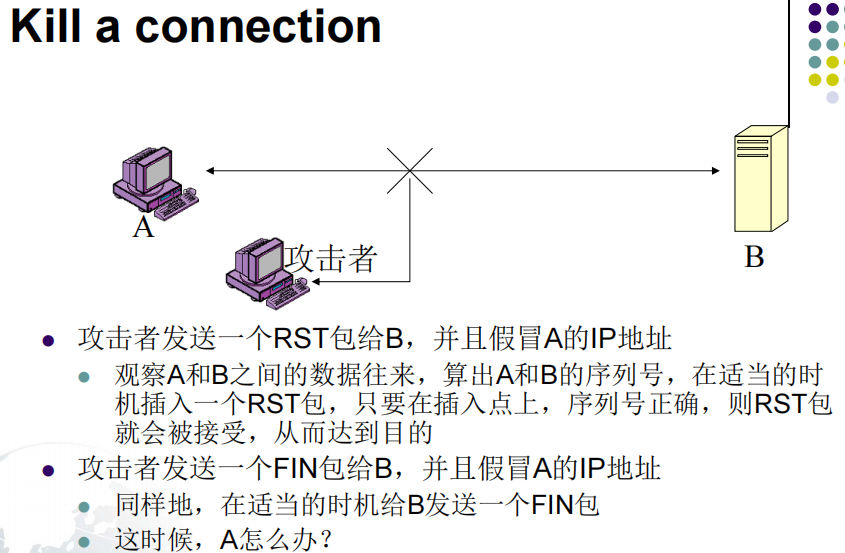
使被劫持方下线

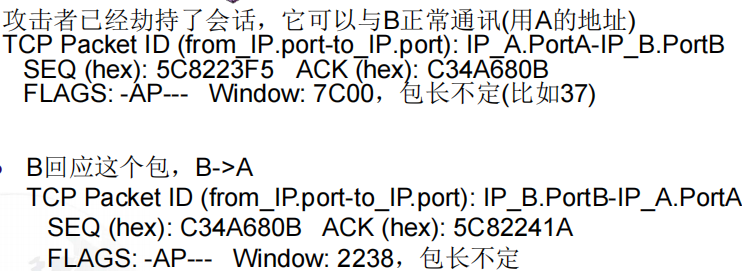
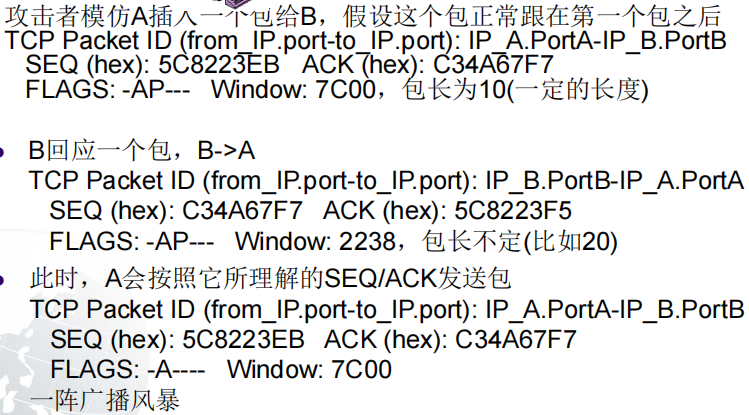
ACK风暴，拒绝服务

接管会话

如果在同一个子网中，则可以收到响应，否则要猜测服务器的动作

### 会话劫持详细过程





### TCP 会话劫持方法与形式

可以把会话劫持攻击分为两种类型：

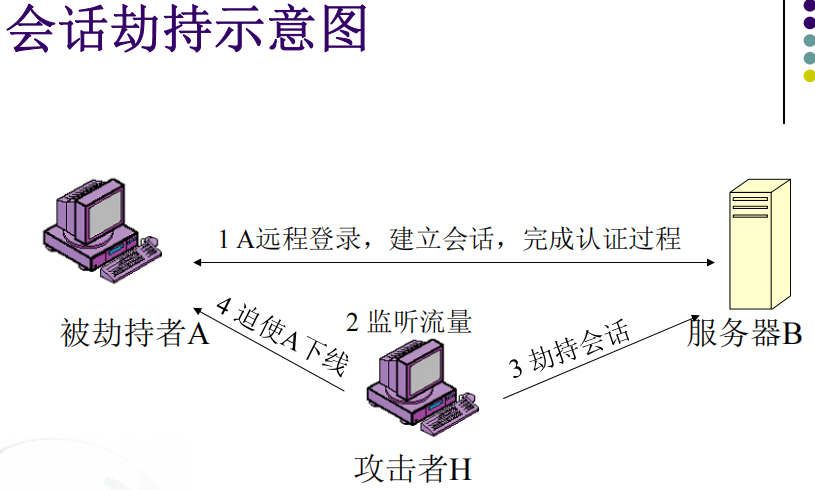
1）中间人攻击(Man In The Middle，简称MITM)；

2）注射式攻击（Injection）；

还可以把会话劫持攻击分为两种形式：

1）被动劫持:被动劫持实际上就是在后台监听双方会话的数据流，从中获得敏感数据。如在Telnet、FTP、HTTP、SMTP等传输协议中，用户和密码信息都是以明文格式传输的，若攻击者利用数据包截取工具便可很容易收集到帐户和密码信息。

2）主动劫持:主动劫持则是将会话当中的某一台主机“踢”下线，然后由攻击者取代并接管会话，这种攻击方法危害非常大，攻击者可以做很多事情，比如“cat etc/master.passwd”（FreeBSD下的Shadow文件）



### 不在一个子网中的劫持(欺骗)手法

有时候也称作“Blind spoofing”，攻击者发送一个SYN包然后猜测服务器的ISN，只要能够猜得到，就可以建立连接。但是攻击者收不到服务器给客户的包（使用源路由技术？）

条件：真正的客户不能发送RST包、攻击者能够猜测服务器每个包的大小

## TCP ACK风暴（ACK Storm）

当一个主机接收到一个不期望的数据包的时候， 它会用自己的序列号发送ACK，而这个包本身也是不可被接受的。于是，两边不停地发送ACK包，形成ACK包的循环，是为ACK风暴。

如果有一个ACK包丢掉，则风暴停止

在不同步的情况下，当服务器发送数据给客户

如果攻击者不对这份数据响应ACK的话，这份数据会被重传，因为服务器收不到ACK，并且会形成ACK风暴，最终，连接会被终止

如果攻击者对这份数据作出响应，则只有一个ACK风暴

### 如何到达不同步的状态

在建立连接的时候劫持会话

当攻击者听到握手过程第二步的时候，它给服务器发送一个RST包，然后发送用同样的TCP和端口号构造的一个SYN包，但是序列号与前面的SYN包不同

服务器关闭第一个连接，打开第二个连接，并且送回第二个SYN/ACK给客户，攻击者听到这个包之后，给服务器送出一个ACK包

至此，客户、服务器、攻击者都进入到TCP ESTABLISHED状态，但是攻击者和服务器之间是同步的，而客户和服务器之间是不同步的

注意，攻击者选择的序列号与客户的序列号一定要不同，否则不能成功

给一方发送空数据

攻击者首先观察会话

然后，给服务器发送一些无关紧要的数据，这些数据会导致服务器的序列号发生变化

攻击者给客户也可以发送数据

这种手段成功的要点在于可以发送一些无关紧要的数据，并且能够把握发送的时机

### 防止会话劫持

部署交换式网络，用交换机代替集线器

TCP会话加密

防火墙配置

限制尽可能少量的外部许可连接的IP地址

检测

ACK包的数量明显增加

### 会话劫持工具

Juggernaut可以进行TCP会话攻击的网络sniffer程序

Hunt功能与Juggernaut类似

功能特点

监听当前网络上的会话

重置会话(reset a session)

劫持会话，在劫持之后，使连接继续同步

确定哪些主机在线

四个守护进程

自动reset

Arp欺骗包的转发

收集MAC地址

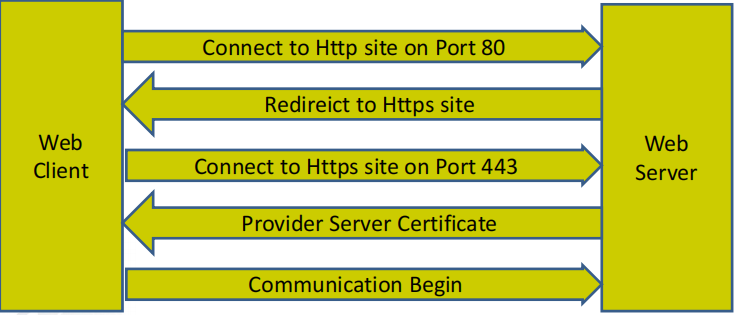
具有搜索功能的sniffer

TTY Watcher免费程序，针对单一主机上的连接

IP Watcher商用的会话劫持工具

## HTTP会话劫持

### 一般HTTPS通信过程



### 以访问Gmail为例的基本过程如下：

1. 客户端浏览器使用HTTP连接到端口80的http://mail.google.com；

2. 服务器试用HTTP代码302重定向客户端HTTPS版本的这个网站；

3. 客户端连接到端口443的网站https://mail.google.com；

4. 服务器向客户端提供包含其电子签名的证书，该证书用于验证网址；

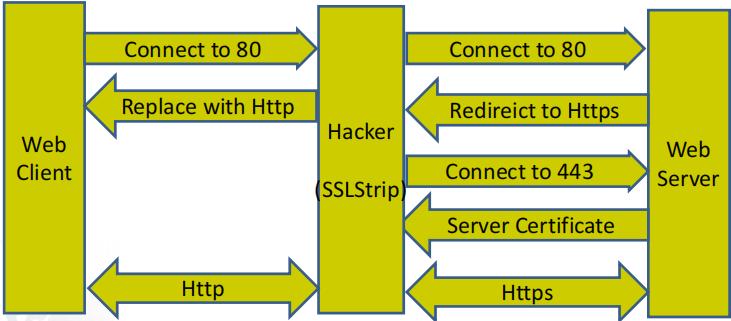
5. 客户端获取该证书，并根据信任证书颁发机构列表来验证该证书；

6. 加密通信建立。

如果证书验证过程失败的话，则意味着无法验证网址的真实度。这样的话，用户将会看到页面显示证书验证错误，或者他们也可以选择冒着危险继续访问网站，因为他们访问的网站可能是欺诈网站。

### SSLstrip工作原理：

SSLstrip通过监视Http传输进行工作，当用户试图进入加密的https会话时它充当代理。当用户认为安全的会话已经开始时，SSLstrip也通过https连接到安全服务器，所有用户到SSLstrip的连接是http，这就意味着浏览器上警告提示已经被阻止，浏览器看起来正常工作，在此期间所有的用户敏感信息都可以轻易被截获。



劫持HTTPS通信

1. 客户端与web服务器间的流量被拦截;

2. 当遇到HTTPS URL时，sslstrip使用HTTP链接替换它，并保存了这种变化的映射；

3. 攻击机模拟客户端向服务器提供证书；

4. 从安全网站收到流量提供给客户端；

这个过程进展很顺利，服务器仍然在接收SSL流量，服务器无法辨别任何改变。用户可以感觉到唯一不同的是，浏览器中不会标记HTTPS，所以某些用户还是能够看出不对劲。

## 采取哪些主要措施来保障WEB服务器安全

访问控制（IP地址限制、Windows帐户、请求资源的Web权限、资源的NTFS权限）

用虚拟目录隐藏真实的网站结构；

设置基于SSL的加密和证书服务，以保证传输安全；

完善定期审核机制；

安装防火墙及杀毒软件；

及时安装操作系统补丁，减少操作系统漏洞等等。

### 如何理解信息安全领域"三分技术，七分管理”这名话的

目前组织普遍采用现代通信、计算机、网络技术来构建组织的信息系统。但大多数组织的最高管理层对信息资产所面临的威胁的严重性认识不足，缺乏明确的信息安全方针、完整的信息安全管理制度，相应的管理措施不到位，如系统的运行、维护、开发等岗位不清，职责不分，存在一人身兼数职的现象。这些都是造成信息安全事件的重要原因。缺乏系统的管理思想也是一个重要的问题。

