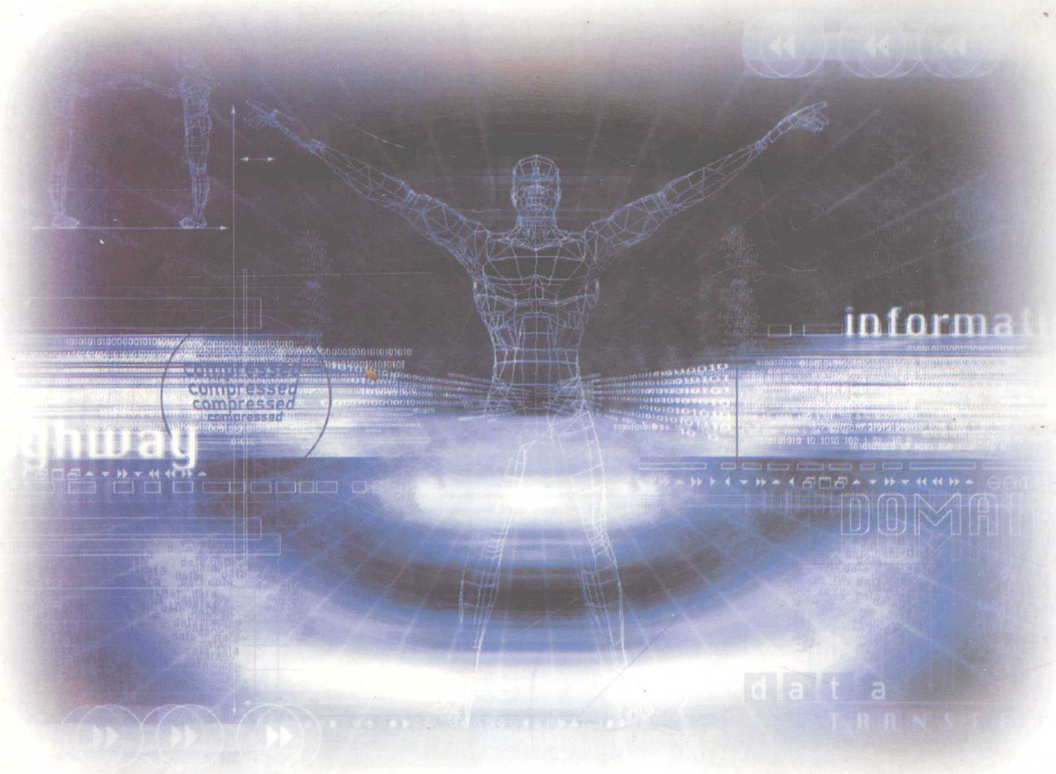
计算机网络实用技术系列

维护网络安全

代　伟 主编



国防工业出版社

计算机网络实用技术系列

维护网络安全

代伟 主编

国防工业出版社

·北京·

JISUANJI WANGLUO SHIYONG JISHU XILIE

内 容 简 介

该书全面地讲述了计算机网络安全与通信协议、操作系统以及相关安全问题、密码破解、黑客的后门程序和炸弹、入侵者的攻击方式、计算机病毒、防火墙安全问题的防范等网络安全知识与方法，旨在帮助非专业人员了解安全领域中相关各个方面的知识，建立安全意识，把握安全的衡量准则，最终提高信息系统的整体水平。

图书在版编目(CIP)数据

维护网络安全/代伟主编.—北京：国防工业出版社，2002.1
(计算机网络实用技术系列)
ISBN 7-118-02679-4

Ⅰ.维… Ⅱ.代… Ⅲ.计算机网络－安全技术
Ⅳ.TP393.08

中国版本图书馆 CIP 数据核字(2001)第 073685 号

国防工业出版社出版发行
(北京市海淀区紫竹院南路 23 号)
(邮政编码 100044)
北京奥隆印刷厂印刷
新华书店经售
*
开本 787×1092 1/16 印张 22 502 千字
2002 年 1 月第 1 版 2002 年 1 月北京第 1 次印刷
印数：1—4000 册 定价：29.00 元

前　言

以 Internet 为代表的全球性信息化浪潮日益深刻，信息网络技术的应用正日益普及和广泛，应用层次正在深入，应用领域从传统的、小型业务系统逐渐向大型、关键业务系统扩展，典型的如党政部门信息系统、金融业务系统、企业商务系统等。伴随网络的普及，安全日益成为影响网络效能的重要问题，而 Internet 所具有的开放性、国际性和自由性在增加应用自由度的同时，对安全提出了更高的要求。

开放性的网络，导致网络的技术是全开放的，任何一个人、团体都可能获得，因而网络所面临的破坏和攻击可能是多方面的，可以是来自物理传输线路的攻击，也可以是对网络通信协议和实现实施攻击；可以是对软件实施攻击，也可以是对硬件实施攻击。比如：通过在网络上监听获取网上用户的帐号和密码，监听密钥分配过程，攻击密钥分配服务器；利用 Unix 操作系统提供的 Daemon 和利用 FTP 采用匿名用户访问进行攻击，等等。

