😆

PPT3 4 7 章加红重点

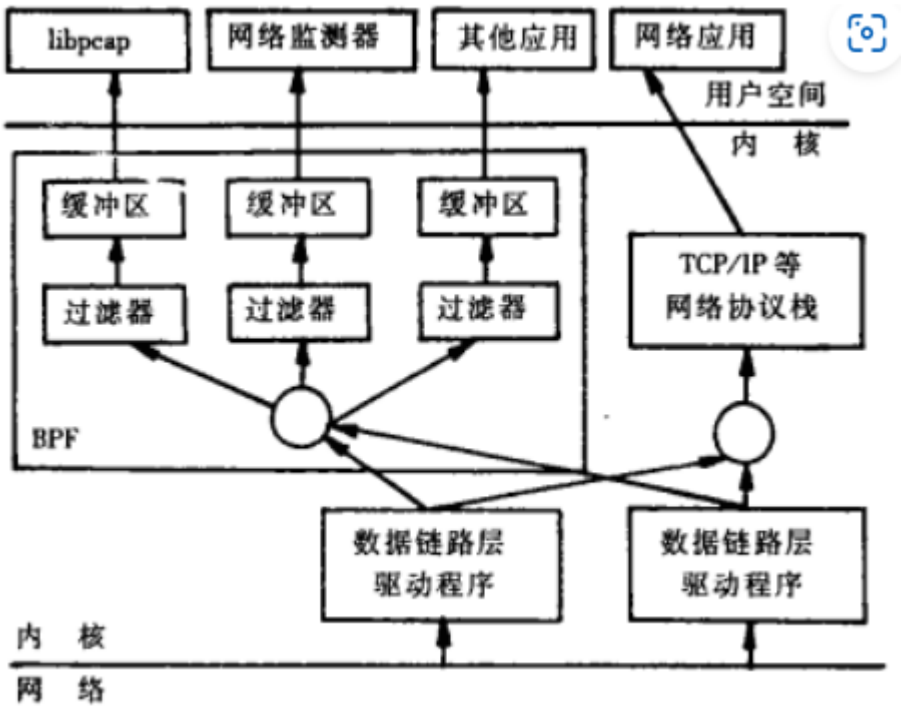
第三章网络侦察技术2

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

BPF和lipcap：

Libpcap 主要由两部份组成：网络分接头(Network Tap)和数据过滤器(Packet Filter)。网络分接头从网络设备驱动程序中收集数据拷贝，过滤器决定是否接收该数据包。Libpcap利用BSD Packet Filter(BPF)算法对网卡接收到的链路层数据包进行过滤。BPF算法的基本思想是在有BPF监听的网络中，网卡驱动将接收到的数据包复制一份交给 BPF 过滤器，过滤器根据用户定义的规则决定是否接收此数据包以及需要拷贝该数据包的那些内容，然后将过滤后的数据给与过滤器相关联的上层应用程序。BPF的架构如图所示：



WINPCAP

WinPcap 是 BPF 模型和 Libpcap 函数库在 Windows 平台下网络数据包捕获和网络状态分析的一种体系结构，这个体系结构是由一个核心的包过滤驱动程序，一个底层的动态连接库 Packet.dll 和一个高层的独立于系统的函数库 Libpcap 组成。

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。

利用winpcap进行网络数据包的捕获和过滤的设计步骤

1）打开网卡，并设为混杂模式。

2）回调函数 Network Tap 在得到监听命令后，从

网络设备

驱动程序处收集数据包把监听到的数据包负责传送给过滤程序

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
- 检测手段：
  - 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
  - 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

