第 7 章

假消息攻击

内容提要

网络协议在设计和实现过程中由于缺乏安全性考虑或者为保证兼容性等原因，可能导致存在安全缺陷。假消息攻击就是利用网络协议的安全缺陷，通过发送伪造的协议数据包或篡改协议数据包内容的方式，达到窃取网络通信数据、窥探隐私、拒绝服务等目的。本章介绍了利用假消息攻击的基本原理，其中包括 ARP（Address Resolution Protocol）欺骗、DNS（Domain Name Service）欺骗、HTTP（Hyper Text Transfer Protocol，超文本传输协议）中间人攻击等假消息攻击实验。

本章重点

- ARP 欺骗原理及实践；
- DNS 欺骗原理及实践；
- HTTP 中间人攻击原理及实践。

7.1 概述

TCP/IP 协议簇是目前互联网中使用最为广泛的协议簇，它起源于 20 世纪 60 年代末美国政府资助的一个分组交换网络研究项目，它也被称做互联网的基础。然而在设计之初，设计者们只是想办法将遍布在全世界的各个孤立的计算机连接在一起，而并没有考虑其中可能存在的安全隐患。

1．缺乏严格的身份验证机制

设计者们假设 TCP/IP 被应用于一个可信的网络环境中，没有充分考虑数据传输过程中可能存在伪造身份的问题，而仅以 IP 地址作为通信身份的标志。因此有些协议可能非常容易被攻击者利用来欺骗受害者，使得欺骗者能够与被欺骗者建立信任连接。

2．缺乏有效的数据加密机制

出于同样的原因，设计者们也没有充分考虑数据传输过程中可能存在的恶意监听和篡改数据的问题，因此大部分的 TCP/IP 协议都没有使用加密技术。即使到了今天，仍然有许多广为流行的协议采用明文数据传输，如 HTTP、DNS 和 SMTP（Simple Mail Transfer Protocol）等。

假消息攻击充分利用了上述这两类隐患，采取假消息攻击方式进行攻击。按不同分类方式，假消息攻击又可以分为三种形式。

（1）按攻击效果可以将假消息攻击分为：

① 信息窃取攻击。以窃取通信一方或通信双方内容为目的，窃取的敏感信息一般包括用户名和密码、文档、密钥、证书等。

② 拒绝服务攻击。以击垮对方的网络服务为目的，使之无法提供正常的服务。

（2）按攻击者所处的位置可以分为：

① 中间人攻击。攻击节点位于被攻击节点与其他节点通信的必经信道上，可以获取、转发和篡改它们的通信数据。

② 嗅探攻击。攻击节点位于被攻击节点与其他节点通信信道的附近，可以获取它们的通信数据。

（3）按攻击协议的不同可以分为：

① 数据链路层的攻击，典型的如针对 ARP 协议的欺骗攻击；

② 网络层的攻击，如 ICMP（Internet Control Message Protocol)路由重定向攻击及 IP 分片攻击；

③ 传输层的攻击，如 SYN 洪水攻击和 TCP 序号猜测攻击；

④ 应用层的攻击，如 DNS 欺骗攻击和 HTTP 中间人攻击。

下面重点介绍和实践几种典型的假消息攻击方式。

7.2 假消息攻击原理

7.2.1 ARP 欺骗

IP 数据包在通过以太网发送时，以太网设备并不识别 32bit 的 IP 地址，而是以 48bit

的 MAC 地址传输以太网数据包。因此，操作系统必须通过目的 IP 地址获得相应的目的 MAC 地址。ARP（Address Resolution Protocol，地址解析协议）就是用于确定这两种地址之间映射关系的协议，它通过请求/响应机制将 IP 地址转换为 MAC 地址。

ARP 数据包格式如图 7.1 所示。

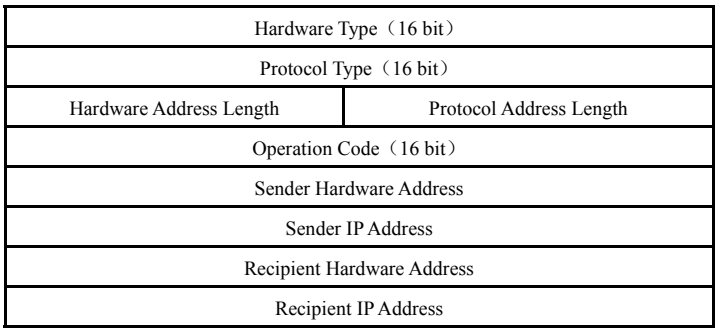

图 7.1　ARP 数据包格式

其中，Operation Code 域用来指定这个包是 ARP 请求包还是响应包，分别对应数字 1 和 2。在 ARP 请求包中，Sender Hardware Address 和 Sender IP Address 填充的是请求方的 MAC 地址和 IP 地址，此时 Recipient IP Address 填充被请求方的 IP 地址，而由于被请求方的 MAC 地址未知，Recipient Hardware Address 会填充为全 0；反之在 ARP 响应包中，这后四个选项均填充为相应内容，于是请求方就能从响应包中的 Sender Hardware Address 字段获得被请求方的 MAC 地址了。