国际性的一个网络还意味着网络的攻击不仅仅来自本地网络的用户，它可以来自 Internet 上的任何一台机器，也就是说，网络安全所面临的是一个国际化的挑战。

自由意味着网络最初对用户的使用并没有提供任何的技术约束，用户可以自由地访问网络，自由地使用和发布各种类型的信息。用户只对自己的行为负责，而没有任何的法律限制。

尽管，开放的、自由的、国际化的 Internet 的发展给政府机构、企事业单位带来了革命性的改革和开放，使得他们能够利用 Internet 提高办事效率和市场反应能力，以便更具竞争力。通过 Internet，他们可以从异地取回重要数据，同时又要面对 Internet 开放带来的数据安全的新挑战和新危险。如何保护企业的机密信息不受黑客和工业间谍的入侵，已成为政府机构、企事业单位信息化健康发展所要考虑的重要事情之一。

由于我国网络研究起步晚，网络安全技术还有待整体的提高和发展。面对日益严重的安全问题，我们应该如何去认识、去分析、去防范，是我们当前每一个投身于 Internet 的人所面临的一个迫切的问题，本书即是从这个目的出发，希望通过对安全问题的一些探讨和对网络入侵(黑客)技术的分析，来提高广大读者朋友在网络及整个系统安全方面的意识。

全书由代伟主编，周小明、欧阳黎明、杜建中、耿小宇、周正、谭永军、宋辉、刘朋、刘敏、葛好华、秦水勇、王玲玲、岳本华、舒伟、孙宝静、马华、丁丽、刘菲菲、赵雅鹃、郑浩、胡海东等同志参与了本书的编写工作。在此谨向所有为本书的出版作出贡献的人士表示感谢。

由于网络安全技术和其中所涉及的黑客技术都要涉及很多很深的领域，加上作者水平有限，不当之处在所难免，邀请各位读者不吝指正。

编　者

目　录

第1章　网络安全简介 …… 1
1.1　网络的安全性 …… 1
1.1.1　Internet 的不安全因素 …… 1
1.1.2　网络系统安全管理措施 …… 2
1.1.3　网络安全工具 …… 2
1.2　黑客与网络安全 …… 3
1.2.1　黑客发展简史 …… 3
1.2.2　黑客眼中的黑客 …… 8
1.2.3　Hacker 与 Cracker …… 9
1.2.4　今日黑客 …… 11
第2章　计算机网络安全与通信协议 …… 12
2.1　TCP/IP 协议简介 …… 12
2.1.1　TCP/IP 协议以及工作原理 …… 12
2.1.2　以太网 …… 14
2.2　网络协议以及它们可能存在的安全问题 …… 15
2.2.1　地址解析协议 ARP …… 15
2.2.2　Internet 控制消息协议 ICMP …… 18
2.2.3　IP 协议与路由 …… 18
2.2.4　TCP 协议 …… 19
2.2.5　Telnet 协议 …… 20
2.2.6　文件传输协议 FTP …… 21
2.2.7　简单电子邮件传输协议 SMTP …… 23
2.2.8　超连接传输协议 HTTP …… 23
2.2.9　网络新闻传输协议 NNTP …… 26
2.3　WWW 安全 …… 26
2.3.1　CGI 程序的安全 …… 26
2.3.2　Active X 的安全性 …… 29
2.3.3　Cookies 安全 …… 30
2.3.4　SSL 加密安全性 …… 30
2.3.5　Plugin 安全性 …… 31
2.3.6　电子邮件的安全 …… 31
2.3.7　WWW 服务器安全的一个实例 …… 33
2.4　Java 和 Java applet …… 38

2.4.1 Java 的优势和安全隐患 …… 38
2.4.2 Java 的安全机制 …… 40
2.4.3 安全使用的原则 …… 42
2.5 WWW 的欺骗攻击和防御 …… 43
2.5.1 欺骗攻击 …… 43
2.5.2 安全决策 …… 44
2.5.3 暗示 …… 44
2.5.4 Web 欺骗 …… 44
2.5.5 对 WWW 欺骗的防御措施 …… 47
第 3 章 操作系统以及相关安全问题 …… 49
3.1 Unix 管理员的安全管理 …… 49
3.1.1 安全管理 …… 49
3.1.2 超级用户 …… 49
3.1.3 文件系统安全 …… 49
3.1.4 作为 root 运行的程序 …… 53
3.1.5 /etc/passwd 文件 …… 55
3.1.6 /etc/Group 文件 …… 56
3.1.7 安全检查 …… 56
3.1.8 加限制的环境 …… 59
3.1.9 小系统安全 …… 60
3.1.10 物理安全 …… 61
3.1.11 用户意识 …… 61
3.1.12 系统管理员意识 …… 62
3.1.13 Unix 的子程序 …… 65
3.2 Sun 工作站的安全实例 …… 72
3.2.1 安装系统补丁程序(Patch) …… 72
3.2.2 采用最新版本的服务方软件 …… 72
3.2.3 口令安全 …… 73
3.2.4 文件目录权限 …… 73
3.2.5 限制网络用户对系统的访问 …… 73
3.2.6 关闭不必要的服务端口 …… 73
3.2.7 定期对服务器进行备份 …… 73
3.2.8 设置系统日志 …… 74
3.2.9 定期检查系统安全性 . …… 74
3.3 Unix 的用户安全 …… 75
3.3.1 口令安全 …… 75
3.3.2 文件许可权 …… 75
3.3.3 目录许可 …… 76
3.3.4 umask 命令 …… 76