交换以太网监听技术：

交换以太网中，交换机能根据数据帧中的目的MAC地址将数据帧准确地送到目的主机的端口，而不是所有的端口。所以交换式网络环境在一定程度上能抵御Sniffer攻击。

交换机的工作原理不同于HUB的共享式报文方式，交换机转发的报文是一一对应的，能够隔离冲突域和有效的抑制广播风暴的产生。由此看来，交换环境下再采用传统的共享式以 太网下网络监听是不可能了，由于报文是一一对应转发的，普通的网络监听软件此时无法监听到交换环境下其他主机任何有价值的数据。但是，以太网内主机数据包的传送完成不是依 靠IP地址，而是依靠ARP找出IP地址对应的MAC地址实现的。而ARP协议是不可靠和无连接的 ，通常即使主机没有发出ARP请求，也会接受发给他的ARP回应，并将回应的MAC和IP对应关 系放入自己的ARP缓存中。因此利用ARP协议，交换机的安全性也面临着严峻的考验。

1.2.1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。

1.2.2ARP协议和欺骗

在以太网中传输的数据包是以太包，而以太包的寻址是依据其首部的物理地址（MAC地址 ）。仅仅知道某主机的逻辑地址（IP地址）并不能让内核发送一帧数据给此主机，内核必须 知道目的主机的物理地址才能发送数据。ARP协议的作用就是在于把逻辑地址变换成物理地 址，也既是把32 b的IP地址变换成48 b的以太地址。每一个主机都有一个ARP高速缓存，此 缓存中记录了最近一段时间内其他IP地址与其MAC地址的对应关系。如果本机想与某台主机 通信，则首先在ARP高速缓存中查找此台主机的IP和MAC信息，如果存在，则直接利用此MAC 地址构造以太包；如果不存在，则向本网络上每一个主机广播一个ARP请求包。其意义是“ 如果你有此IP地址，请告诉我你的MAC地址”，目的主机收到此请求包后，发送一个ARP响应 包，本机收到此响应包后，把相关信息记录在ARP高速缓存中。可以看出，ARP协议是有缺点 的，第三方主机可以构造一个ARP欺骗包，而源主机却无法分辨真假。

假定A为进行监听的主机，B为被监听的主机，C为其他网络主机。当A收到B向C发出的ARP 请求包后，向B回应一个ARP应答。向C主动发送一个应答，修改C缓存中的关于B的IPMAC映射。当A收到C向B发出的ARP请求时，向B主动发送一个应答，修改B缓存中的关于C的 IPMAC映射。这样，构造了ARP欺骗包（欺骗B对C的连接）。事实上，A成了B的代理 可以全部捕获到B和C的相关数据。

口令破解

分析漏洞破解口令：如果能够得到口令验证代码

- Crack:如果能够修改口令验证代码
- Vulnerability:如果不能修改代码，但可以构造恶意输入
  通过引发代码指针异常，改变控制流绕过口令
  检查通过引发数据指针异常，导致任意位置读或
- Keygen:如果能够破解口令验证算法通过口令验证算法漏洞，逆推口令

离线破解口令

- 获取口令密文:网络包,注册表、口令文件,内存镜像
- 尝试口令:Rainbow table Dictionary(用户的名字、生日、电话号码、身份证号码、所居住街道的名字等)

在线破解口令

- 尝试口令:同上
- 防Anti

第四章 DOS

定义：阻止或拒绝合法使用者存取网络服务器的一种破坏性攻击方式。攻击者想办法让目标机器停止提供服务

这种攻击往往是针对TCP／IP协议中的某个弱点，或者系统存在的某些漏洞，对目标系统发起的大规模进攻使服务器充斥大量要求回复的信息，消耗网络带宽或系统资源，导致目标网络或系统不胜负荷以至于瘫痪而无法向合法的用户提供正常的服务

攻击原理：

- syn flood：SYN Flood是当前最流行的DoS（拒绝服务攻击）与DDoS（Distributed Denial Of Service分布式拒绝服务攻击）的方式之一，这是一种利用TCP协议缺陷，发送大量伪造的TCP连接请求，使被攻击方资源耗尽（CPU满负荷或内存不足）的攻击方式