由于发送 ARP 请求包时并不知道被请求方的 MAC 地址，所以 ARP 请求包是以广播方式在以太网中传播的。如果每台主机每次发送数据之前都要询问一次 MAC 地址，就会给以太网带来不小的广播压力。因此 ARP 协议在实现时都采用了 ARP 缓存机制，即将获得的 IP-MAC 地址对缓存起来，以节约不必要的 ARP 通信开销。另外为了提高网络的传输效率，ARP 协议在实现时还采取了另外两个措施：

（1）响应 ARP 请求的主机将请求者的 IP-MAC 地址对映射于缓存；

（2）主动的 ARP 响应会被视为有效信息而被目的主机接收。

通过以上介绍可以发现，ARP 协议并没有采用加密机制，也没有做严格的身份验证。实际上，以太网上的任何主机都可以冒充网内其他主机来发送 ARP 请求或响应包，如构造虚假的 Sender Hardware Address 和 Sender IP Address 数据发送给被欺骗的主机。按照以上 ARP 实现机制，无论这个数据包是请求类型还是响应类型，被欺骗主机都会将虚假的 IP-MAC 地址对映射于缓存，这样做的后果是：被欺骗主机今后再往虚假 IP 地址发送数据时，都会被发送到虚假 MAC 地址上。这就是 ARP 欺骗的原理。

ARP 欺骗的危害包括：

（1）拒绝服务。ARP 欺骗用错误的 IP-MAC 地址对污染目标主机的 ARP 缓存，使目标主机丧失与某 IP 主机的通信能力，如果将欺骗应用于目标主机与网关之间，会使得目标主机无法连接外部网络。

（2）中间人攻击。攻击者同时欺骗目标主机与网关，重定向它们之间的数据传输到

自身，相当于在两者间建立了一条间接的通信通道，从而可以以中间人身份嗅探和篡改通信的全部数据。

ARP 欺骗的防御对策：

（1）建立 DHCP 服务器，使得所有客户机的 IP 地址及其相关主机信息，只能从网关取得；给每个网卡绑定固定唯一的 IP 地址，以保持网内的主机 IP-MAC 地址对的对应关系。

（2）建立 MAC 数据库，把网内所有网卡的 MAC 地址记录下来，将每个 MAC 和 IP 地理位置信息统统装入数据库，以便及时查询备案。

（3）给网关关闭 ARP 动态刷新的过程，使用静态路由，使得攻击者无法用 ARP 欺骗攻击网关，确保局域网的安全。

（4）利用网关监听网络安全。由于 ARP 欺骗攻击包一般有两个特点，存在任何一个特点即可视为攻击包，立刻报警：① 以太网数据包头的源地址、目标地址与 ARP 数据包的协议地址不匹配；② ARP 数据包的发送和目标地址不在自己网络网卡的 MAC 数据库内，或者与自己网络网卡内 MAC 数据库的 IP-MAC 地址对不匹配。因此可以据此对局域网内的 ARP 数据包进行分析。

（5）使用 VLan 或 PVLan 技术，将网络分段使 ARP 欺骗的影响范围降至最小。

7.2.2　DNS 欺骗

DNS 是域名服务的缩写，DNS 协议用于解析网络中域名和 IP 地址的映射关系。当客户端向 DNS 服务器发出域名查询请求时，DNS 服务器提供对应的 IP 地址以作响应。

DNS 域名空间是一种树状结构，包括根、一级域名、二级域名、多级子域名和主机名，如图 7.2 所示。

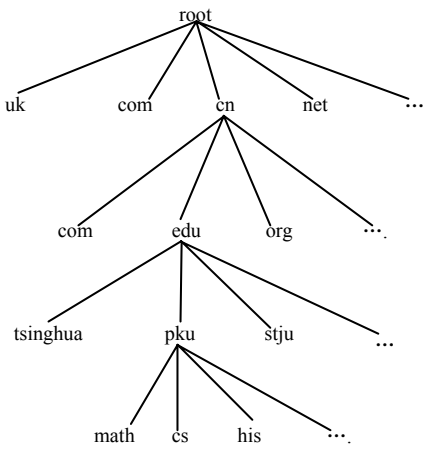

图 7.2　DNS 域名空间树状结构

当客户端向 DNS 服务器提出查询请求时，每个查询信息都包括两部分信息：一是指定的 DNS 域名，要求使用完整名称；二是指定查询类型，既可以指定资源记录类型又可以指定查询操作的类型。例如，指定的名称为一台计算机的完整主机名称