# 第八章 利用程序错误攻击

## windows

### windows漏洞

XP的热键漏洞

Windows Redirector

资源管理器内存破坏漏洞

帮助支持中心接口欺骗

不安全的脚本

### 漏洞防护

系统补丁

Internet连接防火墙

支持多用户的加密文件系统

改进的访问控制

对智能卡的支持

实例

p180

### ASP (Active Server Page)

当浏览器浏览ASP网页时, Web服务器就会根据请求生成相应的HTML代码然后再返回给浏览器,这样浏览器端看到的 就是动态生成的网页。

通过ASP，可能可以很方便地入侵web server、窃取服务器上的文件、捕获 web 数据库等系统的用户口令，甚至恶意删除服务器上的的文件。

## Unix

### Unix漏洞

处理畸形ELF二进制文件

拒绝服务攻击

ptrace

Samba共享应用系统

获得Root用户权限

惠普的Tru64Unix (Ipsec,SSH)

迫使服务器离线

## 黑客的攻击按如下六个步骤进行：

收集资料

取得普通用户的权限

远程登录

取得超级用户的权限

留下后门

清除日志

## Web入侵

定义、Web的安全问题、Web安全漏洞p197

### Web安全问题的来源

管理员为了管理方便而设立远程管理功能

为了方便用户使用而设立上传/下载机制

由于疏忽而缺乏应有的安全检查

为了省钱而使用不足够安全的软件和硬件

## CGI

定义、安全性、安全问题、CGI漏洞p198

## 跳板的作用是什么？

隐藏自己的地址，让别人无法查找到自己的位置

跳板机是一种安全中转节点，主要用于提供远程访问企业内部网络的通道（通俗来讲就是不能直接访问企业内部的服务器，必须通过跳板机这一层屏障才可以有机会访问企业内部的服务器）。

**跳板机的主要功能**

流量转发：跳板机可以将外部网络的流量转发到内部网络中，从而实现对内部网络的访问。

身份验证：跳板机可以对进入内部网络的用户进行身份验证，确保只有授权的用户才能访问内部网络。

安全管理：跳板机可以对内部网络的安全进行管理，例如限制某些应用程序的使用、限制访问某些网站等。

配置管理：跳板机可以对内部网络进行配置管理，例如对网络设备、应用程序进行配置。

记录和审计：跳板机可以记录内部网络的流量和使用情况，以便后续的追踪和审计。

## 如何避免多个服务系统之间的连带关系？

避免多个服务系统之间的连带关系是一项重要的系统设计和架构任务。下面是一些方法和技巧，可帮助减少服务系统之间的连带关系：

1. 松耦合（Loose Coupling）：确保系统中的服务组件之间松散耦合，即减少它们之间的依赖性。使用适当的接口和协议来定义服务之间的通信，并避免直接依赖于具体的实现细节。

2. 服务抽象化（Service Abstraction）：在设计服务系统时，通过使用抽象层和接口来隐藏底层服务的具体实现细节。这样可以减少服务之间的直接依赖，并使系统更具灵活性和可替换性。

3. 服务发现与注册（Service Discovery and Registration）：使用服务发现和注册机制，例如使用服务注册表、服务目录或中心化的服务发现工具，以便服务能够自动发现和识别其他服务的位置和功能。

4. 异步通信（Asynchronous Communication）：使用异步消息传递或事件驱动架构，以减少对其他服务的同步调用和直接依赖。通过异步通信，可以在系统中解耦服务之间的时间和空间依赖关系。

5. 服务网关（Service Gateway）：引入服务网关作为服务系统的入口，它可以负责路由和转发请求到不同的服务。服务网关可以隐藏底层服务的细节，并提供一致的接口给外部系统，从而减少系统之间的直接联系。

6. 事件驱动架构（Event-Driven Architecture）：采用事件驱动架构，将系统的不同组件之间的通信和交互建立在事件的基础上。通过发布-订阅模式，组件之间的关系变得更加松散，并可以减少直接的耦合。

7. 微服务架构（Microservices Architecture）：采用微服务架构将系统拆分为多个独立的、自治的服务。每个微服务负责特定的业务功能，并通过API进行通信。这样可以将系统的复杂性分解为较小的、可管理的部分，并减少不同服务之间的直接连带关系。

需要注意的是，减少服务系统之间的连带关系是一个复杂的设计任务，需要综合考虑系统的需求、可扩展性和可维护性。在设计和实施系统时，需要权衡各种因素，并选择合适的架构和技术来实现所需的解耦和自治性。

## 简述用ASP编写的网站的常见攻击方式有哪些？

泄漏ASP源代码

FileSystemObject组件篡改下载FAT分区上的任何文件的漏洞

输入标准的HTML语句或者JavaScript语句会改变输出结果

Access MDB数据库有可能被下载的漏洞

asp程序密码验证漏洞

IIS4或者IIS5中安装有Index Server服务器漏洞ASP源程序

NT Index Server存在返回上级目录的漏洞

1. SQL注入攻击：攻击者通过在输入字段中注入恶意的SQL代码来篡改或绕过数据库查询。这可能导致数据泄露、数据库被删除或篡改、未经授权的访问等问题。

2. XSS攻击（跨站脚本攻击）：攻击者在网页中插入恶意的脚本代码，当其他用户访问受影响的页面时，这些脚本会在他们的浏览器中执行。这可能导致用户会话劫持、窃取敏感信息或进行其他恶意操作。

3. CSRF攻击（跨站请求伪造攻击）：攻击者利用用户已经通过认证的会话，发送伪造的请求给目标网站。当受害者访问包含恶意请求的页面时，他们的操作会触发这些请求，导致未经授权的操作或数据泄露。

4. 文件包含攻击：攻击者利用网站中的漏洞，通过包含恶意文件或外部资源来执行恶意代码。这可能导致服务器受到控制、执行任意代码或访问敏感文件。

5. 会话劫持：攻击者窃取合法用户的会话凭证，然后使用这些凭证冒充用户，获取其权限并进行未经授权的操作。

6. 敏感数据泄露：由于错误的配置或漏洞，网站可能意外泄露用户的敏感信息，如用户名、密码、信用卡号等。

这些攻击方式都可以对使用ASP编写的网站造成严重的安全威胁。为了防范这些攻击，开发人员应采取安全编码实践，如输入验证和过滤、参数化查询、安全的会话管理、访问控制和安全配置。定期更新和修补网站以修复已知的漏洞也是重要的安全措施。

## 攻陷一个Windows server + IIS + ASP的网站

使用代理跳板做扫描+端口扫描+操作系统版本扫描+操作系统漏洞POC

验证+应用程序(IIS/CGI)漏洞POC测试+脚本注入漏洞测试

## 攻陷一个UNIX + CGI + Perl的网站

使用代理跳板做扫描+端口扫描+操作系统版本扫描+操作系统漏洞POC

验证+应用程序(IIS/CGI)漏洞POC测试+脚本注入漏洞测试

# 第九章 访问控制技术

原因、基本任务p207

## 主体、客体与访问控制关系

主体：信息系统中用户或进程，系统所有的用户与进程形成主体集合

客体：系统中被处理、被控制或被访问的对象（如文件、程序、存储器等）

访问控制关系：根据制定的系统安全策略，形成了主体与客体、主体与主体、客体与客体相互间的关系。有的属于访问性质的、有的属于信息的流动问题、要实现信息系统安全的目的，需要解决系统中的访问控制问题和信息流控制问题。

## 访问控制的有效性

建立在两个前提上

用户鉴别与确证

信息受保护，不会被非法修改

## 访问控制模式

网络的访问主要采用基于争用和定时两种方法

基于争用的方法意味着网上所有站点按先来先服务原则争用带宽

对网络的访问控制是为了防止非法用户进入系统和合法用户对系统的非法使用

## 访问控制功能

访问控制要对访问的申请、批准和撤消的全过程进行有效的控制

审计跟踪

对用户使用何种系统资源、使用的时间、执行的操作等问题进行完整的记录，以备非法事件发生后能进行有效的追查

附加控制

除了对直接的访问进行控制外，还应对信息的流动和推理攻击施加控制

## 访问控制的类型

自主访问控制（DAC）

强制访问控制（MAC）

用户和资源都是一个固定的安全属性，系统利用安全属性来决定一个用户是否可以访问某个资源

由于强制访问控制的安全属性是固定的，因此用户或用户程序不能修改安全属性

## 自主访问控制 DAC

### 定义

用户可以按自己的意愿对系统参数做适当的修改，可以决定哪个用户可以访问系统资源

由客体自主地来确定各个主体对它的直接访问权限（又称访问模式）

自主访问控制基于对主体或主体所属的主体组的识别来限制对客体的访问，这种控制是自主的

实现方法：

p219

前缀表缺点

主体的前缀表可能很大，增加了系统管理的困难。

只能由系统管理员进行修改。这种管理方法有些超出了DAC原则。

消与删除困难。

要系统回答“谁对某一客体具有访问权”这样的问题比较困难。但这个问题在安全系统中却是很重要的。

口令缺点

利用口令机制对客体实施的访问控制是比较麻烦的和脆弱的

系统不知谁访问了客体：对客体访问的口令是手工分发的，不需要系统参与

安全性脆弱: 需要把该客体的口令写在程序中，这样很容易造成口令的泄露。

使用不方便: 每个用户需要记忆许多需要访问的客体的口令，很不友好。

管理麻烦: 撤消某用户对某客体的访问权，只能改变该客体的口令，必须通知新口令给其他用户。

保护位

在保护位中包含了主体组的名字和拥有者的名字。保位机制中不包含可访问该客体的各个主体的名字。

由于保护位的长度有限，用这种机制完全表示访问矩阵实际上是不可能的。

访问控制表

该表中的每一项包括主体的身份和对该客体的访问权。

如果利用组或通配符的概念，可以使ACL表缩短。一般而言，一个单位内部工作内容相同的人需要涉及的客体大部分是相同的，把他们分在一个组内作为一个主体对待，可以显著减少系统中主体的数目。再利用通配符手段加快匹配速度，同时也能简化ACL表的内容。通配符用“\*”表示，可以代表任意组名或主体标识符。

基于行的DAC缺陷

着眼于某一个主体的访问权限，以主体为出发点描述控制信息，因此很难回答对于某一个客体而言，有哪些对它拥有访问的权限。

### 自主访问控制优缺点

优点

方便、实用

可由用户自由定制

可扩展性强

缺点

允许用户自主地转授访问权，这是系统不安全的隐患。

系统无法区分是用户合法的修改还是木马程序的非法修改；

无法解决因用户无意（如程序错误、某些误操作等）或不负责任的操作而造成的敏感信息的泄漏问题。

## 自主访问控制的类型（控制许可权手段）

### 层次型（等级型）

系统管理员可修改所有文件的ACL表，文件主可以修改自己文件的ACL表。

层次型的优点是可以通过选择可信的人担任各级权限管理员，

缺点是一个客体可能会有多个主体对它具有控制权，发生问题后存在一个责任问题。

### 属主型

该类型的访问权控制方式是为每一个客体设置拥有者，一般情况下客体的创建者就是该客体的拥有者。

拥有者是唯一可以修改自己客体的ACL表的主体，也可以对其他主体授予或撤消对自己客体的访问操作权。如果主体（用户）被调离他处或死亡，系统需要利用某种特权机制来删除该主体拥有的客体。

### 自由型

客体的拥有者（创建者）可以把对自己客体的许可权转授给其他主体，并且也可以使其他主体拥有这种转授权，而且这种转授能力不受创建者自己的控制。

但由于这种许可权（修改权）可能会被转授给不可信的主体，因此这种对访问权修改的控制方式是很不安全的。

授权于

访问控制表

基于访问者

基于访问对象

基于角色的访问控制技术

RBAC (Role-Based Access Control）

基于任务的访问控制技术

TBAC (Task-Based Access Control）

基于组机制的访问控制技术 (基于偏序关系)

## 强制访问控制 MAC

定义、Bell-La Padual模型、Biba模型P221

### 实现方法

第一是访问控制策略要符合MAC的原则，把这些权利交给全系统权利最高和最受信任的安全管理员。

第二是对系统中的每一个主体与客体都要加安全标记，使它和主体或客体紧密相连而无法分开

在MAC机制下，创建客体是受严格控制的，这样就可以阻止某个进程通过创建共享文件的方式向其他进程传递信息。

用户为某个目的运行的程序，由于他不能修改他自己及其他任何客体的安全属性，因此，即使用户程序中或系统中包含恶意程序（如特洛伊木马），也很难获取与用户程序无关的客体的敏感信息。