3.3.5 设置用户 ID 和同组用户 ID 许可 …… 76
3.3.6 cp mv ln 和 cpio 命令 …… 76
3.3.7 su 和 newgrp 命令 …… 77
3.3.8 文件加密 …… 78
3.3.9 其他安全问题 …… 78
3.3.10 保持户头安全的要点 …… 80
3.4 uucp 系统与其他 Unix 网络 …… 81
3.4.1 uucp 系统概述 …… 81
3.4.2 uucp 系统的安全问题 …… 82
3.4.3 Honeydanber uucp …… 84
3.4.4 其他网络 …… 89
3.4.5 通信安全 …… 90
3.4.6 Sun OS 系统的网络安全 …… 92
3.5 Unix 系统安全评估和监测工具 …… 99
3.5.1 配置管理 …… 99
3.5.2 网络访问 …… 100
3.5.3 口令管理 …… 100
3.5.4 锁屏 …… 101
3.6 Windows NT 操作系统简介 …… 101
3.6.1 Windows NT 系统的安全概述 …… 102
3.6.2 Windows NT 系统中的术语 …… 103
3.6.3 Windows NT 操作环境 …… 107
3.6.4 Windows NT 系统登录和认证 …… 113
3.6.5 Microsoft Internet Information Server …… 115
3.6.6 Microsoft 代理服务器 …… 118
3.6.7 Windows NT 目录服务模型 …… 119
3.6.8 Windows NT 文件系统安全性 …… 120
3.7 Windows NT 操作系统的安全漏洞和解决办法 …… 120
3.7.1 概况 …… 120
3.7.2 Windows NT 服务器和工作站的安全漏洞 …… 121
3.7.3 与浏览器和 Windows NT 计算机有关的安全漏洞 …… 128
3.7.4 其他安全必要的措施 …… 129
3.7.5 WWW 上的 Windows NT 系统 …… 131
3.8 对 Windows NT 安全性的评估和监测工具 …… 136
3.8.1 Enterprise Administrator …… 136
3.8.2 Internet Security Systems …… 136
3.8.3 RADIUS …… 136
第 4 章 破解 …… 137
4.1 口令破解器 …… 137

4.2 口令破解器的工作原理 …… 138
4.2.1 Unix 口令破解简介 …… 138
4.2.2 Windows 95 屏幕保护口令密码破解简介 …… 141
4.3 破解工具 …… 145
4.3.1 John the Ripper 简介 …… 145
4.3.2 John 程序的解密模式 …… 148
4.3.3 Soft ICE 的使用方法 …… 151
4.4 注册码破解 …… 153
4.4.1 破解 WinZip 8.0 …… 153
4.4.2 破解 MP3 Explorer …… 153
4.5 跟踪与反跟踪 …… 157
4.5.1 跟踪技术 …… 157
4.5.2 反跟踪技术 …… 158
4.5.3 反跟踪技术主要采用的方法 …… 158
第 5 章 黑客的后门程序和炸弹 …… 167
5.1 特洛伊木马 …… 167
5.1.1 特洛伊木马基础知识 …… 167
5.1.2 Back Orifice …… 171
5.1.3 NetBus …… 176
5.2 特洛伊木马的编制原理 …… 177
5.2.1 ExitWindowsEx 函数介绍 …… 177
5.2.2 服务器程序 …… 177
5.2.3 客户端程序 …… 181
5.3 后门的清除和防御 …… 184
5.3.1 NetSpy 及防范措施 …… 184
5.3.2 一些常见的后门所开放的端口 …… 187
5.4 炸弹 …… 191
5.4.1 E – mail 炸弹 …… 191
5.4.2 浏览器炸弹 …… 193
5.4.3 OICQ 炸弹和 OICQ 安全 …… 194
第 6 章 入侵者的攻击方式 …… 196
6.1 攻击的一般步骤 …… 196
6.1.1 对 Unix 系统的攻击 …… 196
6.1.2 对 Windows NT 系统的攻击 …… 203
6.2 缓冲区溢出及其攻击 …… 205
6.2.1 缓冲区溢出原理 …… 205
6.2.2 制造缓冲区溢出 …… 205
6.2.3 通过缓冲区溢出获得用户 Shell …… 208
6.2.4 利用缓冲区溢出进行的系统攻击 …… 213