“hostname.example.microsoft.com”，指定的查询类型为该名称的 IP 地址，可以理解为客户端询问服务器“你有关于计算机的主机名称为 hostname.example.microsoft.com 的 IP 地址记录吗？”当客户端收到服务器的回答信息时，从中获得查询名称的 IP 地址。

DNS 的查询解析可以通过多种方式实现：①客户端利用缓存记录的以前的查询信息直接回答查询请求；②DNS 服务器利用缓存中的记录信息回答查询请求；③DNS 服务器通过查询其他服务器获得查询信息并将它发送给客户端，这种查询方式称为递归查询；④客户端通过 DNS 服务器提供的地址直接尝试向其他 DNS 服务器提出查询请求，这种查询方式称为反复（迭代）查询。

与 ARP 协议的实现类似，DNS 协议的实现也没有采用加密机制和严格的身份验证机制，因此很容易对 DNS 的解析过程进行欺骗。其欺骗的过程如下：

（1）客户端首先以特定的 ID 向 DNS 服务器发送域名查询数据包；

（2）DNS 服务器查询之后以相同的 ID 给客户端发送域名响应数据包；

（3）攻击者捕获到这个响应包后，将域名对应的 IP 地址修改为其他 IP 地址，并向客户端返回该数据包；

（4）客户端将收到的 DNS 响应数据包 ID 与自己发送的查询数据包 ID 相比较，如果匹配则信任该响应信息。此后客户端在访问该域名时将被重定向到虚假的 IP 地址。

实施 DNS 欺骗的关键是给出正确的 ID，这可以通过中间人攻击或网络嗅探来解决。

防御对策：

与防范 ARP 欺骗类似，可以通过对常用站点构造静态“域名——IP 地址”映射表来达到防范 DNS 欺骗的目的。在各种操作系统中都允许使用这样的静态表，比如在 Windows 系统中，可以编辑 system32\drivers\etc\hosts 文件来建立这样的静态表。

7.2.3　HTTP 中间人攻击

HTTP 主要用于 Web 程序通信，是一个属于应用层的面向对象的协议，于 1990 年提出。目前广泛使用的是其 1.1 版本，2.0 版本已在 2013 年 8 月开始测试。HTTP 支持客户端/服务器模式，采用简单快速的请求/响应方式，常用的请求有 GET、HEAD、POST 等方式。由于 HTTP 协议简单，使得 HTTP 服务器的程序规模小，因而通信速度很快。此外 HTTP 协议还有以下特点：

（1）灵活。HTTP 允许传输任意类型的数据对象，如图片、多媒体、二进制数据流等，由 Content-Type 标记数据类型。

（2）无连接。无连接的含义是指限制每次连接只处理一个请求，服务器处理完客户端的请求，并得到客户端的响应后，即断开连接。

（3）无状态。HTTP 协议是无状态协议。所谓无状态是指协议对于事务处理没有记忆能力，虽然在发生错误时会带来重传的损耗，但能够简化逻辑，因而适用于大规模并行传输。

HTTP 协议的实现由客户端发送的 Request 包和服务器返回的 Response 包构成，其中 Request 包由以下部分组成：

（1）请求行，由请求方法字段、URL 字段和 HTTP 协议版本字段三个字段组成，它

们用空格分隔，如 GET /index.html HTTP/1.1。

（2）请求头部，由（关键字:<空格>值）对组成，每行一对，关键字和值用英文冒号“:<空格>”分隔。请求头部通知服务器有关客户端请求的信息，典型的请求头部有：

① User-Agent，产生请求的浏览器类型。

② Accept，客户端可识别的内容类型列表。

③ Host，请求的主机名，允许多个域名同处一个 IP 地址，即虚拟主机。

④ Cookie，客户端发送的与当前域名有关的本地信息。

Response 包由以下部分组成：

（1）状态行：包括 HTTP 协议版本号、状态码、状态码的文本描述信息，如 HTTP/1.1 200 OK。其中，状态码由一个三位数组成，状态码一般有五种含义：

① 1xx，表示指示信息，意思是请求信息收到，继续处理。

② 2xx，表示成功，指操作信息成功收到，理解和接受。例如，200 表示请求成功，206 表示断点续传。

③ 3xx，表示重定向。为了完成请求，必须采取进一步措施，如跳转到新的地址。

④ 4xx，表示客户端错误，指请求的语法有错误或不能完全被满足，如 404 表示文件不存在。

⑤ 5xx，表示服务器错误，指服务器无法完成明显有效的请求，如 500 表示内部错误。

（2）响应头部：与请求头部类似，一般包括以下内容：

① Set-Cookie：Set-Cookie 由服务器发送，它包含在响应请求的头部，用于在客户端创建一个 Cookie，Cookie 头由客户端发送，包含在 HTTP 请求的头部中。其设置格式是 name=value，设置多个参数时中间用分号隔开。