无法防范用户自己用非计算机手段将自己有权阅读的文件泄漏出去

在高安全级（B级及以上）的计算机系统中同时实现MAC机制与DAC机制，是在DAC机制的基础上增加更强的访问控制以达到强制访问控制的目的。

主体必须首先通过DAC和MAC的控制检查后，得到允许后方能访问某个客体。客体受到了双重保护，DAC可以防范未经允许的用户对客体的攻击，而MAC不允许随意修改主体、客体的安全属性，提供了一个不可逾越的保护层，因而又可以防范任意用户随意滥用DAC机制转授访问权。

### 支持MAC的措施

1、防止恶意程序从外部进入系统。恶意程序从外部进入系统有两种渠道

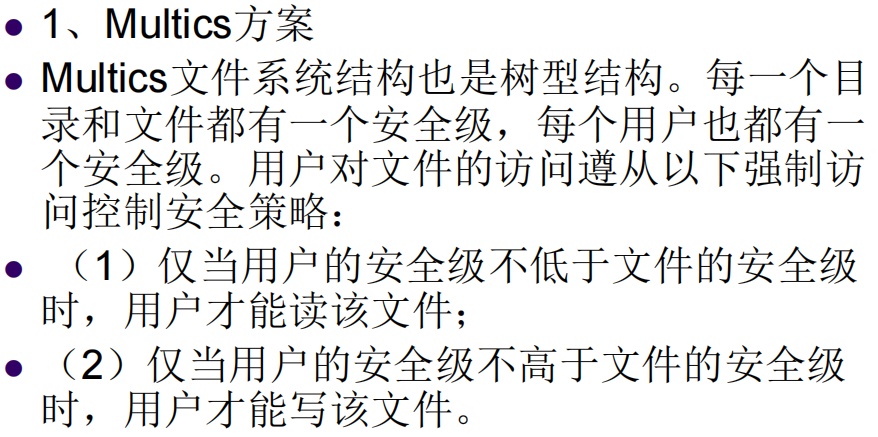
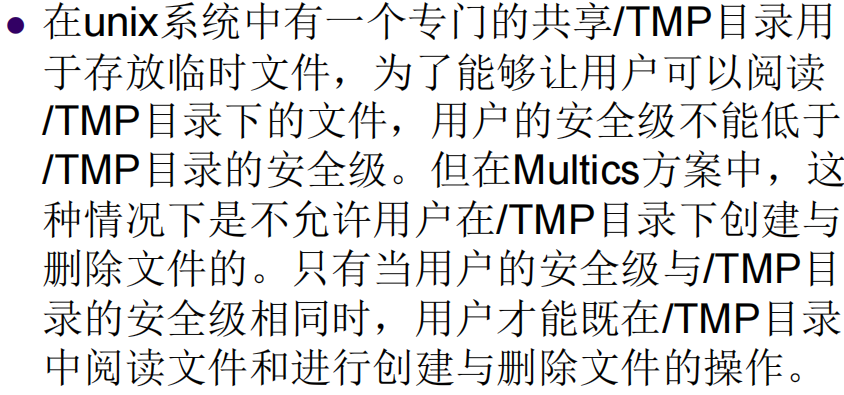
一是通过软盘、光盘或网络下载等方式，由用户自己“主动地”把未被认证是“纯净”的软件装入到系统中。

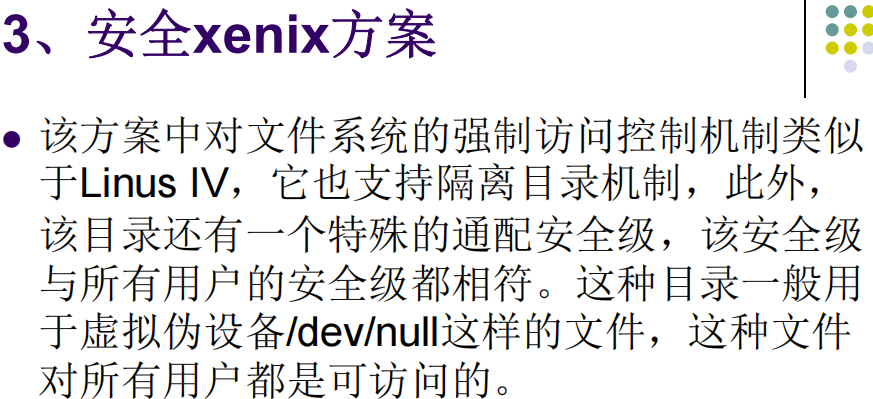
二是利用系统存在的漏洞，通过网络攻击等手段把木马类程序装入系统。

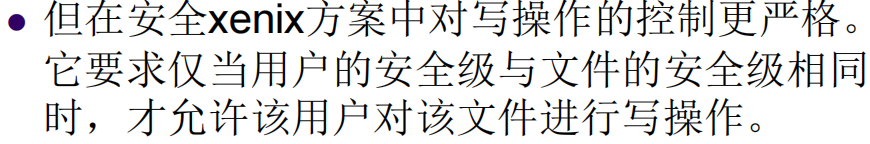
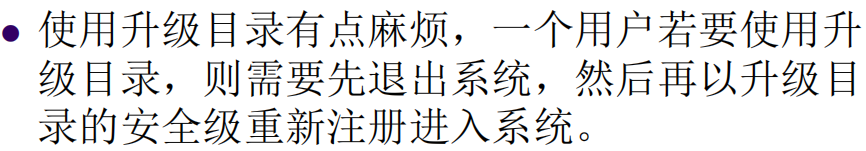
用户对自己还要加强过程性控制，防止使用别人的木马程序或让木马进入自己的控制目录

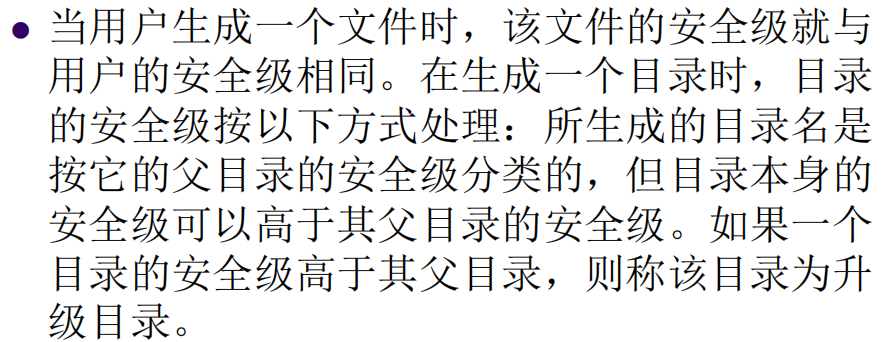
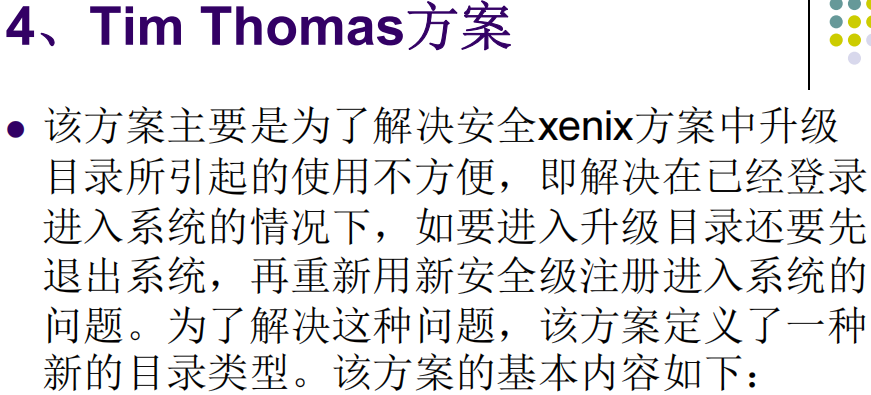
2、消除利用系统自身的支持而产生木马的可能性。 在MAC机制下，由于系统中有很强的访问控制措施，外来的木马很难顺利工作与达到目的。但是，如果内部某个有不良意图的合法用户利用自己的权限在系统编程工具的支持下，编写藏有木马的程序，并使它在系统中合法的运行，这种情况下木马很难防范

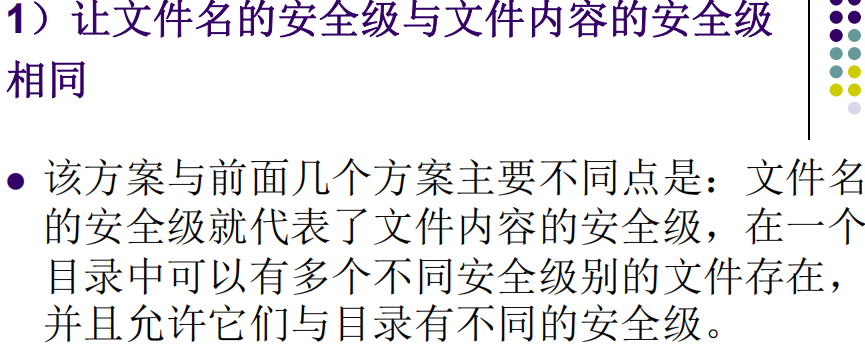
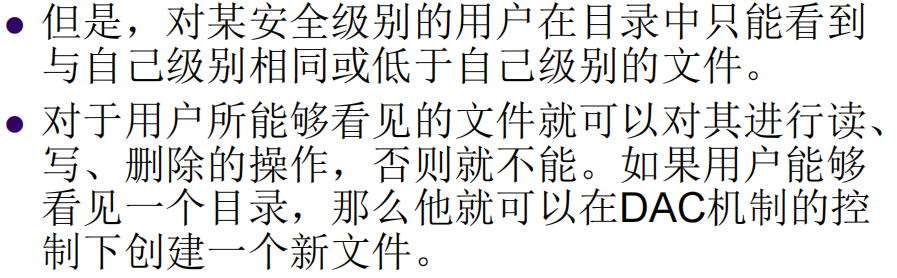
### 实现MAC的实例

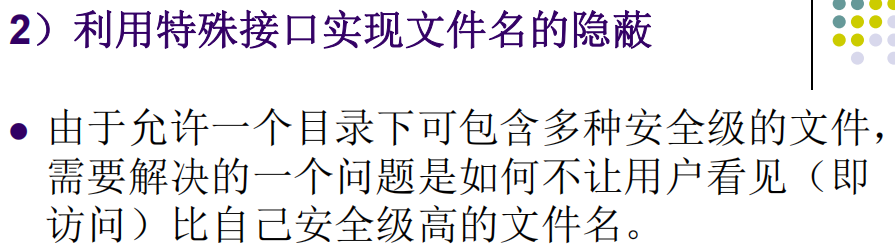
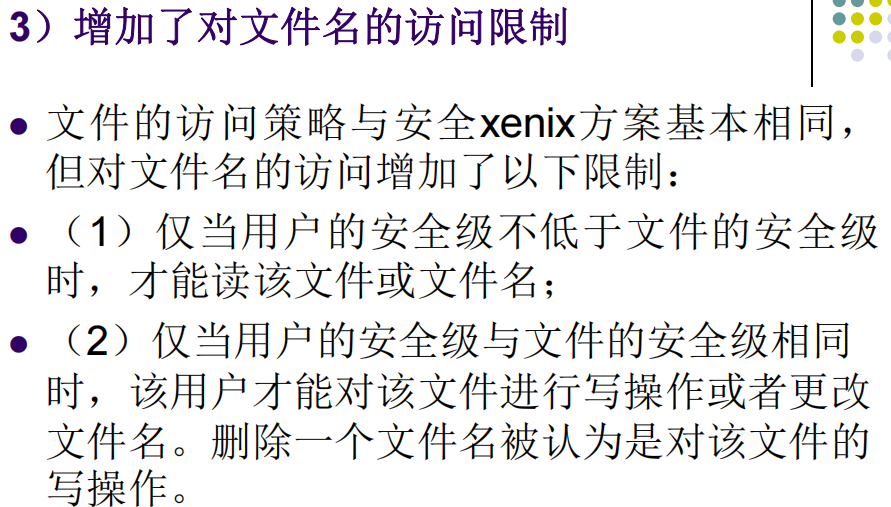
 