6.2.5 缓冲区溢出应用攻击实例 …… 215
6.3 IP 欺骗及其原理 …… 216
6.3.1 IP 欺骗 …… 219
6.3.2 使被信任主机丧失工作能力 …… 219
6.3.3 序列号取样和猜测 …… 219
6.3.4 IP 欺骗的防止 …… 220
6.3.5 抛弃基于地址的信任策略 …… 221
6.3.6 降低 TCP/IP 欺骗的危险 …… 227
6.4 Sniffer …… 227
6.4.1 HUB 和网卡的工作原理 …… 227
6.4.2 Sniffer …… 228
6.4.3 关于接收模式 …… 230
6.4.4 如何发现 Sniffer …… 230
6.5 扫描和监听 …… 231
6.5.1 几个常用相关网络命令 …… 231
6.5.2 端口扫描器 …… 237
6.5.3 扫描器的编写 …… 239
第 7 章 计算机病毒 …… 245
7.1 计算机病毒历史 …… 245
7.2 计算机病毒原理 …… 247
7.3 计算机病毒防范 …… 249
7.4 计算机病毒实例 …… 250
7.4.1 CIH 病毒 …… 250
7.4.2 Word 宏病毒透视 …… 253
7.4.3 Melissa 病毒及源代码 …… 255
第 8 章 防火墙 …… 259
8.1 防火墙概述 …… 260
8.1.1 防火墙的概念 …… 260
8.1.2 防火墙的功能 …… 261
8.1.3 防火墙的优缺点 …… 262
8.2 防火墙的体系结构 …… 263
8.2.1 防火墙的组成 …… 263
8.2.2 防火墙的结构 …… 265
8.3 实现防火墙的技术 …… 267
8.3.1 应用代理服务器 …… 267
8.3.2 回路级代理服务器 …… 267
8.3.3 代管服务器 …… 268
8.3.4 IP 通道 …… 268
8.3.5 网络地址转换器 …… 268

8.3.6 隔离域名服务器 …… 269
8.3.7 电子邮件转发技术 …… 269
8.4 制作防火墙的方法 …… 269
8.4.1 包过滤型 …… 269
8.4.2 代理型 …… 270
8.4.3 监测型 …… 271
8.5 维护防火墙 …… 271
8.5.1 日常管理 …… 271
8.5.2 系统监控 …… 273
8.5.3 不断更新 …… 277
8.6 防火墙产品大观 …… 279
8.6.1 Checkpoint Firewall.1 …… 279
8.6.2 Sonicwall 系列防火墙 …… 280
8.6.3 NetScreen Firewall …… 280
8.6.4 Alkatel Internet Devices 系列防火墙 …… 281
8.6.5 北京天融信公司网络卫士防火墙 …… 281
8.6.6 NAI Gauntlet 防火墙 …… 282
8.7 Internet 网络监视器 …… 282
8.7.1 概述 …… 282
8.7.2 功能与作用 …… 283
8.7.3 因特网服务在防火墙中的配置及实例 …… 283
8.8 防火墙实例 …… 299
8.8.1 子网过滤结构的防火墙 …… 299
8.8.2 被屏蔽主机体系结构 …… 307
8.9 TIS 防火墙工具箱 …… 311
8.9.1 理解 TIS 防火墙工具箱 …… 311
8.9.2 在不同操作系统下的编译方法 …… 311
8.9.3 安装 TIS 代理服务器 …… 319
8.9.4 设置 TIS FWTK …… 320
第 9 章 安全问题的防范 …… 327
9.1 如何发现漏洞 …… 327
9.1.1 心理学上问题 …… 327
9.1.2 变换角色漏洞 …… 327
9.1.3 缓冲区溢出漏洞 …… 328
9.1.4 吞噬资源漏洞 …… 329
9.1.5 信赖未经确信的信道漏洞 …… 330
9.1.6 错误的缺省设定漏洞 …… 330
9.1.7 大接口漏洞 …… 330
9.1.8 薄弱的安全部件 …… 331

9.1.9 被忽略的情况 …… 331
9.1.10 低级错误 …… 331
9.2 Log日志文件 …… 331
9.2.1 Unix系统的Log日志文件 …… 332
9.2.2 Windows NT的审计跟踪 …… 333
9.3 跟踪入侵者 …… 333
9.3.1 通信过程的记录设定 …… 334
9.3.2 记录的位置 …… 335
9.3.3 跟踪地理位置 …… 337
9.3.4 来话者电话侦测(Caller ID) …… 337
9.3.5 靠IP地址或Domain Name找出入侵者位置 …… 337
9.3.6 Domain Name命名的三种情况 …… 338
9.3.7 由Domain Name查出连接单位信息 …… 338
9.3.8 只有IP地址的查找方法 …… 339