② Location：当服务器返回 3xx 重定向时，由该参数实现重定向。

③ Content-Length：指明附属体（数据实体）的长度。

（3）附属体：返回页面的实际内容。

从以上介绍可以发现，HTTP 协议内容都采用了明文定义，因此很容易受到嗅探和中间人攻击。

防御对策：

虽然 HTTP 的安全版本 HTTPS（Hyper Text Transfer Protocol over Secure Socket Layer）协议采用了加密机制来传输数据，但仍然对其有许多攻击方法，如伪造证书、SSL Strip 攻击。因此防范 HTTP 中间人攻击还是要从防范形成中间人攻击的手段入手，如 ARP 欺骗。同时，以可以在浏览器上使用黑白名单、网址验证等方法来检查网页的正确性，以提醒用户当前的安全状态。

7.3　ARP 欺骗实验

7.3.1　实验目的

ARP 欺骗实验要求了解 ARP 欺骗的原理，掌握 ARP 欺骗的实现过程。

7.3.2　实验内容及环境

1．实验内容

ARP 欺骗实验通过 Cain 工具实现 ARP 欺骗，帮助读者验证和掌握 ARP 欺骗的原理。

2．实验环境

宿主机为被欺骗主机，其操作系统为 Windows 7 SP1，32 位；

虚拟机为攻击主机，其操作系统为 Windows XP SP3，32 位。

实验工具：

WinPcap 4.1.3：详见本书 5.5 节实验工具介绍。

Cain v4.9（简写为 Cain）：用于 Windows 环境下的 ARP 欺骗、网络嗅探等功能。

7.3.3　实验步骤

1．环境准备

（1）将宿主机连入互联网，并将宿主机实体网卡的 IP 地址配置为 10.104.171.141，网关 IP 地址配置为 10.104.171.1。

（2）在宿主机上安装虚拟机 VMware 10.0.0，网络设置为桥接模式，虚拟机网卡 IP 地址配置为 10.104.171.133，网关 IP 地址配置为 10.104.171.1。

（3）在虚拟机上安装 WinPcap 4.1。

2．运行 Cain 主程序

在虚拟机上运行 Cain 主程序，单击“配置”菜单，在其对话框中选择 IP 地址为 10.104.171.133 的网卡适配器，如图 7.3 所示。

图 7.3　配置嗅探网卡

3．扫描活动主机

单击主界面的“嗅探器”标签，在下方的标签中单击“主机”，如图 7.4 所示。

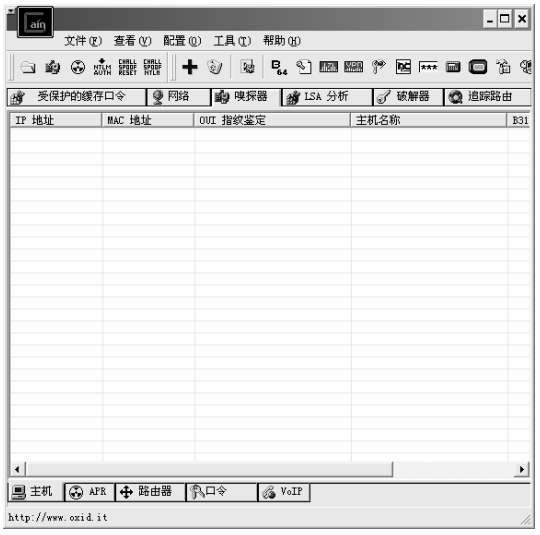

图 7.4　Cain 的界面

单击上方的“开始/停止嗅探”按钮，在空白处以鼠标右键单击，弹出菜单，选择“扫描 MAC 地址”，扫描完毕后，屏幕显示当前局域网中所有活动主机的 IP 地址和 MAC 地址列表，如图 7.5 所示。