## 新型访问控制技术

基于角色的访问控制技术RBAC以及与DAC、MAC对比p228

## 木马窃取敏感文件的方法

一是通过修改敏感文件的安全属性（如敏感级别、访问权等）来获取敏感信息。

另一种方法是躲在用户程序中的木马利用合法用户读敏感文件的机会，把所访问文件的内容拷贝到入侵者的临时目录下

## DAC和MAC的区别以及结合？

DAC的优点

方便、实用

可由用户自由定制

可扩展性强

缺点

管理分散、用户关系不清

权限易被滥用

MAC一般与自主访问控制结合使用，并实施一些附加的、更强的访问限制

缺点

限制访问控制的灵活性

过程控制

**DAC与MAC结合**

自主访问控制防范其他用户对自己客体的攻击

强制访问控制防止其他用户偶然或者滥用自主访问控制权利

减少特洛伊木马攻击成功的可能性

限制访问控制的灵活性

过程控制：警告用户不要运行系统目录以外的任何程序

## 身份认证的选择标准

可行性

安全性

认证粒度

真实性

兼容性

## 面向过程的访问控制

面向过程的访问控制是指在主体访问客体的过程中对主体的访问操作进行监视与限制。

该访问控制过程实际上是为被保护的客体建立一个保护层，它对外提供一个可信赖的接口，所有对客体的访问都必须通过这个接口才能完成

面向对象技术与抽象数据类型都要求数据隐蔽功能，即数据隐藏在模块内部，这些数据中有的局部于模块内，外界永远不得访问；有的虽然允许外界访问，但必须通过模块接口才能完成。面向过程的保护机制可以实现这种信息隐蔽要求，但要付出执行效率的代价

有的系统中实现的保护子系统机制就是面向过程的访问控制的典型例子

在子系统中的数据文件是受保护的对象，子系统中的过程是用来管理受保护对象的，并按用户要求实施对这些客体的访问控制。外部进程只能通过调用管理程序对子系统内部的客体进行访问操作

可以利用多个保护子系统来完成某项作业，这就有可能出现这些子系统互相调用对方内部过程的情况。为了防止调用了不可信程序而对子系统内部的客体造成破坏，各个子系统内部都应该按互相猜疑策略进行防范。

## 实现DAC的实例

VAX/VMS曾经是非常典型的小型机操作系统，其中采用的支持DAC的文件系统的保护机制是一种很有效的文件安全的保护方法，在许多操作系统中广泛地应用。

VAX/VMS提供了两种基本的文件保护机制，一是基于用户识别码（UIC——User Identification Code）的标准保护机制，简称UIC保护机制；另一种是基于访问控制表ACL的保护机制。在VAX/VMS系统中，文件用户被划分为系统（system）类、拥有者（ownner）类、用户组（group）类和所有（world）类等四类，world类包括了前三类的用户

系统在用户的授权文件UAF（User Authorize File）中为每一个用户定义一个UIC，UIC由组号与成员号组成，其形式为[group, member]。对系统中的每一个客体也定义UIC和一个保护码，客体的UIC与其拥有者的UIC相同，保护码则表明允许哪些用户类对客体进行访问，以及进行何类访问

在VAX/VMS系统中按以下步骤控制用户对文件的每一次访问的：

1、首先检查文件是否带有访问控制表ACL，如果有，系统就按ACL表控制用户对该文件的访问。

2、如果ACL表中没有直接允许或拒绝该用户对该客体进行访问，那么系统就转而根据UIC机制来判决是否允许本次用户的访问。特别是如果ACL表直接拒绝了用户的访问请求，那么系统就仅根据UIC机制中的system与owner域来进一步判断是否允许用户的本次访问。

3、如果被访问的客体没有ACL表，系统就直接基于UIC的保护机制判决是否允许用户本次的访问。

4、对于拥有某些系统特权的用户可以不受ACL与UIC机制的限制而获得对客体的访问权。这些特权包括GRPPRV（组特权）、SYSPRV（系统特权）、READALL（读特权）以及BYPASS（全权）等特权。

## 口令

### 口令选择原则

易记

难以被别人猜中或发现

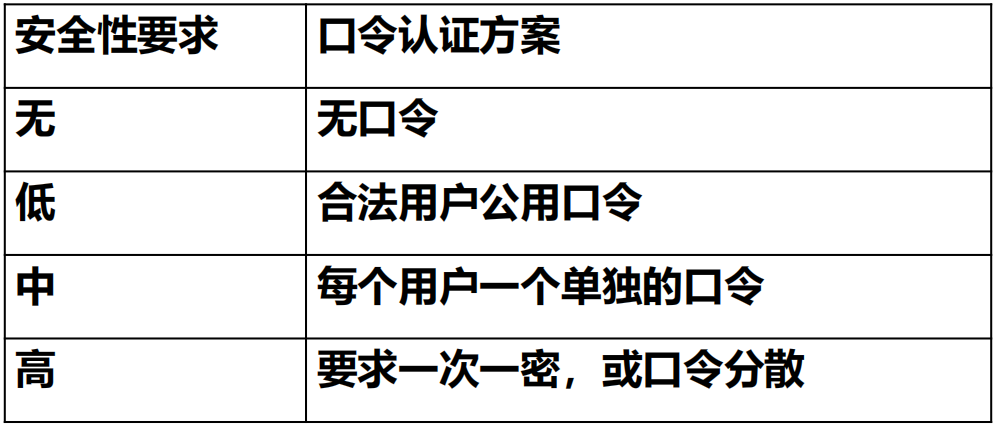
抗分析能力强

需要考虑的方面

选择方法、使用期限、字符长度、分配和管理以及在计算机系统内的保护

### 口令等级

系统中不存储口令的原文



### 认证方式

单向认证

双向认证

询问认证

受理的用户可利用他所知道、而别人不太知道的一些信息向申请用户提问一系列不大相关的问题

### 口令机制缺陷

系统不知谁访问了客体。对客体访问的口令是手工分发的，不需要系统参与

安全性脆弱。需要把该客体的口令写在程序中，这样很容易造成口令的泄露。

使用不方便。每个用户需要记忆许多需要访问的客体的口令，很不友好。

管理麻烦。撤消某用户对某客体的访问权，只能改变该客体的口令，必须通知新口令给其他用户。

### 一次性口令（OTP One Time Password）

在认证口令中插入不确定的因素。不确定因素通常以明文方式传输，使得客户端和服务端都能够识别。用散列函数将口令和不确定因素的结合进行散列，其结果作为认证口令

### 一次性口令的生成方式

Token Card（硬件卡）

Soft Token（软件）

IC卡

### 不确定因子选择方式

口令序列(S/KEY)

挑战/回答(CRYPTOCard)

时间同步(SecureID)

事件同步(Safe Word)

## 物理隔离

网络安全隔离卡p212

### 物理隔离优点

安全级别高，保障强

易于在现有涉密网上安装

### 物理隔离的缺点

资源消耗大

没有实时功能

缺乏管理

认证、访问控制、审计、取证

妨碍应用

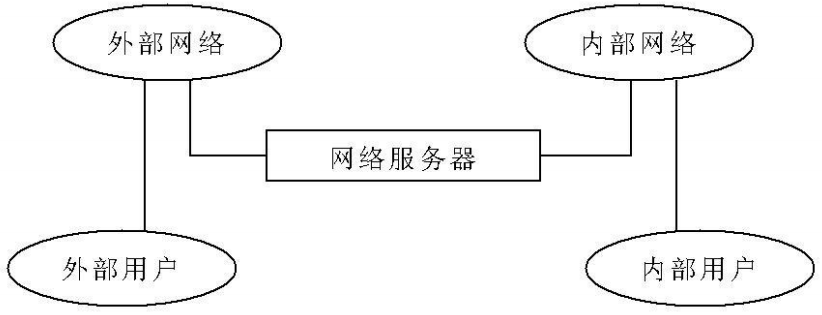
### 逻辑隔离

逻辑隔离技术是一种将内外网络从物理上断开，但保持逻辑连接的网络安全技术

任何时候内外网络都不存在连通的物理连接，同时原有的传输协议必须被中断

逻辑连接指能进行适度的数据交换

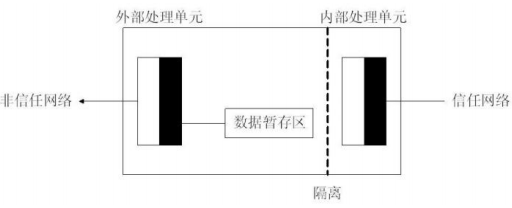
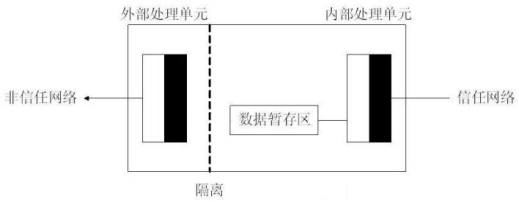
### 服务器端物理隔离



### 隔离网闸

p216

工作原理

### 网闸工作逻辑图



### 网闸能够交换的数据类型

文件交换模块：实现不同安全等级网络间文件的安全交换

数据库同步模块：通过灵活的同步机制，保证安全等级不同的网络中的数据库系统实现数据同步更新

邮件交换模块：保证在内外网隔离的环境下实现安全的邮件收发

安全浏览模块：保证在内外网隔离的环境下，内网用户安全浏览外网资源

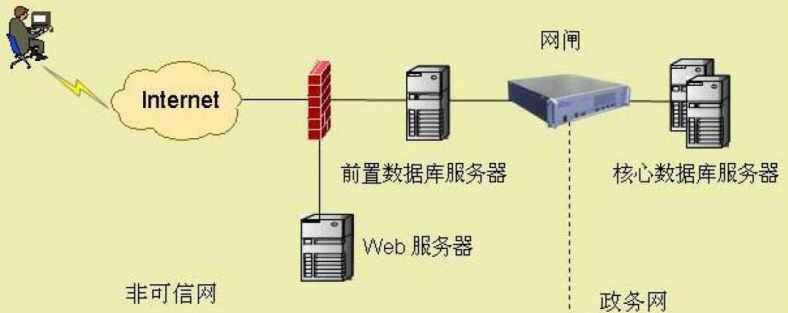
通用模块：保证内外网隔离的同时实现FTP、DNS、TNS 等协议及其他

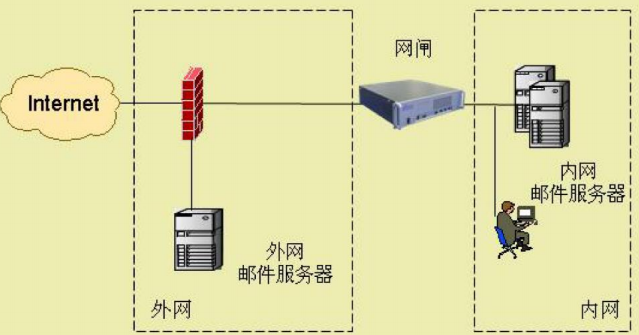
通用TCP/IP 协议的定制交换。

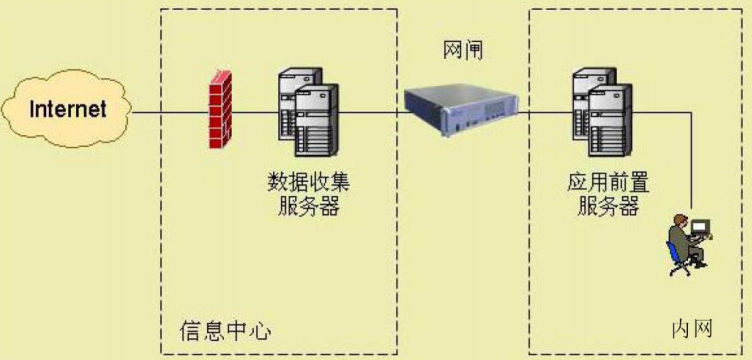
其它定制用户专有应用模块。

### 网闸应用例子

数据库同步、邮件同步、文件同步







## 网络隔离卡和网卡的区别

网路隔离卡见课本

网卡是工作在链路层的网络组件，是局域网中连接计算机和传输介质的接口，不仅能实现与局域网传输介质之间的物理连接和电信号匹配，还涉及帧的发送与接收、帧的封装与拆封、介质访问控制、数据的编码与解码以及数据缓存的功能等。

专网专用，连系统也必须各是各的，内网是内网系统、外网是外网系统，不能一个系统可以同时连接两个网络（双网卡既是一个系统可以同时连接两个网络）.