第 1 章　网络安全简介

1.1　网络的安全性

互联网络发展到今天，大多已成为了 Internet 的一部分。Internet 的有偿使用趋势，使互联网的安全性显得尤其重要。这里主要就下面几个问题进行讨论。

1.1.1　Internet 的不安全因素

Internet 本来就不安全，因为最初的 Internet 建设者们认为安全不是问题。

Internet 在其早期是一个开放的为研究人员服务的网际网，是完全非赢利性的信息共享载体，所以几乎所有的 Internet 协议都没有考虑安全机制。这点从 Internet 上最通用的应用如 FTP、Telnet 和电子邮件中的用户口令的明文传输以及 IP 报文在子网段上的广播传递就可以充分地体现出来。只是近些年来，Internet 的性质和使用人员的情况发生了很大的变化，使得 Internet 的安全问题显得越来越突出。随着 Internet 的全球普及和商业化，用户越来越多，并且非常私人化，一些私人信息如信用卡号等和其自身利益相关的信息也通过 Internet 传输，而且越来越多的信息放在网上是为了赢利，而不是完全免费的信息共享，所以其安全性也成为人们日趋关注的问题。

Internet 不安全的再一个因素是因为人们很容易从 Internet 上获得相关的核心技术资料，特别是有关 Internet 自身的技术资料，比如 RFC、FAQ 文件、各类应用程序源代码，如 TCP/IP、Sendmail、FTP 等，还有各类安全工具的源代码也是公开免费的，像颇有争议的 SATAN、Crack 等。这些资料拿出来共享的愿望是好的，但也难免产生事与愿违的效果。

Internet 不安全的另一个致命因素是使用者普遍缺乏安全意识，特别是那些对计算机和 Internet 技术不了解的用户。很少用户会去读 RFC1244 安全手册，或关注 CERT 安全组织提出的忠告。对大多数用户来说，能管好自己的密码就万事大吉了，但又有多少用户愿意取一个不好记的密码呢？方便性和安全性总是相互冲突，很难兼得的。

说来说去，Internet 不安全归根到底还是因为人自己，只要看一看那越来越厚的防盗门就明白了。如今，Internet 上也出现了比防盗门更复杂的防火墙(Firewall)来防范那些不怀好意或那些只是为了逞能和好奇的的黑客们，还出现了各种各样的私人密钥(private key)或公共密钥(public key)来保护在网上传送的信息。

Internet 不安全是一个不可回避的现实，必须正视这个现实，下面谈谈可以采取的安全措施。

1.1.2 网络系统安全管理措施

网上大部分的攻击是针对网络上的服务器系统，其中包括电子邮件、匿名 FTP、WWW、DNS、News 等服务系统。这些系统大都是运行在 Unix 系统上的，其中有 Sun 的 Solaris、SCOUnix、HPUX、SGI 的 IRIX、IBM 的 AIX 和 Linux 等。系统之所以易受攻击，有各方面的因素。首先是这些系统知名度高，容易引起注意。其次是系统本身存在漏洞。我们一方面可采用一些防卫性措施，另一方面，网络系统管理员可以应用一些工具，来查找系统安全漏洞，或从网上截获报文进行分析。

1.1.3 网络安全工具

1. COPS——系统安全检测

COPS 用来对 Unix 系统进行安全检查，它的主要检测目标有以下几点：

- 文件、目录和设备文件的权限检查。
- 重要系统文件的内容、格式和权限检查。
- 检查是否存在所有者为 root 的 SUID 文件。
- 对重要系统二进制文件进行 CRC 校验和检查，检查是否被修改过。
- 对匿名 FTP、Sendmail、tFTP 等网络应用进行检查。

值得一提的是，COPS 只是检测工具，并不做实际的修复。

2. 口令密码解破

Crack 是最著名的 Unix 系统上破解 Unix 口令的工具。Crack 的工作原理很简单：由于一般加密口令是不会被解开的，即加密算法是不可逆的，所以一般的口令入侵是通过生成口令进行加密去匹配原口令密码，或直接从网上截获明文口令。Crack 程序中包含了几个很大的字典库，进行解破时它会按照一定的规则将字词进行组合，然后对之进行加密，再与要解破的加密口令匹配。所以运行 Crack 通常要占用大量的 CPU 资源，并要运行相当长的时间才结束。

3. Sniffer——网络监听

Sniffer 这个词是指黑客或其他人通过一些软件来截获在网络上传送的数据包。常用的工具有 Traceroute、Snoop、Gobbler、TCP.dump、ethload、Netman、Sunsniff 等。防止 Sniffer 有两种办法，加密和分段。比如设置 SSH(Secure Shell)替换 rlogin 和 Telnet 等，这样可解决用户名和口令在网络上明文传输的问题。所谓分段，是根据分段越多，网络信息可靠性越高的原则来处理网络信息。因为 IP 报文只能在本子网段上广播。有些单位甚至直接将服务器和交换机联接来保证服务器的安全运行，但这样的做法开销太大，对较小的单位是不可行的。在后面的章节，有专门的部分讨论 Sniffer。

4. 防火墙——对付远程攻击

防火墙的共性是它们都有基于源地址基础上的区分或拒绝某些访问的能力。下面介绍几种具有防火墙功能的软件。目前使用较多的软件有下面两类：

- 数据包过滤工具：TCP _ Wrappers，NetGate；
- 应用代理和应用网关：Netscape Proxy，Socks，TIS.FWTK；

TCP _ Wrappers 是通过 IP 地址来控制对服务器的访问，它的主要配置文件有两个：

/etc/Hosts.allow和/etc/Hosts.deny。而 Netscape Proxy 则可以根据用户帐号来进行访问控制和记帐。

5. 扫描器——Scanner

扫描器是能够自动检测远地或本地主机安全性的程序。它通过扫描 TCP 端口,并记录反馈信息,从而发现网络的弱点,促使系统安全。常见的工具有:Host、Traceroute、Rusers、Finger、Showmount、NSS(网络安全扫描器)、Strobe(超级优化 TCP 端口检测程序)及 SATAN 等。