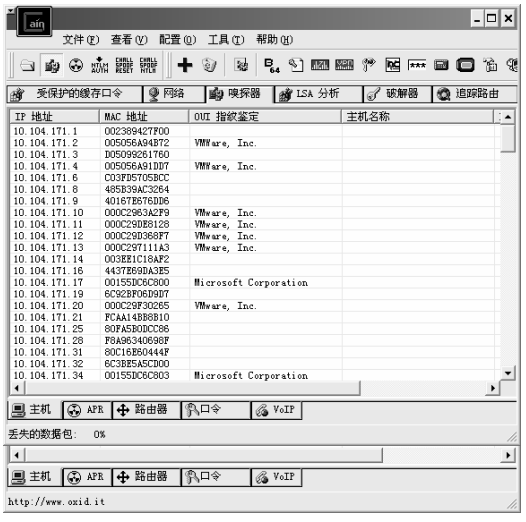

图 7.5　MAC 的扫描结果

4．配置信息

单击主界面的“嗅探器”标签，在下方的标签中单击“ARP”，配置 ARP 界面如图 7.6 所示。

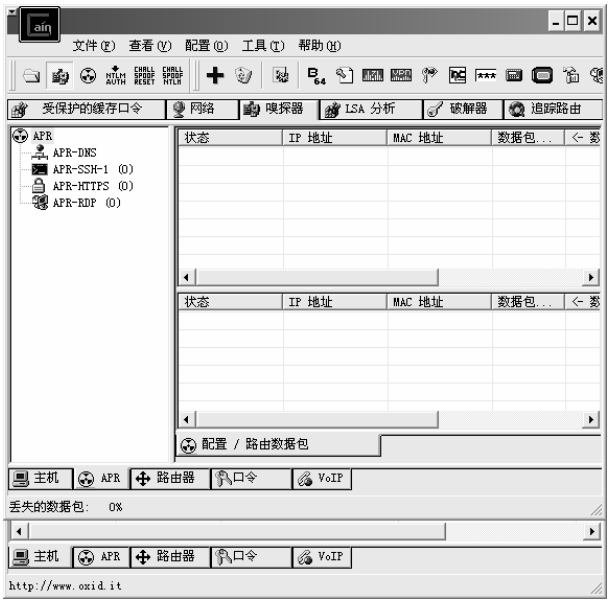

图 7.6　配置 ARP 界面

5．添加欺骗对象

在右上列表空白处单击鼠标左键，然后单击上方标签的“+”按钮，在弹出的对话框中的左列选择网关 IP 地址“10.104.171.1”，在对话框右列选择宿主机 IP 地址“10.104.171.141”，单击“确定”按钮，如图 7.7 所示。

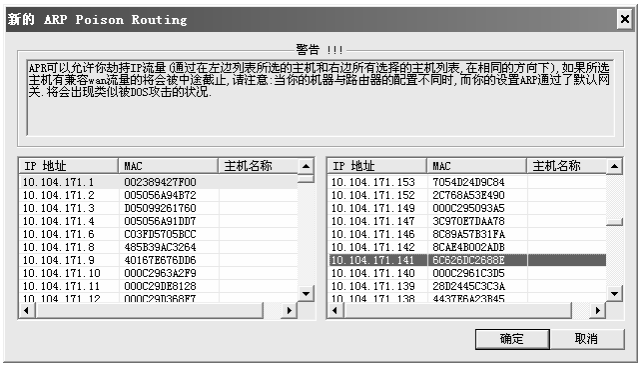

图 7.7　设置 ARP 欺骗的双方地址

6．开始 ARP 欺骗

单击主界面上方的“开始/停止 ARP”按钮，在其右上方列表显示的“Poisoning”状态，表示正在对宿主机和网关进行 ARP 欺骗，同时右下方列表会实时显示截获的通信数据条目，如图 7.8 所示。

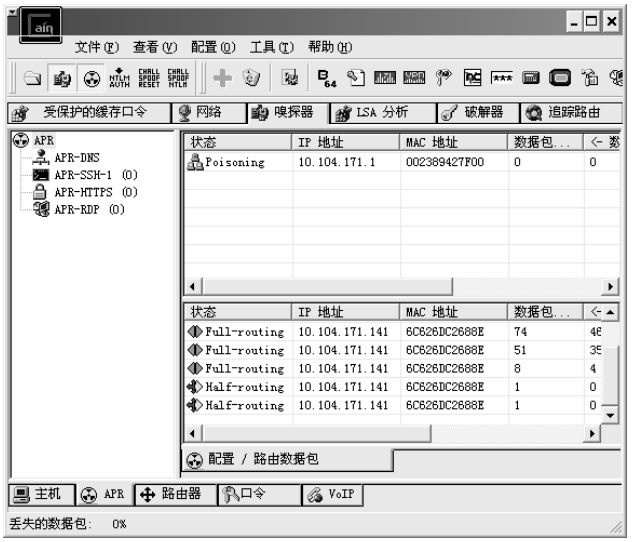

图 7.8　开始 ARP 欺骗

7．观察 ARP 缓存

为进一步了解 ARP 欺骗原理，在虚拟机上运行“cmd.exe”命令行程序，输入“ipconfig/all”命令，查看网卡的 IP 和 MAC 地址信息，可看到虚拟机的 MAC 地址是“00-0c-29-ea-dd-df”，如图 7.9 所示。

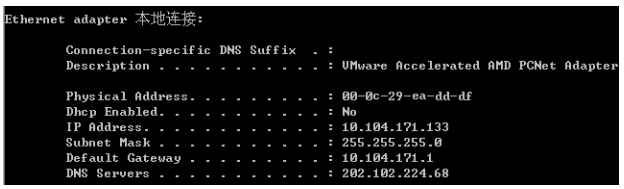

图 7.9　虚拟机的 MAC 地址

在宿主机上运行“cmd.exe”命令行程序，输入“arp -a”命令，然后查看当前的 ARP 缓存，可以看到网关 IP 地址“10.104.171.1”和虚拟机 IP 地址“10.104.171.133”所对应的 MAC 地址都是“00-0c-29-ea-dd-df”，如图 7.10 所示。