## 网闸与防火墙对比

网闸

交换模块的安全不依赖于任何操作系统

分析应用层数据

防火墙

软件的运行依赖于操作系统的安全

一般只关心地址和端口

## 入网认证

身份认证、身份认证依据、身份认证评价标准、入网认证步骤p208

## S/Key一次性口令认证系统

RFC2289 . Alice->Victor

1. Alice向出示Victor自己的ID，请求认证。Victor发给Alice一个挑战。挑战（challenge）由一个种子（seed）值和一个迭代值（iteration）组成。

2. Alice根据挑战和自己口令生成一个通行证

3. Victor收到通行证后，把通行证再散列一次，其值同上次Alice成功登录的口令（或初始口令）比较

一定注意: 绝不要让您的计数器减少为0

**分析**

当Victor向Alice发出挑战时，Eve窃取下来，假如挑战中的迭代值为m，Eve将迭代值修改为一个较小的整数k，k<m，把修改后的挑战发给Alice。

Alice根据修改的挑战计算出一个通行证OTP，向Victor提交验证。（Victor验证不通过。）

Eve窃取出OPT，将OPT散列m-k次，生成通行证，用于冒充Alice进行验证。

Victor通过验证。Eve冒充成功。

Victor下一次冒充将窃取的OPT散列m-k-1次，生成通行证也能通过验证。这样可以冒充m-k次

## Kerberos相关

补充资料

# 第十章 防火墙

定义、经典安全模型、规则、防火墙策略、匹配条件p230

堡垒机、双重宿主主机、周边网络p239

## 防火墙的基本原理

是对内部网络与外部网络之间的信息流传递进行控制

控制的功能是通过在防火墙中预先设定一定的安全规则（也称为安全策略）实现的

## 防火墙分类

p234、两种实现技术的比较p237、

个人防火墙

禁止Internet文件共享、隐藏端口、过滤IP信息流、控制Internet应用程序、警告和日志、漏洞检查、保护主机

网络防火墙

对网络数据流进行分析，并按照规则进行过滤

保护内网、外网

软件防火墙

一般基于某个操作系统平台开发，直接在计算机上进行软件的安装和配置。

硬件防火墙

把“软件防火墙”嵌入在硬件中，把“防火墙程序”加入到芯片里面，由硬件执行这些功能，从而减少计算机或服务器的CPU负担。

### 体系结构

P239

## 构建

* 选择体系 P251 + PPT LMPPT
* 安装堡垒机 P254 + PPT
* 数据包过滤规则 P256 + PPT

**看书上例子**

注意路由器的向内/向外，内部接口和外部接口的向内都是向路由器本身。

# 第十一章 入侵检测

## 概述

P287 + PPT 定义:动态网络安全技术

~~为什么具有实时性和自适应性？结合后面内容详细解释~~

具体问题可以对照P288图和PPT的原理图答

## 分类和评估

P290 每种分类方式都需要了解

LMPPT有更详细点

集中式/分布式看LMPPT

IDS评估标准 P293 + PPT 误报&漏报 其他扩充答一下就行

* 准确性，指系统正确识别入侵行为的能力；
* 可靠性，指系统的容错能力和可持续运行能力；
* 可用性，指系统开销大小，对网络性能影响的大小；
* 可测试，指系统能够通过模拟攻击进行检测测试；
* 适应性，指系统易于开发和扩展；
* 实时性，指系统能够及早发现入侵企图；
* 安全性，指系统可确保自身的安全。

# 第十二章 VPN

## 概述

P305 + PPT 定义和优点（按照PPT示例答即可）

P306 组成和基本结构

~~VPN为什么需要两个服务器，VPN工作原理？整理一下~~

## 协议与实现

### 身份验证

P309 + PPT验证过程了解即可

### 隧道协议

P310 +PPT 隧道的定义：针对VPN协议重新**打包**

重点了解PPTP和L2TP区别 P315 + PPT + P319的表格

PPTP和L2TP都属于第二层隧道协议，使用PPP协议对数据进行封装，然后添加附加包头用于数据在互联网络上的传输。

* **PPTP** 点对点隧道协议
* **L2TP** 第二层隧道协议

和PPTP本质区别就是可以从MAC层封装，从数据链路层开始数据包就是安全的。

* **IPSec** IP安全协议

定义和优势看PPT就行

PPT 封装方法 常用模式Tunnel Mode1

加密过程细节都不用关注。

### 数据加密

# 盲猜一手

## 简答题

### 攻击步骤

### 交换式以太网和共享式以太网

* *简述以太网网络监听？*工作原理、扫描方式等
* *如何防范网络监听？*ARP欺骗例子及防御方法

### DoS

* *外部用户针对网络连接发动拒绝服务攻击有哪几种模式？*

DoS分类 P57

### 缓冲区溢出

* *缓冲区溢出的一般目标？*

提权；

* *要让程序跳转到安排好的地址空间执行 一般有哪些方法？*

### 病毒 蠕虫 木马的区别

### 访问控制

* DAC和MAC的区别以及如何结合

### VPN

* PPTP和L2TP区别

## 综合题

### DDoS实例

不知道会怎么出，可能会结合攻击步骤知识点，写一个完整的攻击思路？

### 缓冲区溢出

方法和步骤，会偏技术

* *Printf()有哪些漏洞？*举例说明

### 欺骗攻击

* 给图解释DNS欺骗过程
* ***IP欺骗，他讲了很久，感觉很有可能考***

### 防火墙

重点在 数据包过滤规则 给拓扑图写过滤规则

但其他流程答全即可

### \*入侵检测

HIDS&NIDS区别

# 补充

**二、信息收集的过程，结合经验和理解**

1. 从一些社会信息入手：通过社工的方式，查找到目标的一些信息。可以参考CYPPT，上面有很多信息收集的内容。

2. 找到网络地址范围：比如通过社交软件获得目标所在地，获得大致ip范围。通过扫描技术，地址扫描

3. 找到关键的机器地址：通过扫描技术，地址扫描

4. 找到开放端口和入口点：端口扫描技术（以上都在课本p12）

5. 找到系统的制造商和版本：体系结构探测（课本p13）

**三、网络监听—交换式以太网**

**1.其中的安全问题**

1.交换机缓冲区溢出攻击

　　 交换机大多使用存贮转发技术，工作时维护着一张MAC地址与端口的映射表，这个表中记 录着交换机每个端口绑定的MAC地址。他的工作原理是对某一段数据包进行分析判别寻址， 并进行转发，在发出前均存贮在交换机的缓冲区内。但是，交换机缓冲区是有限的。如用大 量无效IP包，包含错误MAC地址的数据帧对交换机进行攻击。该交换机将接收到大量的不符 合分装原则的包，造成交换机处理器工作繁忙，从而导致数据包来不及转发，进而导致缓 冲区溢出产生丢包现象。这时交换机就会退回到HUB的广播方式，向所有的端口发送数据包 。这样，监听就变得非常容易了。

**2. ARP协议和欺骗**

**1. ARP欺骗的原理**

在每台主机都有一个ARP缓存表，缓存表中记录了IP地址与MAC地址的对应关系，而局域网数据传输依靠的是MAC地址。

假设主机 A 192.168.1.2,B 192.168.1.3,C 192.168.1.4; 网关 G 192.168.1.1; 在同一局域网，主机A和B通过网关G相互通信，就好比A和B两个人写信，由邮递员G送信，C永远都不会知道A和B之间说了些什么话。但是并不是想象中的那么安全，在ARP缓存表机制存在一个缺陷，就是当请求主机收到ARP应答包后，不会去验证自己是否向对方主机发送过ARP请求包，就直接把这个返回包中的IP地址与MAC地址的对应关系保存进ARP缓存表中，如果原有相同IP对应关系，原有的则会被替换。

这样C就有了偷听A和B的谈话的可能，继续思考上面的例子：

C假扮邮递员，首先要告诉A说：“我就是邮递员” （C主机向A发送构造好的返回包，源IP为G 192.168.1.1，源MAC为C自己的MAC地址），愚蠢的A很轻易的相信了，直接把“C是邮递员”这个信息记在了脑子里；

C再假扮A，告诉邮递员：“我就是A” （C向网关G发送构造好的返回包，源IP为A 192.168.1.2，源MAC地址为自己的MAC地址），智商捉急的邮递员想都没想就相信了，以后就把B的来信送给了C，C当然就可以知道A和B之间聊了些什么

上面ABCG的故事就是ARP双向欺骗的原理了

ARP单向欺骗就更好理解了，C只向A发送一个返回包，告诉A：G 192.168.1.1 的MAC地址为 5c-63-bf-79-1d-fa（一个错误的mac地址），A把这个信息记录在了缓存表中，而G的缓存表不变，也就是说，A把数据包给了C，而G的包还是给A，这样就是ARP单向欺骗了。

**2. 防范ARP欺骗的方法**

建立正确的 ARP 映射关系、检测并过滤伪造的 ARP 报文，保证经过其转发的 ARP报文正确合法；

抑制短时间内大量 ARP报文的冲击。

在无线网络中，防御ARP欺骗的有效方法是网络管理员分别在主机或接入点上和路由器或控制器上对**IP和MAC地址进行静态映射绑定**。（ARP欺骗是通过重复应答实现的，那么只需要在主机添加一条静态的ARP映射，这样就不需要询问网关MAC地址了；对于网关欺骗还需要在网关中也添加一条到主机的静态ARP映射。）

1.主机级被动检测

  当系统接收到来自局域网上的ARP请求时，系统检查该请求发送端的IP地址是否与自己的IP地址相同。如果相同，则说明该网络上另有一台机器与自己具有相同的IP地址。

2.主机级主动检测

  主机定期向所在局域网发送查询自己IP地址的ARP请求报文。如果能够收到另一ARP响应报文，则说明该网络上另有一台机器与自己具有相同的IP地址。

3.服务器级检测

  当服务器收到ARP响应时，为了证实它的真实性，根据反向地址解析协议(RARP)就用从响应报文中给出的 地 MAC址再生成一个RARP请求，它询问这样一个问题：“如果你是这个MAC地址的拥有者，请回答你的IP地址”。这样就会查询到这个MAC地址对应的IP地址，比较这两个IP地址，如果不同，则说明对方伪造了ARP响应报文。

4.网络级检测

  配置主机定期向中心管理主机报告其ARP缓存的内容。这样中心管理主机上的程序就会查找出两台主机报告信息的不一致，以及同一台主机前后报告内容的变化。这些情况反映了潜在的安全问题。或者利用网络嗅探工具连续监测网络内主机硬件地址与IP地址对应关系的变化。

**3. 网络监听的防范**

1.网络分段

　　网络分段通常被认为是控制网络广播风暴的一种基本手段，但其实也是防范网络监听的一 项重要措施。将网络划分为不同的网段，其目的是将非法用户与敏感的网络资源相互隔离， 从而防止可能的非法监听。

2.以交换式集线器代替共享式集线器

　　 对局域网的中心交换机进行网络分段后，局域网监听的危险仍然存在。这是因为网络最终 用户的接入往往是通过分支集线器而不是中心交换机，而使用最广泛的分支集线器通常是共享式集线器。这样，当用户与主机进行数据通信时，两台机器之间的数据包(单播包)还是会被同一 台集线器上的其他用户监听。因此，应该以交换式集线器代替共享式集线器，使 单播包仅在两个节点之间传送，从而防止非法监听。

3.使用加密技术

　　数据经过加密后，通过监听仍然可以得到传送的信息,但显示的是乱码。使用加密技术的 缺点是影响数据传输速度以及使用一个弱加密术比较容易被攻破。

4.划分VLAN

　　运用VLAN（虚拟局域网）技术，将以太网通信变为点到点通信，VLAN子网隔离了广播风暴 ，可以防止大部分基于网络监听的侵入，对一些重要部门实施了安全保护。且当某一部门物 理位置发生变化时，只需对交换机进行设置，就可以实现网络的重组，非常方便、快捷，同 时节约了成本。为保证不同职能部门管理的方便性和安全性以及整个网络运行的稳定性，可 采用VLAN技术进行虚拟网络划分。

**四、拒绝服务攻击—ACK Flood攻击**

1.原理：

ACK Flood攻击是在TCP连接建立之后，所有的数据传输TCP报文都是带有ACK标志位的，主机在接收到一个带有ACK标志位的数据包的时候，需要检查该数据包所表示的连接四元组是否存在，如果存在则检查该数据包所表示的状态是否合法，然后再向应用层传递该数据包。如果在检查中发现该数据包不合法，例如该数据包所指向的目的端口在本机并未开放，则主机操作系统协议栈会回应RST包告诉对方此端口不存在。