- TCP连接的三次握手中，假设一个用户向服务器发送了SYN报文后突然死机或掉线，那么服务器在发出SYN+ACK应答报文后是无法收到客户端的ACK报文的（第三次握手无法完成），这种情况下服务器端一般会重试（再次发送SYN+ACK给客户端）并等待一段时间后丢弃这个未完成的连接。这段时间的长度我们称为SYN Timeout，一般来说这个时间是分钟的数量级（大约为30秒-2分钟）；一个用户出现异常导致服务器的一个线程等待1分钟并不是什么很大的问题，但如果有一个恶意的攻击者大量模拟这种情况（伪造IP地址），那么服务器端将为了维护一个非常大的半连接列表而消耗非常多的资源。即使是简单的保存并遍历也会消耗非常多的CPU时间和内存，何况还要不断对这个列表中的IP进行SYN+ACK的重试。实际上，如果服务器的TC P/IP栈不够强大，那么最后的结果往往是堆栈溢出崩溃——即使服务器端的系统足够强大，服务器端也将忙于处理攻击者伪造的TCP连接请求而无暇理睬客户的正常请求（毕竟客户端的正常请求比率非常之小），此时，从正常客户的角度看来，服务器失去响应，这种情况就称做：服务器端受到了SYN Flood攻击
- 应对：优化系统配置；优化路由器配置； 使用防火墙；主动监视； 完善基础设施

- smurf攻击：通过使用将回复地址设置成受害网络的广播地址的ICMP应答请求（ping）数据包来淹没受害主机的方式进行。最终导致该网络的所有主机都对此ICMP应答请求作出答复，导致网络阻塞。它比ping of death洪水的流量高出1或2个数量级。更加复杂的Smurf将源地址改为第三方的受害者，最终导致第三方崩溃

- teardrop攻击:在TCP/IP堆栈中实现信任IP碎片中的包的标题头所包含的信息来实现自己的攻击。IP分段含有指明该分段所包含的是原包的哪一段的信息，某些TCP/IP（包括service pack 4以前的NT）在收到含有重叠偏移的伪造分段时将崩溃
- Land攻击：**利用服务程序中的处理错误使服务失效。**用一个特别打造的SYN包，它的原地址和目标地址都被设置成某一个服务器地址。此举将导致接受服务器向它自己的地址发送SYN-ACK消息，结果这个地址又发回ACK消息并创建一个空连接。被攻击的服务器每接收一个这样的连接都将保留，直到超时，对Land攻击反应不同，许多UNIX实现将崩溃，NT变的极其缓慢（大约持续5分钟）

攻击模式：消耗资源；网络带宽、存储空间、 CPU时间等；破坏或改变配置信息； 物理破坏或者改变网络部件；利用服务程序中的处理错误使服务失效。

发起方式：传统的拒绝服务攻击；分布式拒绝服务攻击DDos

分布式拒绝服务供给DDoS:

传统的拒绝服务攻击的缺点:受网络资源的限制;隐蔽性

分布式拒绝服务攻击可以使很多的计算机在同一时间遭受到攻击，使攻击的目标无法正常使用，分布式拒绝服务攻击已经出现了很多次，导致很多的大型网站都出现了无法进行操作的情况

原理：分布式拒绝服务攻击DDoS是一种基于DoS的特殊形式的拒绝服务攻击，是一种分布的、协同的大规模攻击方式。单一的DoS攻击一般是采用一对一方式的，它利用网络协议和操作系统的一些缺陷，采用欺骗和伪装的策略来进行网络攻击，使网站服务器充斥大量要求回复的信息，消耗网络带宽或系统资源，导致网络或系统不胜负荷以至于瘫痪而停止提供正常的网络服务。与DoS攻击由单台主机发起攻击相比较，分布式拒绝服务攻击DDoS是借助数百、甚至数千台被入侵后安装了攻击进程的主机同时发起的集团行为