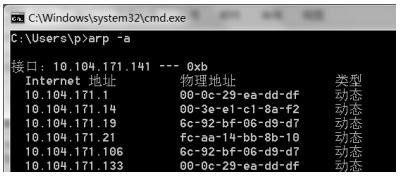

图 7.10　宿主机的 ARP 缓存

由此说明宿主机的 ARP 缓存已被欺骗，所有发给网关的数据都会被发给虚拟机 10.104.171.133。

此时虚拟机已经成功实现了 ARP 欺骗攻击，同时欺骗了网关和宿主机的 ARP 缓存，

使双方都认为对方的 MAC 地址是虚拟机的 MAC 地址。虚拟机成为了宿主机和网关通信的“中间人”，它们之间所有的通信数据都被虚拟机截获并转发。单击主界面下方的“口令”标签，可以看到被截获的实现多种协议传输的敏感信息，如 FTP、HTTP、SMTP 等。

7.3.4　实验要求

使用 Cain 工具完成 ARP 欺骗，并观察宿主机和网关的 ARP 缓存。

7.4　DNS 欺骗实验

7.4.1　实验目的

DNS 欺骗实验要求了解 DNS 欺骗的原理，掌握 DNS 欺骗的实现过程。

7.4.2　实验内容及环境

1．实验内容

DNS 欺骗实验通过 Cain 工具实现 DNS 欺骗，使读者能够验证和掌握 DNS 欺骗原理。

2．实验环境

宿主机为被欺骗主机，其操作系统为 Windows 7 SP1，32 位；
虚拟机为攻击主机，其操作系统为 Windows XP SP3，32 位。

实验工具：

WinPcap 4.1.3：详见本书 5.5 节的实验工具介绍。
Cain v4.99（简写为 Cain）：详见本书 7.3 节实验工具介绍。

7.4.3　实验步骤

1．重复实验 7.3

完成实验 7.3 后，即实现了 ARP 欺骗。

2．配置欺骗网址

在 ARP 界面单击其左侧列表中的“ARP-DNS”项，如图 7.11 所示。

在图 7.11 界面所示的右边空白处单击鼠标右键，选择“添加到列表”项，在弹出对话框中的“请求的 DNS 名称”文本框填入待欺骗网址“www.baidu.com”，再单击“解析”按钮，输入重定向网址“www.163.com”，单击“确定”按钮后，“www.163.com”所对应的 IP 地址自动填入对话框中，如图 7.12 所示。

单击“确定”按钮，在图 7.11 所示界面中右侧的列表多了一项内容，如图 7.13 所示。

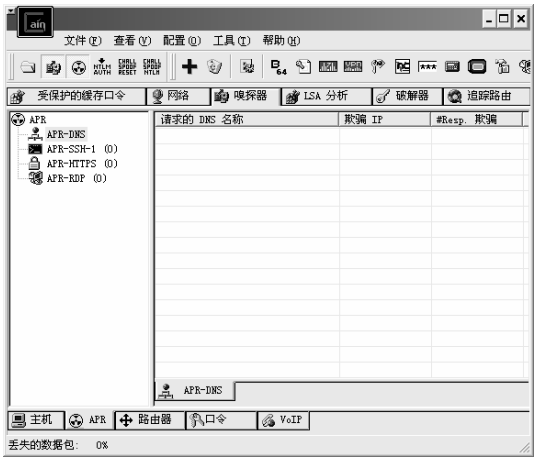

图 7.11　Cain 工具的 DNS 界面

图 7.12　设置虚假的 DNS 响应包

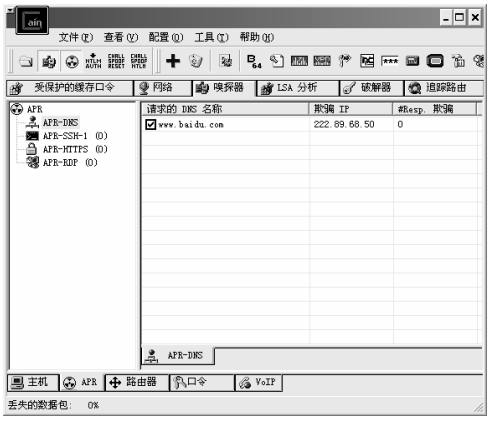

图 7.13　配置 DNS 重定向信息

此时虚拟机已对宿主机做好了将“www.baidu.com”域名重定向到“www.163.com”的准备。

3．开始 DNS 欺骗

为防止 DNS 缓存对实验的干扰，在宿主机上运行“cmd.exe”命令行程序，输入“ipconfig/flushdns”命令清空 DNS 缓存。接着打开浏览器，输入“www.baidu.com”网址，可以看到打开的是“www.163.com”网站，说明 DNS 欺骗成功，如图 7.14 所示。