这里，服务器要做两个动作：查表、回应ACK/RST。这种攻击方式显然没有SYN Flood给服务器带来的冲击大，因此攻击者一定要用大流量ACK小包冲击才会对服务器造成影响。按照我们对TCP协议的理解，随机源IP的ACK小包应该会被Server很快丢弃，因为在服务器的TCP堆栈中没有这些ACK包的状态信息。但是实际上通过测试，发现有一些TCP服务会对ACK Flood比较敏感，比如说JSP Server，在数量并不多的ACK小包的打击下，JSP Server就很难处理正常的连接请求。对于Apache或者IIS来说，10kpps的ACK Flood不构成危胁，但是更高数量的ACK Flood会造成服务器网卡中断频率过高，负载过重而停止响应。可以肯定的是，ACK Flood不但可以危害路由器等网络设备，而且对服务器上的应用有不小的影响。

如果没有开放端口，服务器将直接丢弃，这将会耗费服务器的CPU资源。如果端口开放，服务器回应RST。

2 ACK Flood防护

利用对称性判断来分析出是否有攻击存在。所谓对称型判断，就是收包异常大于发包，因为攻击者通常会采用大量ACK包，并且为了提高攻击速度，一般采用内容基本一致的小包发送。这可以作为判断是否发生ACK Flood的依据，但是目前已知情况来看，很少有单纯使用ACK Flood攻击，都会和其他攻击方法混合使用，因此，很容易产生误判。

一些防火墙应对的方法是：建立一个hash表，用来存放TCP连接“状态”，相对于主机的TCP stack实现来说，状态检查的过程相对简化。例如，不作sequence number的检查，不作包乱序的处理，只是统计一定时间内是否有ACK包在该“连接”(即四元组)上通过，从而“大致”确定该“连接”是否是“活动的”。

（每建立一个监听套接字，就将套接字挂入监听哈希表的某个表项链表中。则在内核收到连接请求的SYN或ACK等包传到TCP层时，要根据请求包的请求连接地址与端口号到哈希表中查找对应的服务器监听套接字是否存在）

**五、DoS攻击的防范方法**

1. 确保服务器的系统文件是最新的版本,并及时更新系统补丁。

要定期扫描现有的网络主节点,清查可能存在的安全漏洞,对新出现的漏洞及时进行清理。骨干节点的计算机因为具有较高的带宽,是黑客利用的最佳位置,因此对这些主机本身加强主机安全是非常重要的。而且连接到网络主节点的都是服务器级别的计算机,所以定期扫描漏洞就变得更加重要了。

1. 关闭不必要的服务

过滤不必要的服务和端口.可以使用Inexpress、Express、Forwarding等工具来过滤不必要的服务和端口,即在路由器上过滤假IP。比如Cisco公司的 CEF(Cisco Express Forwarding)可以针对封包Source IP和Routing Table做比较,并加以过滤。只开放服务端口成为目前很多服务器的流行做法,例如WWW服务器那么只开放80而将其他所有端口关闭或在防火墙上做阻止策略。

1. 限制同时打开的SYN半连接数目,缩短SYN半连接的timeout时间,限制SYN/ICMP流量

用户应在路由器上配置SYN/ICMP的最大流量来限制SYN/ICMP封包所能占有的最高频宽,这样,当出现大量的超过所限定的SYN/ICMP流量时,说明不是正常的网络访问,而是有黑客入侵。早期通过限制SYN/ICMP流量是最好的防范DOS的方法,虽然目前该方法对于Ddos效果不太明显了, 不过仍然能够起到一定的作用。

1. 正确设置防火墙

禁止对主机的非开放服务的访问；限制特定IP地址的访问，（过滤所有RFC1918 IP地址RFC1918 IP地址是内部网的IP地址,像10.0.0.0、192.168.0.0和172.16.0.0,它们不是某个网段的固定的IP地址,而是Internet内部保留的区域性IP地址,应该把它们过滤掉。此方法并不是过滤内部员工的访问,而是将攻击时伪造的大量虚假内部IP过滤,这样也可以减轻Ddos的攻击。）启用防火墙的防DDoS的属性；严格限制对外开放的服务器的向外访问；运行端口映射程序祸端口扫描程序,要认真检查特权端口和非特权端口。

1. 认真检查网络设备和主机/服务器系统的日志。只要日志出现漏洞或是时间变更,那这台机器就可能遭到了攻击。
2. 限制在防火墙外与网络文件共享。这样会给黑客截取系统文件的机会,主机的信息暴露给黑客,无疑是给了对方入侵的机会。
3. 充分利用网络设备保护网络资源

所谓网络设备是指路由器、防火墙等负载均衡设备,它们可将网络有效地保护起来。当网络被攻击时最先死掉的是路由器,但其他机器没有死。死掉的路由器经重 启后会恢复正常,而且启动起来还很快,没有什么损失。若其他服务器死掉,其中的数据会丢失,而且重启服务器又是一个漫长的过程。特别是一个公司使用了负载 均衡设备,这样当一台路由器被攻击死机时,另一台将马上工作。从而最大程度的削减了Ddos的攻击。

1. 用足够的机器承受黑客攻击

如果用户拥有足够的容量和足够的资源给黑客攻击,在它不断访问用户、夺取用户资源之时,自己的能量也在逐渐耗失,或许未等用户被攻死,黑客已无力支招儿了。不过此方法需要投入的资金比较多,平时大多数设备处于空闲状态,和目前中小企业网络实际运行情况不相符。

1. 检查访问者的来源

使用Unicast Reverse Path Forwarding等通过反向路由器查询的方法检查访问者的IP地址是否是真,如果是假的,它将予以屏蔽。许多黑客攻击常采用假IP地址方式迷惑用户, 很难查出它来自何处。因此,利用Unicast Reverse Path Forwarding可减少假IP地址的出现,有助于提高网络安全性。

**六、DoS与DDoS攻击的原理和区别**

**DOS攻击的原理**：首先攻击者向被攻击的服务器发送大量的虚假ip请求，被攻击者在收到请求后返回确认信息，等待攻击者进行确认，（此处需要拥有HTTP协议工作方式和tcp三次握手的基本知识）该过程需要TCP的三次握手，由于攻击者发送的请求信息是虚假的，所以服务器接收不到返回的确认信息，在一段时间内服务器会处与等待状态，而分配给这次请求的资源却被有被释放。当被攻击者等待一定的时间后，会因连接超时而断开，这时攻击者在次发送新的虚假信息请求，这样最终服务器资源被耗尽，直到瘫痪。

**DDOS的原理**，它是在DOS基础上进行的大规模，大 范围的攻击模式，DOS只是单机和单机之间的攻击模式，而DDOS是利用一批受控制的僵尸主 机向一台服务器主机发起的攻击，其攻击的强度和造成的威胁要比DOS严重很多，更具破坏 性。首先DDOS攻击者要寻找僵尸主机，在互联网上寻找一些有后门漏洞的主机，然后入侵系 统安装控制程序，入侵的越多，控制的僵尸主机就越多，攻击源就更多，然后把入侵的主机 分配，一部分充当攻击的主要控制端，一部分充当攻击源，各负其责，在攻击者统一指挥下 对被攻击的服务器发起攻击，由于这个攻击模式是在幕后操作，所以很难被监控系统跟踪， 身份不容易被发现。

**七、反弹技术—DRDoS**

　DRDoS分布反射式拒绝服务攻击这是DDoS攻击的变形,它与DDoS的不同之处就是DrDoS不需要在攻击之前占领大量的“肉鸡”。它的攻击原理和Smurf攻击原理相近,不过DRDoS是可以在广域网上进行的,而Smurf攻击是在局域网进行的。它的作用原理是基于广播地址与回应请求的。一台计算机向另一台计算机发送一些特殊的数据包如ping请求时,会接到它的回应;如果向本网络的广播地址发送请求包,实际上会到达网络上所有的计算机,这时就会得到所有计算机的回应。这些回应是需要被接收的计算机处理的,每处理一个就要占用一份系统资源,如果同时接到网络上所有计算机的回应,接收方的系统是有可能吃不消的,就象遭到了DDoS攻击一样。不过是没有人笨到自己攻击自己,不过这种方法被黑客加以改进就具有很大的威力了。黑客向广播地址发送请求包,所有的计算机得到请求后,却不会把回应发到黑客那里,而是发到被攻击主机。这是因为黑客冒充了被攻击主机。黑客发送请求包所用的软件是可以伪造源地址的,接到伪造数据包的主机会根据源地址把回应发出去,这当然就是被攻击主机的地址。黑客同时还会把发送请求包的时间间隔减小,这样在短时间能发出大量的请求包,使被攻击主机接到从被欺骗计算机那里传来的洪水般的回应,就像遭到了DDoS攻击导致系统崩溃。骇客借助了网络中所有计算机来攻击受害者,而不需要事先去占领这些被欺骗的主机,这就是Smurf攻击。而DRDoS攻击正是这个原理,黑客同样利用特殊的发包工具,首先把伪造了源地址的SYN连接请求包发送到那些被欺骗的计算机上,根据TCP三次握手的规则,这些计算机会向源IP发出SYN+ACK或RST包来响应这个请求。同Smurf攻击一样,黑客所发送的请求包的源IP地址是被攻击主机的地址,这样受欺骗的主机就都会把回应发到被攻击主机处,造成被攻击主机忙于处理这些回应而瘫痪。

**八、DNS欺骗**

**九、访问控制技术—DAC**

主体是用户的身份，客体是资源或者说是文件（切记：一切皆文件）。由客体的属主对自己的客体进行管理，由主体自己决定是否将自己的客体访问权限或部分访问权限授予其他主体，这种控制方式是自主的。也就是说，**在自主访问控制下，用户可以按自己的意愿，有选择地与其他用户共享他的文件**。

DAC系统有两个至关重要的标准：

1.**文件的所有权**：系统中的每个文件（一些特殊文件可能没有，如块设备文件等）都有所有者。在DAC系统中，文件的所有者是创建这个文件的计算机的使用者（或事件，或另一个文件）。那么此文件的自主访问控制权限由它的创建者来决定如何设置和分配；

2.**访问权限**：文件的所有者拥有访问权限，并且可以将访问权限分配给自己及其他用户。

上述两个标准说明：

1.文件的所有权的优先级高于访问权限

1)文件的所有者即便没有任何权限，也可以在为自己分配权限之后获得访问文件的能力。

2)非文件的所有者即便已经获得了访问权限，也可能会被所有者随时收回，从而导致无权访问该文件。

2.权限是文件访问的关键

1)无论是不是文件的所有者，关系到使用者能否访问文件的最直接的因素是其所对应的用户是否获得了可访问该文件的权限

2)使用者被分配的何种权限，就只能以该权限所规定的操作来访问文件，无法越权。

**强制访问控制MAC**

MAC是利用策略将访问控制规则“强加”给访问主体的，即系统强制主体服从访问控制策略。MAC的主要作用对象是所有主体及其所操作的客体（如：进程、文件等）。**MAC为这些主体及其所操作的客体提供安全标记**，这些标记是实施强制访问控制的依据。

**系统通过比较主体和客体的安全标记来判断一个主体是否能够访问其要操作的客体。用户发起的进程无法改变其自身及其它客体的安全标记，利用这样的机制，系统可以比较有效地防止特洛伊木马攻击以及root身份冒用或盗用等安全威胁**。

MAC一般与DAC共同使用，两种访问控制机制的过滤结果将累积，以此来达到更佳的访问控制效果。也就是说，**一个主体只有通过了DAC限制检查与MAC限制检查的双重过滤装置之后，才能真正访问某个客体。一方面，用户可以利用DAC来防范其它用户对那些所有权归属于自己的客体的攻击；另一方面，由于用户不能直接改变MAC属性，所以MAC提供了一个不可逾越的、更强的安全保护层以防止其它用户偶然或故意地滥用DAC**。