1.2 黑客与网络安全

1.2.1 黑客发展简史

黑客是计算机专业领域中的一个特殊的群体,随着目前新闻中计算机系统被攻击的报道越来越多,黑客就越发成为社会关注的焦点。关于黑客的描述有很多,在这里只简单地介绍其发展历史,这样可以更好的理解黑客这个概念,明白网络安全与他们之间的密切关系。

在介绍黑客之前,先要说说 Real Programmer。这些人从不自称是 Real Programmer、Hacker 或任何特殊的称号。Real Programmer 这个名词是在 20 世纪 80 年代才出现的,但早自 1945 年起,计算机科学便不断地吸引世界上头脑最顶尖、想像力最丰富的人投入其中。从 Eckert & Mauchly 发明 ENIAC 后,便不断有狂热的程序员(Programmer)投入其中,他们以撰写软件与玩弄各种程序设计技巧为乐,逐渐形成具有自我意识的一种科技文化。

当时,这批 Real Programmer 主要来自工程界与物理界,他们戴着厚厚的眼镜,穿聚酯纤维 T 恤与纯白袜子,用计算机语言、汇编语言、FORTRAN 及很多古老的语言写程序。他们是 Hacker 时代的先驱者,默默贡献,鲜为人知。从二次大战结束后到 1970 早期,是打卡计算机与所谓"大铁块"的 mainframes 流行的年代,由 Real Programmer 主宰计算机文化。

至今,一些 Real Programmer 仍在世且十分活跃。超级计算机 Cray 的设计者 Seymour Cray,据说亲手设计 Cray 全部的硬件及其操作系统,操作系统是他用计算机码"硬干"出来的,没有出过任何 bug 或错误。Stan KellyBootle, The Devil's DP Dictionary 一书的作者(McGraw.Hill,1981 年初版,ISBN 0.07.034022.6)是 Hacker 传奇专家,当年在一台 Manchester Mark I 开发程序。他现在是计算机杂志的专栏作家,写一些科学幽默小品,文笔生动有趣,投今日 Hacker 所好,所以很受欢迎。其他人像 David E. Lundstorm,写了许多关于 Real Programmer 的小故事,收录在 *A Few Good Men From UNIVAC* 这本书中。

看到这里,读者应该能了解,所谓 Real Programmer 指的就是用组合语言甚至计算机码,把程序用打卡机 Punch 出一片片纸卡片,再由主机读卡机输入计算机的那种"石器时代"的 Programmer。

随 Real Programmer 的时代步入尾声,取而代之的是逐渐盛行的 Interactive Comput-

ing,一些大学开始成立计算机相关科系及计算机网络专业。Hacker 时代的滥觞始于1961 年,MIT(麻省理工学院)出现第一台计算机 DEC PDP.1。MIT 的 TMRC(Tech Model Railroad Club)的 Power and Signals Group 买了这台计算机后,把它当成最时髦的科技玩具,各种程序工具与计算机术语开始出现,整个环境与文化一直发展下去直至今日。

这在 Steven Levy 的书 Hacker 前段有详细的记载(Anchor/Doubleday 公司,1984 年出版,ISBN 0.385.19195.2)。最先使用 Hacker 这个字应该是 MIT。20 世纪 80 年代早期学术界人工智能的权威,MIT 的 Artificial Intelligence Laboratory(AIL),其核心人物皆来自 TMRC。从 1969 年起,正好是 ARPANET 建置的第一年,这群人在计算机科学界便不断有重大突破与贡献。ARPANET 是第一个横跨美国的高速网络,由美国国防部出资兴建。它是一个实验性质的数位通信网络,后来逐渐成长成联系各大学、国防部承包商及研究机构的大网络。各地研究人员能以史无前例的速度与弹性交流信息,超高效率的合作模式导致了科技的突飞猛进。ARPANET 的另一项好处是,信息高速公路使得全世界的 Hacker 能聚在一起,不再像以前那样孤立在各地形成一股股的短命文化。网络把他们汇流成一股强大力量。这时候,开始有人感受到 Hacker 文化的存在,并动手整理术语放上网络,在网上发表讽刺文学与讨论 Hacker 所应有的道德规范(Jargon File 的第一版出现在 1973 年,就是一个好例子)。Hacker 文化在接有 ARPANET 的各大学间快速发展,特别是在信息相关科系。

一开始,整个 Hacker 文化的发展以 MIT 的 AI Lab 为中心,但 Stanford University 的 SAIL(Artificial Intelligence Laboratory)与稍后的 CMU(Carnegie Mellon University)正在快速崛起中。三个都是大型的信息科学研究中心及人工智能的权威,聚集着世界各地的精英,不论在技术上或精神层次上,对 Hacker 文化都有极高的贡献。