图 7.14　DNS 欺骗成功

7.4.4　实验要求

使用 Cain 完成 DNS 欺骗，使得访问指定域名网站时被重定向到其他网站。

7.5　HTTP 中间人攻击实验

7.5.1　实验目的

HTTP 中间人攻击实验用于了解 HTTP 中间人攻击原理，使读者掌握 HTTP 中间人攻击的实现方法。

7.5.2　实验内容及环境

1．实验内容

HTTP 中间人攻击实验是通过 Mitmproxy 工具修改 Response 包的“Location”字段来重定向网页，以实现 HTTP 中间人攻击，使读者可验证和掌握 HTTP 中间人攻击的原理。

2．实验环境

宿主机为被欺骗主机，其操作系统为 Windows 7 SP1，32 位；

虚拟机为攻击主机，其操作系统为 Ubuntu 12.04 LTS 版，32 位。

实验工具：

Mitmproxy 0.12.1：该工具为 python 语言编写的 HTTP 协议中间人工具，可以拦截和修改 HTTP 数据包内容并进行转发。

7.5.3　实验步骤

1．环境准备

（1）先将宿主机连入互联网，然后将宿主机实体网卡的 IP 地址配置为 10.104.171.141，网关 IP 地址配置为 10.104.171.1。

（2）在宿主机上安装虚拟机 VMware 10.0.0，其网络设置为 Bridged 模式，虚拟机网卡的 IP 地址配置为 10.104.171.133，网关 IP 地址配置为 10.104.171.1。

2．安装 Mitmproxy

在虚拟机上运行“sudo pip install mitmproxy”命令，安装 Mitmproxy，其 pip 程序会自动下载最新版本的 Mitmproxy。如果事先没有安装 pip 程序，可以通过“sudo apt-get install python-pip”来安装。在安装 Mitmproxy 的过程中可能需要手动安装一些依赖包，具体可以查看 Mitmproxy 的安装说明文档，这里不再赘述。

3．输入拦截条件

安装成功后，运行 Mitmproxy 的命令，打开主界面。按“i”键设置要拦截的条件，输入“g\.cn”，表示如果 URL 中出现“g.cn”则拦截数据包，如图 7.15 所示。

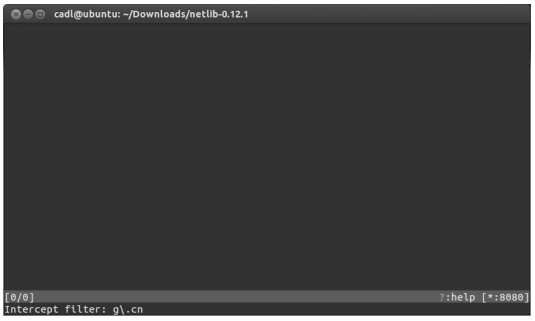

图 7.15　设置 Mitmproxy 的拦截条件

4．设置 IE 浏览器的代理服务器

按“回车”键确定拦截条件。注意此时 Mitmproxy 已经默认打开了 8080 端口等待连接。在宿主机上打开 IE 浏览器，设置其代理服务器地址为“10.104.171.133”，端口为 8080，如图 7.16 所示。

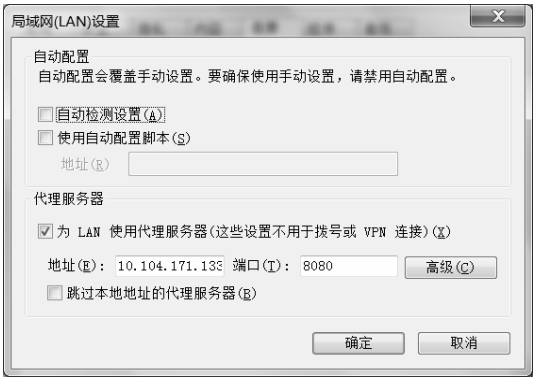

图 7.16　设置 IE 浏览器的代理服务器

5．拦截 GET 请求包

在 IE 浏览器的地址栏输入“www.g.cn”并按“回车”键，观察到 Mitmproxy 已经拦截到 IE 浏览器发送的数据包，方式为“GET”，URL 字段为“http://www.g.cn”，显示为红色，如图 7.17 所示。

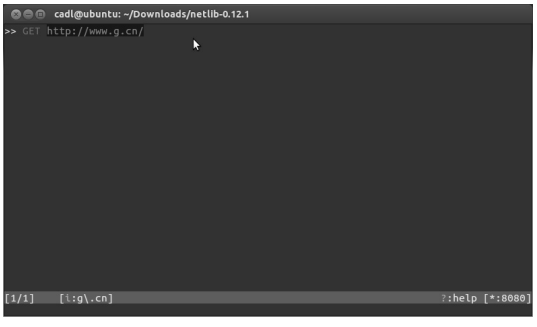

图 7.17　拦截 GET 请求包

单击 URL 字段，可以看到有三个标签，其中 Request 标签被标注为 intercepted，表明 Request 包已被拦截，且还能够看到请求头部的信息，如图 7.18 所示。