**DAC与MAC的对比**

**1. DAC（Discretionary Access Control，自主访问控制）**

① DAC是传统的Linux的访问控制方式，DAC可以对文件、文件夹、共享资源等进行访问控制。

② 在DAC这种模型中，文件客体的所有者（或者管理员）负责管理访问控制。

③ DAC使用了ACL（Access Control List，访问控制列表）来给非管理者用户提供不同的权限，而root用户对文件系统有完全自由的控制权。

**2. MAC（Mandatory Access Control，强制访问控制）**

① SELinux在内核中使用MAC检查操作是否允许。

② 在MAC这种模型中，系统管理员管理负责访问控制，用户不能直接改变强制访问控制属性。

③ MAC可以定义所有的进程（称为主体）对系统的其他部分（文件、设备、socket、端口和其它进程等，称为客体）进行操作的权限或许可。

**3. DAC和MAC的其它区别**

① DAC的主体是真实有效的用户和组ID，MAC的主体是安全上下文，两者的UID是各自独立的。

② DAC的访问控制模式是rwxrwxrwx，MAC的访问控制模式是user:role:type。

**十一、网闸与防火墙的区别**

网闸与防火墙的区别是：1、防火墙首先保证网络连通性，然后考虑安全问题，网闸在保证安全的基础上进行数据交换；2、防火墙是单主机架构，网闸是双主机架构；3、网闸相对于防火墙同步类功能安全性更高。

**1、应用场景区别**

防火墙：防火墙首先要保证网络的连通性，其次才是安全问题；

网闸：网闸是保证安全的基础上进行数据交换。网闸是两个网络已经存在，现在两个网络不得不互联，互联就要保证安全，网闸是现在唯一最安全的网络边界安全隔离的产品，只有网闸这种产品才能解决这个问题，所以必须用网闸。

**2、硬件区别**

防火墙是单主机架构，早期使用包过滤的技术，网闸是双主机2+1架构，通过私有的协议摆渡的方式进行数据交换，基于会话的检测机制，由于网闸是双主机结构，即使外网端被攻破，由于内部使用私有协议互通，没办法攻击到内网，防火墙是单主机结构，如果被攻击了，就会导致内网完全暴露给别人。

**3、功能区别**

网闸主要包含两大类功能，访问类功能和同步类功能，访问类功能类似于防火墙，网闸相对于防火墙安全性更高的是同步类功能。



**十二、防火墙—代理技术**

**1.代理的执行流程（两种情况）**

1.代理服务器监听内联网络中主机的请求，当请求到达代理服务器后，代理服务器根据安全策略对数据包的首部和数据部分信息进行检查，检查之后将内联网络中的主机源地址改为自己的地址之后将这个数据包发给外部的目的主机，这样外部的主机接到的信息是来自于代理服务器而不是源主机。目的主机应答的数据包也将发送到代理服务器，经过代理服务器检查允许通过后，代理服务器将数据包的目的地址改为内联主机的地址，之后发往目的主机。类似于在内联网络和外联网络之间设置一个中转，外联网络并不清楚内联网络的拓补结构。

2.内部的主机只接受代理服务器的信息而不接收任何外部地址主机发来的信息，外部主机只能讲信息发往代理服务器，之后由代理服务器将信息转发给内联网路中的主机。代理服务器负责监控整个通信过程来保证通信过程的安全性。

**十、入侵检测系统—评价指标**

**1. 测试评估IDS性能的标准**

根据Porras等的研究，给出了评价IDS性能的三个因素：

1. **准确性**：指IDS从各种行为中正确地识别入侵的能力，当一个IDS的检测不准确时，就有可能把系统中的合法活动当作入侵行为并标识为异常（虚警现象）。
2. **处理性能**：指一个IDS处理数据源数据的速度。显然，当IDS的处理性能较差时，它就不可能实现实时的IDS，并有可能成为整个系统的瓶颈，进而严重影响整个系统的性能。
3. **完备性**：指IDS能够检测出所有攻击行为的能力。如果存在一个攻击行为，无法被IDS检测出来，那么该JDS就不具有检测完备性。也就是说，它把对系统的入侵活动当作正常行为（漏报现象）。由于在一般情况下，攻击类型、攻击手段的变化很快，我们很难得到关于攻击行为的所有知识，所以关于IDS的检测完备性的评估相对比较困难。

在此基础上，Debar等又增加了两个性能评价测度：

1. **容错性**：由于IDS是检测入侵的重要手段／所以它也就成为很多入侵者攻击的首选目标。IDS自身必须能够抵御对它自身的攻击，特别是拒绝服务（Denial-of-Service）攻击。由于大多数的IDS是运行在极易遭受攻击的操作系统和硬件平台上，这就使得系统的容错性变得特别重要，在测试评估IDS时必须考虑这一点。
2. **及时性**：及时性要求IDS必须尽快地分析数据并把分析结果传播出去，以使系统安全管理者能够在入侵攻击尚未造成更大危害以前做出反应，阻止入侵者进一步的破坏活动，和上面的处理性能因素相比，及时性的要求更高。它不仅要求IDS的处理速度要尽可能地快，而且要求传播、反应检测结果信息的时间尽可能少。

按照数据分析发生的时间不同，IDS可以分为**脱机分析和联机分析**

**脱机分析：**就是在行为发生后，对产生的数据进行分析，而不是在行为发生的同时进行分析。如对日志的审核、对系统文件的完整性检查都属于这种。一般而言，脱机分析也会间隔很长时间，脱机只是相对于联机而言的。

**联机分析**：就是在数据产生或者发生改变的同时对其进行检查，以发现攻击行为。这种方式一般用于对网络数据的实时分析，对系统资源要求比较高。

**HIDS和NIDS的对比分析：**（罗敏PPT加对比分析）

HIDS和NIDS的区别如下：

**1. HIDS（Host-based Intrusion Detection System，基于主机的入侵检测系统）**

① HIDS将代理安装在受保护的系统中，它要求与操作系统内核和服务紧密捆绑在一起，监控各种系统事件，如对内核或API的调用，以此来进行防御。

② HIDS还可以监测特定的系统文件和可执行文件调用，以及Windows NT下的安全记录和Unix环境下的系统记录。

③ HIDS能对检测的入侵行为、事件给予积极的反应，比如断开连接、封掉用户账号、杀死进程、提交警报等。

④ HIDS技能要求非常高，要求开发HIDS的企业对有关的操作系统非常了解，而且安装在主机上的代理必须非常可靠，系统占用小，自身安全性要好，否则将会对系统产生负面影响。

⑤ HIDS关注的是到达主机的各种安全威胁，并不关注网络的安全。

⑥ HIDS由于采取的是对事件和系统调用的监控，衡量它的技术指标非常少。

**2. NIDS（Network Intrusion Detection System，网络入侵检测系统）**

① NIDS最大的特点在于不需要改变服务器等主机的配置，它不需在业务系统的主机中安装额外的软件。

② NIDS不是系统中的关键路径，即使发生故障也不会影响正常业务的运行。

③ NIDS是以网络包作为分析数据源。它通常运用一个工作在混杂模式下的网卡来实时监视并分析通过网络的数据流。

④ NIDS的分析模块通常运用模式匹配、统计分析等技能来识别攻击行为，一旦检测到了攻击行为，NIDS的响应模块就作出适当的响应，比如报警、切断有关用户的网络连接等。

⑤ NIDS收集的是网络中的动态流量信息，因此，攻击特征库数目多少以及数据处理能力，就决定了NIDS识别入侵行为的能力。

⑥ 部署一个NIDS，比HIDS的风险与成本相对较低。

⑦ NIDS采取的基本上都是模式匹配的形式，所以衡量NIDS的技术指标可以数量化。

⑧ 在NIDS的应用过程中，最大的敌人就是误警和漏警。漏警不影响应用环境的可用性，只是用户的投资失误；而误警则有可能导致警报异常、网络紊乱，甚至应用的瘫痪。

**十三、VPN—工作原理**

1、通常情况下，VPN网关采取双网卡结构，外网卡使用公网IP接入Internet。

　　2、网络一（假定为公网internet）的终端A访问网络二（假定为公司内网）的终端B，其发出的访问数据包的目标地址为终端B的内部IP地址。

　　3、网络一的VPN网关在接收到终端A发出的访问数据包时对其目标地址进行检查，如果目标地址属于网络二的地址，则将该数据包进行封装，封装的方式根据所采用的VPN技术不同而不同，同时VPN网关会构造一个新VPN数据包，并将封装后的原数据包作为VPN数据包的负载，VPN数据包的目标地址为网络二的VPN网关的外部地址。

　　4、网络一的VPN网关将VPN数据包发送到Internet，由于VPN数据包的目标地址是网络二的VPN网关的外部地址，所以该数据包将被Internet中的路由正确地发送到网络二的VPN网关。

　　5、网络二的VPN网关对接收到的数据包进行检查，如果发现该数据包是从网络一的VPN网关发出的，即可判定该数据包为VPN数据包，并对该数据包进行解包处理。解包的过程主要是先将VPN数据包的包头剥离，再将数据包反向处理还原成原始的数据包。

　　6、网络二的VPN网关将还原后的原始数据包发送至目标终端B，由于原始数据包的目标地址是终端B的IP，所以该数据包能够被正确地发送到终端B。在终端B看来，它收到的数据包就和从终端A直接发过来的一样。

　　7、从终端B返回终端A的数据包处理过程和上述过程一样，这样两个网络内的终端就可以相互通讯了。

　　通过上述说明可以发现，在VPN网关对数据包进行处理时，有两个参数对于VPN通讯十分重要：原始数据包的目标地址（VPN目标地址）和远程VPN网关地址。根据VPN目标地址，VPN网关能够判断对哪些数据包进行VPN处理，对于不需要处理的数据包通常情况下可直接转发到上级路由；远程VPN网关地址则指定了处理后的VPN数据包发送的目标地址，即VPN隧道的另一端VPN网关地址。由于网络通讯是双向的，在进行VPN通讯时，隧道两端的VPN网关都必须知道VPN目标地址和与此对应的远端VPN网关地址。

**十四、PPTP和L2TP**

**L2TP/IPSec 和 PPTP 在以下方面类似：**

它们提供了一种逻辑传输机制来发送 PPP 有效载荷。

它们提供隧道或封装，以便基于任何协议的 PPP 有效载荷可以通过 IP 网络发送。

它们依靠 PPP 连接过程来执行用户认证和协议配置。

**L2TP/IPSec 和 PPTP 在以下方面有所不同：**

使用 PPTP，数据加密在 PPP 连接过程（因此，PPP 身份验证）完成后开始。使用 L2TP/IPSec，数据加密在 PPP 连接过程之前开始。

PPTP 连接使用 MPPE，这是一种基于 Rivest-Shamir-Aldeman RC-4 加密算法并提供 40、56 或 128 位加密密钥的流密码。流密码将数据加密为比特流。L2TP/IPSec 连接使用数据加密标准 (DES)，它是一种分组密码，它使用 56 位的 DES 密钥或三个 3-DES 的 56 位密钥。块密码以离散块（64 位块，在 DES 的情况下）中加密数据。

PPTP 连接仅需要通过基于 PPP 的身份验证协议进行用户级身份验证。L2TP/IPSec 连接需要相同的用户级身份验证，此外还需要通过计算机证书进行计算机级身份验证。

（主要是PPTP和L2TP的工作原理LMPPT + 课本P315）

**如何对抗Teardrop攻击**

Teardrop攻击是攻击者A给受害者B发送一些分片IP报文，并且故意将“13位分片偏移”字段设置成错误的值(既可与上一分片数据重叠，也可错开)，B在组合这种含有重叠偏移的伪造分片报文时，某些操作系统收到含有重叠偏移的伪造分片数据包时将会出现系统崩溃、重启等现象。利用UDP包重组时重叠偏移（假设数据包中第二片IP包的偏移量小于第一片结束的位移，而且算上第二片IP包的Data，也未超过第一片的尾部，这就是重叠现象。）的漏洞对系统主机发动拒绝服务攻击，最终导致主机菪掉；对于Windows系统会导致蓝屏死机，并显示STOP 0x0000000A错误。

Teardrop攻击防御方法：网络安全设备将接收到的分片报文先放入缓存中，并根据源IP地址和目的IP地址对报文进行分组，源IP地址和目的IP地址均相同的报文归入同一组，然后对每组IP报文的相关分片信息进行检查，丢弃分片信息存在错误的报文。为了防止缓存益处，当缓存快要存满是，直接丢弃后续分片报文。