应对： 在数据流中搜寻特征字符串；利用攻击数据包的某些特征；监视本地主机端口的使用； 对通信数据量进行统计

第七章欺骗攻击

DNS欺骗

- 简介：也称为 DNS 缓存投毒（DNS Cache Poisoning），是一种网络攻击方式，攻击者通过伪造或篡改DNS服务器返回的DNS解析结果，将受害者重定向到恶意网站或者收集用户的敏感信息。
- 原理：DNS 欺骗攻击利用 DNS 解析过程中的漏洞，向受害者发送伪造的 DNS 解析响应，将目标域名解析到攻击者控制的恶意 IP 地址，从而实现重定向、劫持和信息窃取等恶意行为。
- 防御：检查本机的HOSTS文件，以免被攻击者加了恶意站点进去；其次要确认自己使用的DNS服务器是ISP提供的，因为当前ISP服务器的安全工作还是做得比较好的，一般水平的攻击者无法成功进入；如果是依靠网关设备自带的DNS解析来连接

Internet的，就要拜托管理员定期检查网关设备是否遭受入侵。

EMAIL欺骗

原理：伪造电子邮件头，导致信息看起来来源于某个人或某个地方，而实际却不是真实的源地址。垃圾邮件的发布者通常使用欺骗和恳求的方法尝试让收件人打开邮件，并很有可能让其回复。可以合法地使用欺骗。经典的例子有，发件人喜欢伪装电子邮件源地址，而信的内容则包括了发件人所陈述的从被配偶虐待到福利代理，或者害怕报仇的“告密者”等一系列事件。但欺骗其他人在某些情况下是违法的

原因：可能发生电子邮件欺骗的原因在于发送电子邮件最主要的协议：简单邮件传输协议（SMTP）不包括某种认证机制。即使SMTP服务扩展（工程任务组，请求注解2554号文件中有详细说明）允许SMTP客户端通过邮件服务器来商议安全级别，但这一预防措施并不是什么时候都会被使用。

WEB欺骗：

原理：

攻击者通过伪造某个WWW站点的影像拷贝，使该Web的入口进入到攻击者的Web影像服务器，并经过攻击者机器的过滤作用，从而达到攻击者监控受攻击者的任何活动以获取有用信息的目的。