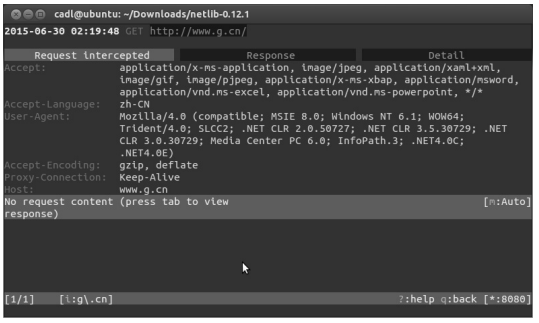

图 7.18　GET 请求包的内容

6．拦截 Response 包

这里不对 Request 包进行修改，因此按“a”键放行该包，很快会看到“www.g.cn”网站服务器返回的 Response 包被拦截下来，返回代码为 301，表示需要重定向到其他页面，由响应头部的 Location 字段指定，可见服务器要求重定向到“www.Google.cn”，如图 7.19 所示。

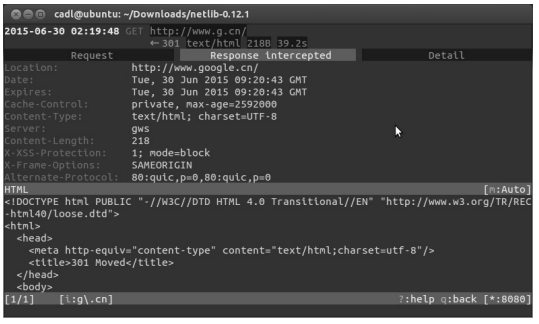

图 7.19　拦截 Response 包

7．修改 Location

尝试修改 Response 包里的内容。这里选择对“Location”进行修改。按“e”键，再按“h”键，修改响应头部“Location”的信息，如图 7.20 所示。

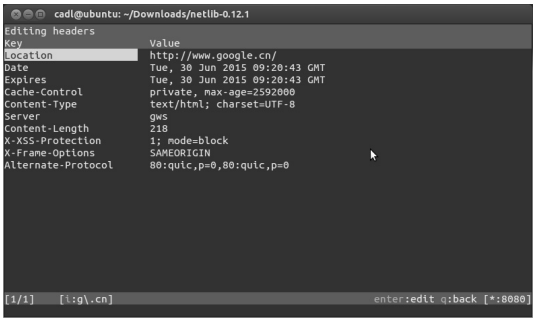

图 7.20　修改“Location”的界面

在“Location”一行按“回车”键，将“http://www.Google.cn”修改为“http://www.163.com”，按“Esc”键返回，如图 7.21 所示。

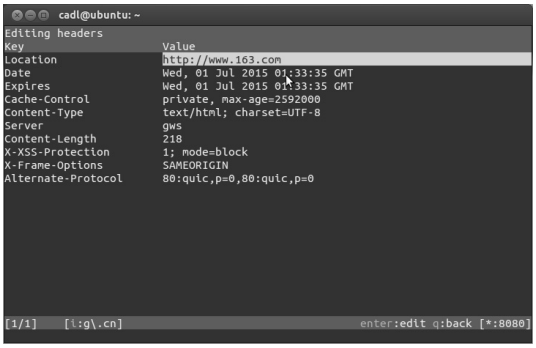

图 7.21　完成对“Location”的修改

8．完成 HTTP 中间人攻击

再按“q”键回到上级界面，按“a”键放行修改后的 Response 包。回到 IE 浏览器，发现原本应该打开的 Google 主页变成了“www.163.com”的主页，HTTP 中间人攻击成功。

7.5.4　实验要求

使用 Mitmproxy 工具拦截浏览器与服务器之间的 HTTP 请求包和响应包，修改“Location”数据，完成 HTTP 中间人攻击。

本章小结

假消息攻击利用了网络协议的弱点，通过篡改数据包的内容达到拒绝服务、窥探隐私等目的。本章先介绍了假消息攻击的原理，接着通过 ARP 欺骗实验，使读者了解和掌握 ARP 欺骗的原理和应用；通过 DNS 欺骗实验，使读者了解和掌握 DNS 欺骗的原理和应用；通过 HTTP 中间人攻击实验，使读者了解和掌握 HTTP 中间人攻击的原理和应用。

问题讨论

1. 在 7.3 节 ARP 欺骗实验中，还可以利用 Cain 工具在 ARP 欺骗的基础上，实现对用户名和口令的嗅探，请通过实验完成。

2. 在 7.5 节 HTTP 中间人攻击实验中，请实验验证还可以对 HTTP 协议结构的哪些内容进行篡改，达到什么效果？