反攻击方法：添加系统补丁程序，丢弃收到的病态分片数据包并对这种攻击进行审计。

**如何发现自己正在受到消耗网络资源的DoS攻击（可参考课本p75）**

1、服务器主机上有大量等待的TCP连接；

2、检查网站后台服务器发现大量无用的数据包；

3、一段时期中IP请求异常且源地址虚假；

4、网络流量出现异常变化突然暴涨；

5、当发现Ping超时或丢包严重时，注意连接错误。假如遇到无法访问网站这种情况，并看到类似于“无法访问站点”之类的错误，且无法访问其他Internet服务，则可能是DoS攻击带来的影响。

6、查看自己的邮箱里是否突然收到大量的垃圾邮件。

具体地，可使用netstat命令：

netstat -n -p | grep SYN\_REC | sort -u

列出所有连接过的IP地址。

netstat -ntu | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -n

使用netstat命令计算每个主机连接到本机的连接数。

netstat -na

该命令将显示所有活动的网络连接。

netstat -n -p | grep SYN\_REC | awk '{print $5}' | awk -F: '{print $1}'

列出所有发送SYN\_REC连接节点的IP地址。

netstat -an | grep :80 | sort

显示所有80端口的网络连接并排序。这里的80端口是http端口，所以可以用来监控web服务。如果看到同一个IP有大量连接的话就可以判定单点流量攻击了。

netstat -ntu | grep ESTAB | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -nr

检查 ESTABLISHED 连接并且列出每个IP地址的连接数量。

netstat -plan|grep :80|awk {'print $5'}|cut -d: -f 1|sort|uniq -c|sort -nk 1

列出所有连接到本机80端口的IP地址和其连接数。80端口一般是用来处理HTTP网页请求。

netstat -anp |grep 'tcp|udp' | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -n

列出所有连接到本机的UDP或者TCP连接的IP数量。

**netstat -n -p|grep SYN\_REC | wc -l**

**这个命令可以查找出当前服务器有多少个活动的 SYNC\_REC 连接。正常来说这个值很小，最好小于5。 当有Dos攻击或者邮件炸弹的时候，这个值相当的高。**尽管如此，这个值和系统有很大关系，有的服务器值就很高，也是正常现象。

如何通过netstat命令缓解DDOS攻击呢？一旦获得攻击服务器的IP地址就可以使用以下命令拒绝此IP的所有连接。

iptables -A INPUT 1 -s $IPADRESS -j DROP/REJECT

注意，你需要将 $IPADRESS 替换成需要拒绝连接的IP地址。执行完以上命令后，使用以下命令结束所有的httpd连接以清理系统。

killall -KILL httpd

然后执行以下命令重启httpd服务。

service httpd start           #RedHat 系统

/etc/init/d/apache2 restart   #Debian 系统

当然通过netstat命令只能缓解部分DDoS攻击，而且还很有可能造成大量误封正常访客。所以墨者安全还是建议最好通过专业的网络安全公司部署DDoS高防IP，对恶意攻击流量进行清洗，隐藏服务器源IP，确保源站稳定运行。

**对付分布式拒绝服务攻击的方法有哪些？举例说明（参考课本p77）**

**企业网管理员**

网管员做为一个企业内部网的管理者，往往也是安全员、守护神。在他维护的网络中有一些服务器需要向外提供WWW服务，因而不可避免地成为DDoS的攻击目标，他该如何做呢？可以从主机与网络设备两个角度去考虑。

主机上的设置  
几乎所有的主机平台都有抵御DoS的设置，总结一下，基本的有几种：

* 关闭不必要的服务
* 限制同时打开的Syn半连接数目
* 缩短Syn半连接的time out 时间
* 及时更新系统补丁

网络设备上的设置  
企业网的网络设备可以从防火墙与路由器上考虑。这两个设备是到外界的接口设备，在进行防DDoS设置的同时，要注意一下这是以多大的效率牺牲为代价的，对你来说是否值得。

１.防火墙

* 禁止对主机的非开放服务的访问
* 限制同时打开的SYN最大连接数
* 限制特定IP地址的访问
* 启用防火墙的防DDoS的属性
* 严格限制对外开放的服务器的向外访问

第五项主要是防止自己的服务器被当做工具去害人。

２.路由器  
以Cisco路由器为例

* Cisco Express Forwarding（CEF）
* 使用 unicast reverse-path
* 访问控制列表（ACL）过滤
* 设置SYN数据包流量速率
* 升级版本过低的ISO
* 为路由器建立log server

其中使用CEF和Unicast设置时要特别注意，使用不当会造成路由器工作效率严重下降，升级IOS也应谨慎。路由器是网络的核心设备，与大家分享一下进行设置修改时的小经验，就是先不保存。Cisco路由器有两份配置startup config和running config，修改的时候改变的是running config，可以让这个配置先跑一段时间（三五天的就随意啦），觉得可行后再保存配置到startup config；而如果不满意想恢复原来的配置，用copy start run就行了。

**假如你的主机正在面临DNS欺骗攻击，你打算采取什么解决策略和方案？**

**DNS欺骗检测和防范思路**

**3.1 检测思路**

      发生DNS欺骗时，Client最少会接收到两个以上的应答数据报文，报文中都含有相同的ID序列号，一个是合法的，另一个是伪装的。据此特点，有以下两种检测办法：

      (1)被动监听检测。即监听、检测所有DNS的请求和应答报文。通常DNS Server对一个请求查询仅仅发送一个应答数据报文(即使一个域名和多个IP有映射关系，此时多个关系在一个报文中回答)。因此在限定的时间段内一个请求如果会收到两个或以上的响应数据报文，则被怀疑遭受了DNS欺骗。

      (2)主动试探检测。即主动发送验证包去检查是否有DNS欺骗存在。通常发送验证数据包接收不到应答，然而黑客为了在合法应答包抵达客户机之前就将欺骗信息发送给客户，所以不会对DNS Server的IP合法性校验，继续实施欺骗。若收到应答包，则说明受到了欺骗攻击。

**3.2 防范思路**

      在侦测到网络中可能有DNS欺骗攻击后，防范措施有：①在客户端直接使用IP Address访问重要的站点，从而避免DNS欺骗; ②对DNS Server和Client的数据流进行加密，Server端可以使用SSH加密协议，Client端使用PGP软件实施数据加密。

      对于常见的ID序列号欺骗攻击，采用专业软件在网络中进行监听检查，在较短时间内，客户端如果接收到两个以上的应答数据包，则说明可能存在DNS欺骗攻击，将后到的合法包发送到DNS Server并对DNS数据进行修改，这样下次查询申请时就会得到正确结果。

**四、DNS防护方案**

**4.1 进行IP地址和MAC地址的绑定**

      (1)预防ARP欺骗攻击。因为DNS攻击的欺骗行为要以ARP欺骗作为开端，所以如果能有效防范或避免ARP欺骗，也就使得DNS ID欺骗攻击无从下手。例如可以通过将Gateway Router 的Ip Address和MAC Address静态绑定在一起，就可以防范ARP攻击欺骗。

      (2)DNS信息绑定。DNS欺骗攻击是利用变更或者伪装成DNS Server的IP Address，因此也可以使用MAC Address和IP Address静态绑定来防御DNS欺骗的发生。由于每个Network Card的MAC Address具有唯一性质，所以可以把DNS Server的MAC Address与其IP Address绑定，然后此绑定信息存储在客户机网卡的Eprom中。当客户机每次向DNS Server发出查询申请后，就会检测DNS Server响应的应答数据包中的MAC Address是否与Eprom存储器中的MAC Address相同，要是不同，则很有可能该网络中的DNS Server受到DNS欺骗攻击。这种方法有一定的不足，因为如果局域网内部的客户主机也保存了DNS Server的MAC Address，仍然可以利用MAC Address进行伪装欺骗攻击。

**4.2 使用Digital Password进行辨别**

      在不同子网的文件数据传输中，为预防窃取或篡改信息事件的发生，可以使用任务数字签名(TSIG)技术即在主从Domain Name Server中使用相同的Password和数学模型算法，在数据通信过程中进行辨别和确认。因为有Password进行校验的机制，从而使主从Server的身份地位极难伪装，加强了Domain Name信息传递的安全性。

      安全性和可靠性更好的Domain Name Service是使用域名系统的安全协议(Domain Name System Security, DNSSEC))，用Digital Signature的方式对搜索中的信息源进行分辨，对DATA的完整性实施校验，DNSSEC的规范可参考RFC2605。因为在设立Domain时就会产生Password，同时要求上层的Domain Name也必须进行相关的Domain Password Signature，显然这种方法很复杂，所以InterNIC域名管理截至目前尚未使用。然而就技术层次上讲，DNSSEC应该是现今最完善的Domain Name设立和解析的办法，对防范Domain Name欺骗攻击等安全事件是非常有效的。

**4.3 优化DNS SERVER的相关项目设置**

      对于DNS Server的优化可以使得DNS的安全性达到较高的标准，常见的工作有以下几种：①对不同的子网使用物理上分开的Domain Name Server,从而获得DNS功能的冗余;②将外部和内部Domain Name Server从物理上分离开并使用Forwarders转发器。外部Domain Name Server可以进行任何客户机的申请查询，但Forwarders则不能，Forwarders被设置成只能接待内部客户机的申请查询;③采用技术措施限制DNS动态更新;④将区域传送(zone transfer)限制在授权设备上;⑤利用事务签名对区域传送和区域更新进行数字签名;⑥隐藏服务器上的Bind版本;⑦删除运行在DNS服务器上的不必要服务，如FTP、telnet和Http;⑧在网络外围和DNS服务器上使用防火墙,将访问限制在那些DNS功能需要的端口上。

**4.4 直接使用IP地址访问**

      对个别信息安全等级要求十分严格的WEB站点尽量不要使用DNS进行解析。由于DNS欺骗攻击中不少是针对窃取客户的私密数据而来的，而多数用户访问的站点并不涉及这些隐私信息，因此当访问具有严格保密信息的站点时，可以直接使用IP地址而无需通过DNS解析，这样所有的DNS欺骗攻击可能造成的危害就可以避免了。除此，应该做好DNS Server的安全配置项目和升级DNS软件，合理限定DNS Server进行响应的IP地址区间，关闭DNS Server的递归查询项目等。

**4.5 对DNS数据包进行监测**

      在DNS欺骗攻击中，Client会接收到至少两个DNS的数据响应包，一个是真实的数据包，另一个是攻击数据包。欺骗攻击数据包为了抢在真实应答包之前回复给Client，它的信息数据结构与真实的数据包相比十分简单，只有应答域，而不包括授权域和附加域。因此，可以通过监测DNS响应包，遵循相应的原则和模型算法对这两种响应包进行分辨，从而避免虚假数据包的攻击。

**TCP/IP的安全性隐患？改进措施？**

**1、链路层上的攻击**

在TCP/IP网络中，链路层这一层次的复杂程度是最高的。其中最常见的攻击方式通常是网络嗅探组成的TCP/IP协议的以太网。

以太网卡有两种主要的工作方式，一种是一般工作方式，另一种是较特殊的混杂方式。这一情况下，很可能由于被攻击的原因而造成信息丢失情况，且攻击者可以通过数据分析来获取账户、密码等多方面的关键数据信息。

**2、网络层上的攻击**

如果ARP识别链接错误，这样的话ARP直接应用可疑信息，那么可疑信息就会很容易进入目标主机当中。ARP协议没有状态，不管有没有收到请求，主机会将任何受到的ARP相应自动缓存。

如果信息中带有病毒，采用ARP欺骗就会导致网络信息安全泄露。因此，在ARP识别环节，应加大保护，建立更多的识别关卡，不能只简单通过IP名进行识别，还需充分参考IP相关性质等。

**3、传输层上的攻击**

在传输层还存在网络安全问题。如在网络安全领域中，IP欺骗就是隐藏自己的有效手段，主要是通过将自身IP地址进行伪造，并向目标主机发送恶意的请求，攻击主机，而主机却因为IP地址被隐藏而无法准确确认攻击源。或者通过获取目标主机信任而趁机窃取相关的机密信息。

**4、应用层上的攻击**

对于因特网而言，IP地址与域名均是一一对应的，这两者之间的转换工作，被称为域名解析。而DNS就是域名解析的服务器。DNS欺骗指的是攻击方冒充域名服务器的行为，使用DNS欺骗能将错误DNS信息提供给目标主机。所以说，通过DNS欺骗可误导用户进入非法服务器，让用户相信诈骗IP。