形式：

- 使用相似域名
- 改写URL
- 劫持web会话

防御：

- 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
  - 使用SSL

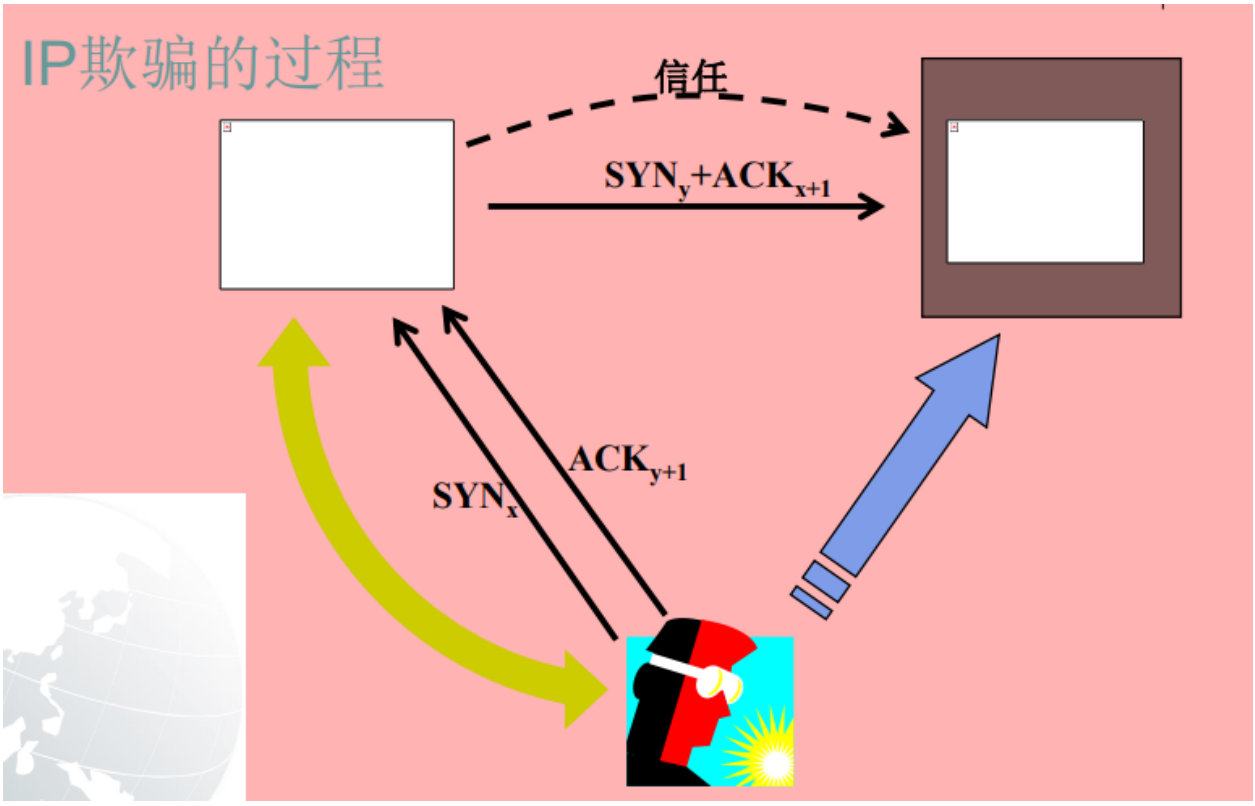
- Web会话劫持
    - 养成显式注销的习惯
    - 使用长的会话ID
- Web的安全问题很多，我们需要更多的手段来保证Web安全

IP欺骗

行动产生的IP数据包为伪造的源IP地址，以便冒充其他系统或发件人的身份。这是一种黑客的攻击形式，黑客使用一台计算机上网,而借用另外一台机器的IP地址,从而冒充另外一台机器与服务器打交道

IP欺骗的动机:

⚫ 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



会话劫持：猜测序列号是成功劫持TCP会话的关键

会话劫持的原理

在TCP会话劫持中，攻击者需要实现以下步骤：

1. **截获会话**：攻击者必须能够监视和截获目标会话的数据包。这通常通过网络嗅探工具实现。
2. **猜测序列号**：TCP会话的每个数据包都有一个序列号。攻击者必须准确猜测出当前会话的有效序列号，才能成功插入恶意数据包或劫持会话。
3. **注入数据包**：一旦成功猜测序列号，攻击者可以伪造一个数据包并插入到会话中。这个数据包看起来像是来自合法的一方，从而欺骗另一方。

序列号猜测

序列号是32位的数字，理论上有2^32种可能性。但在实际应用中，由于某些实现方式和环境因素，序列号的生成可能并不完全随机，这给攻击者提供了一些猜测的线索。常见的序列号预测方法包括：

1. **时间序列法**：有些操作系统根据时间戳生成序列号，攻击者可以通过分析时间差来预测序列号。
2. **顺序法**：某些系统会采用顺序递增的方式生成序列号，攻击者可以通过观察模式来预测下一个序列号。

防御措施

为了防止会话劫持，可以采取以下防御措施：

1. **加密通信**：使用加密协议（如TLS/SSL）来保护数据传输，防止攻击者截获和篡改数据包。
2. **使用随机序列号**：确保TCP/IP栈使用高质量的随机数生成器来生成序列号，增加攻击者猜测序列号的难度。
3. **网络隔离**：通过防火墙、VPN等技术隔离和保护内部网络，减少攻击者接触会话数据包的机会。
4. **入侵检测系统**：部署入侵检测和防御系统，监控网络流量，及时发现和响应可疑活动。