为能了解后来的故事,得先看看计算机本身的变化。随着科技的进步,主角 MIT AIL 也从红极一时到最后淡出舞台。从 MIT 那台 PDP.1 开始,Hacker 们的主要程序开发平台都是 Digital Equipment Corporation 的 PDP 迷用户计算机系列。DEC 率先发展出以商业用途为主的 Interactive Computing 及 Time Sharing 操作系统。当时许多的大学都是买 DEC 的计算机,因为它兼具弹性与速度,还很便宜。便宜的分时系统是 Hacker 文化能快速成长的因素之一。在 PDP 流行的时代,ARPANET 上是 DEC 计算机的天下,其中最重要的便属 PDP.10,PDP.10 受到 Hacker 们的青睐达 15 年,TOPS.10(DEC 的操作系统)与 MACRO.10(它的组译器),许多怀旧的术语及 Hacker 传奇中仍常出现这两个字。MIT 像大家一样用 PDP.10,但他们不屑用 DEC 的操作系统。他们偏要自己写一个:传说中赫赫有名的 ITS。ITS 全名是 Incompatible TimeSharing System,取这个怪名果然符合 MIT 的搞怪作风,就是要与众不同,他们有本事自己去写一套操作系统。

ITS 始终不稳,设计古怪,bug 也不少,但仍有许多独到的创见,似乎还是分时系统中开机时间最久的纪录保持者。ITS 本身是用汇编语言写的,其他部分用 LISP 写成。LISP 在当时是一个威力强大且极具弹性的程序语言。事实上,25 年后的今天,它的设计仍优于目前大多数的程序语言。LISP 让 ITS 的 Hacker 得以尽情发挥想像力与搞怪能力。LISP 是 MTI AIL 成功的最大功臣,现在它仍是 Hacker 们的最爱之一。很多 ITS 的产物到现在仍活着;EMACS 大概是最有名的一个,而 ITS 的野史仍为今日的 Hacker 们

所津津乐道。

在 MIT 红得发紫之际,SAIL 与 CMU 也没闲着。SAIL 的中坚分子后来成为 PC 界或图形使用者界面研发的主要角色。CMU 的 Hacker 则开发出第一个实用的大型专家系统与工业用机器人。

另一个 Hacker 重镇是 XEROX PARC 公司的 PARC(Palo Alto Research Center)。从 20 世纪 70 年代初期到 80 年代中这十几年间,PARC 不断出现惊人的突破与发明,不论质或量,软件或硬件方面。如现今的 Windows 鼠标界面,激光打印机与局域网络。其 D 系列的计算机,催生了能与迷用户计算机一较长短的强大的个人计算机。PARC 这群人对 Hacker 文化有着不可抹灭的贡献。

20 世纪 70 年代 ARPANET 与 PDP.10 文化迅速茁壮成长。Mailing list 的出现使世界各地的人得以组成许多 SIG(Special Interest Group),不只在计算机方面,也有社会与娱乐方面的。DARPA 对这些"非正当性"活动睁一只眼闭一只眼,因为靠这些活动会吸引更多的聪明小伙子们投入计算机领域呢。有名的非计算机技术相关的 ARPANET Mailing List 首推科幻小说迷,时至今日 ARPANET 变成了 Internet,愈来愈多的读者参与讨论。Mailing List 逐渐成为一种公众讨论的媒介,导致许多商业化上网服务如 CompuServe、Genie 与 Prodigy 的成立。

此时在新泽西州的郊外,另一股神秘力量积极入侵 Hacker 社会,终于席卷整个 PDP.10 的传统。它诞生在 1969 年,也就是 ARPANET 成立的那一年,有个在 AT&T 贝尔实验室工作的年轻小伙子 Ken Thompson 发明了 Unix。Thomspon 曾经参与 Multics 的开发,Multics 是源自 ITS 的操作系统,用来试验当时一些较新的 OS 理论,如把操作系统较复杂的内部结构隐藏起来,提供一个界面,使得 Programmer 能不用深入了解操作系统与硬件设备,也能快速开发程序。这样减轻了工作量和很多底层知识的理解量,使得更多的人能投入到 Programmer 的行列之中。

在发现继续开发 Multics 是在浪费时间之后,贝尔实验室很快退出了。Ken Thompson 很喜欢 Multics 上的操作环境,于是他在实验室里一台报废的 DEC PDP.7 上胡乱写了一个操作系统,该系统在设计上有从 Multics 抄来的也有他自己的构想。他将这个操作系统命名 Unix,用来反讽 Multics。另外一种传言说他设计此系统的原因是他写了一个名叫 Star Travel 的游戏,没地方运行,就去找一台报废的计算机 PDP.7 来玩。他同事 Brian Kernighan 嘲笑他说:"你写的系统好差,干脆叫 Unics 算了。"Unics 发音与太监的英文 eunuches 一样,后来才改为 Unix。这个操作系统就是当今大红大紫的网络操作系统。

他的同事 Dennis Ritchie,发明了一种新的程序语言——C 语言,于是他与 Thompson 用 C 语言把原来用汇编语言写的 Unix 重写一遍。C 语言的设计原则就是好用、自由与弹性,C 语言与 Unix 很快地在贝尔实验室受到欢迎。1971 年 Thompson 与 Ritchie 争取到一个办公室自动化系统的项目,于是 Unix 开始在贝尔实验室中流行。不过 Thompson 与 Ritchie 的雄心壮志还不止于此。那时的传统是,一个操作系统必须完全用汇编语言写成,始能让计算机发挥最高的效能。Thompson 与 Ritchie 是头几位掌握了硬件与编译器技术,已经进步到操作系统可以完全用高级语言如 C 语言来写,却能保有不错的效能的人。

五年后，Unix 已经成功地移植到了数种计算机上。Ken Thompson 与 Dennis Ritchie 也成为了惟一两位获得 Turing Award(计算机界的诺贝尔奖)的工程师(其他都是学者)。这在当时是一件不可思议的事！它意味着，如果 Unix 可以在各种平台上运行的话，Unix 软件就能移植到各种计算机上，再也用不着为特定的计算机写软件了。除了跨平台的优点外，Unix 与 C 语言还有许多显著的优势。Unix 与 C 语言的设计哲学是“Keep It Simple, Stupid”。Programmer 可以轻易掌握整个 C 语言的逻辑结构(不像其他之前或以后的程序语言)，而不用一天到晚翻手册写程序。而 Unix 提供许多有用的小工具程序，经过适当地组合(写成 Shell script 或 Perl script)，就可以发挥强大的威力。

C 语言与 Unix 的应用范围之广，出乎原设计者之意料，尽管缺乏一个正式的赞助机构，它们仍在 AT&T 内部中疯狂地散播。到了 1980 年，已蔓延到大学与研究机构中，还有数以千计的 Hacker 想把 Unix 装在家里的计算机上。当时运行 Unix 的主力计算机是 PDP.11、VAX 系列的计算机。不过由于 Unix 的高移植性，它几乎可安装在所有的计算机机型上。一旦新型计算机上的 Unix 安装好，把软件的 C 语言源代码抓来重新编译就可以了，这比移植汇编语言编写的软件省时省力。

这时候，一套专为 Unix 设计的网络——UUCP：一种低速、不稳但成本很低廉的网络出现了。两台 Unix 计算机用条电话线连起来，就可以互传电子邮件了。UUCP 是内建在 Unix 系统中的，不用另外安装。于是 Unix 站点连成了专属的一套网络，形成其 Hacker 文化。在 1980 第一个 Usenet 站点成立之后，组成了一个特大号的分散式布告栏系统，吸引而来的人数很快地超过了 ARPANET，其中还有少数 Unix 站点连上 ARPANET。PDP.10 与 Unix 的 Hacker 文化开始交流。PDP.10 的 Hacker 们觉得 Unix 的拥护者都是些什么也不懂的新手，比起他们那复杂华丽，令人爱不释手的 LISP 与 ITS，C 语言与 Unix 简直原始得令人好笑。第一部 PC 出现在 1975 年，苹果计算机公司在 1977 年成立，并以飞快的速度成长。微型计算机的潜力，立刻吸引了另一批年轻的 Hacker。他们最爱的程序语言是 BASIC，由于它过于简陋，PDP.10 的死忠派与 Unix 迷们根本不屑用它，更看不起使用它的人。这群 Hacker 中有一位读者一定认识，他的名字叫 Bill Gates，最初就是他在 8080 上发展 BASIC Compiler 的。

1980 年同时有三个 Hacker 文化在独立的发展，它们之间彼此偶有接触与交流。ARPANET/PDP.10 文化，用的是 LISP、MACRO、TOPS.10 与 ITS。Unix 与 C 语言的拥护者用电话线把他们的 PDP.11 与 VAX 计算机串起来使用。还有另一群散乱无秩序的微型计算机迷，致力于将计算机科技平民化。三者中 ITS 文化(也就是以 MIT AIL 为中心的 Hacker 文化)可说在此时达到全盛时期，但乌云逐渐笼罩这个实验室。ITS 赖以维生的 PDP.10 逐渐过时，开始有人离开实验室去外面开公司，将人工智能科技商业化。另一些 MTI AIL 的高手挡不住新公司的高薪挖角而纷纷出走，SAIL 与 CMU 也遭遇到了同样的问题。

致命一击终于来临，1983 年 DEC 宣布：为了要集中精力在 PDP.11 与 VAX 生产线上，将停止生产 PDP.10。ITS 没搞头了，因为它无法移植到其他计算机上，或者说根本没人办得到。而 Berkeley Univeristy 修改过的 Unix 在新型的 VAX 机上运行得很顺，是 ITS 理想的替代品。有远见的人都看得出，在快速成长的微型计算机科技下，Unix 一统江湖是迟早的事。差不多在此时 Steven Levy 完成 Hacker 这本书，主要的资料来源是 Richard