

## 多功能组合木马架构的研究

刘志都<sup>1</sup>, 程新党<sup>1</sup>, 廖湖声<sup>2</sup>

(1. 南阳师范学院 计算机与信息技术学院, 河南 南阳 473061; 2. 北京工业大学 计算机学院, 北京 100022)

**摘要:** 对当前主流木马的实现架构进行了分析, 根据分析结果, 提出了多功能组合木马架构。该架构不仅综合利用现有的木马技术来适应各种复杂的环境, 而且实现了木马功能的多样化。最后, 给出了该木马架构的一些关键实现技术。

**关键词:** 木马; 反病毒技术; 多功能; 架构

**中图分类号:** TP309.5

**文献标志码:** A

**文章编号:** 1009-3486(2008)04-0016-05

### Research on architecture of multifunctional assembled Trojan horse

LIU Zhi-du<sup>1</sup>, CHENG Xin-dang<sup>1</sup>, LIAO Hu-sheng<sup>2</sup>

(1. College of Computer & Information Technology, Nanyang Normal Univ., Nanyang 473061, China; 2. College of Computer Science and Technology, Beijing Univ. of Technology, Beijing 100022, China)

**Abstract:** This paper dealt with the architectures of Trojan horse in vogue, and then presented a new architecture of multifunctional assembled Trojan horse, which could not only use all techniques in completing Trojan horse synthetically, but also achieve diversification of Trojan horse function. Finally some key techniques of the architecture were given.

**Key words:** Trojan horse; anti-computer virus technique; multifunction; architecture

近年来, 黑客攻击层出不穷, 网络盗窃日益猖獗, 对网络安全构成了极大的威胁。木马是黑客实施攻击与盗窃的主要手段之一, 其实现技术也不断向前发展, 各个关键模块都发展出了新技术, 但是各种技术都有其局限性, 文中对现有各种木马技术通用的特洛伊多功能木马结构进行了剖析和研究, 希望能为反木马技术提供参考。

### 1 主流木马架构

木马主要通过渗透进入对方主机系统, 从而实施对远程主机的控制操作。目前, 流行的木马多数为 C/S 结构, 即客户机/服务器结构, 也有人称为控制器/服务器结构。一个完整的“木马”程序包含了两部分: “服务器”和“控制器”。植入用户计算机中的是“服务器”部分, 而所谓的“黑客”正是利用“控制器”进入运行了“服务器”端的电脑。木马程序的“服务器”运行以后, 被植入者的计算机就会有一个或几个端口被打开, 使黑客可以利用这些打开的端口进入被植入者的电脑系统, 进行相应的控制操作<sup>[1]</sup>。其次是 C/S 结构变形, 这类木马目前也很多, 木马的“客户端”植入被入侵者的电脑中, 服务器端为 Web 系统, 实现对客户端的控制或者作为信息中转站。该类木马一般用来实现整齐划一的动作, 比如实现

收稿日期: 2008-03-10; 修回日期: 2008-04-21。

基金项目: 北京市自然科学基金资助项目(4052006)。

作者简介: 刘志都(1957—), 男, 副教授, 主要研究方向为计算机应用与软件工程, E-mail: nyliuzd@126.com。

DDOS 攻击,也可以用来刷流量等,当然也可以用来实现盗号等任务。

## 2 多功能木马架构

木马的功能从早期的单一型开始向混合型转变。早期木马一般通过控制他人计算机实现文件浏览、信息盗取、DDOS 攻击等,随后转变为以盗取某种信息为主的单一型木马,如盗取 QQ 号码、网上银行密码、某款游戏帐号等,这类木马不必实现远程控制的功能,是木马特征的一次重要的转变,同时使木马的功利性更加突出。随着人们安全意识的增强、反木马技术与工具的发展,木马的传播越来越难,为了从一个被植入计算机上获取更大的价值,木马的设计者构造出了多功能木马。最简单的办法是木马携带一个下载器,该下载器可以下载多个指定的木马,但这种硬编码的实现方式一定程度上影响了木马的功能扩展,为此木马的设计者们开发出了一套木马架构,实现了木马自动升级、根据种植者的指令下载木马功能模块、自动组合等一系列功能。此外,为了对付反木马技术、逃避法律制裁,现在设计较好的木马都具备自杀功能,即根据指令,在短时间内从被感染者的计算机中全部清除。

这种新型木马的结构是前两种结构的组合,它目前虽然不是很流行,但却是一种趋势。该类木马分为三部分,植入用户电脑的服务器端、作为信息中转站的 Web 系统和主要用来实现对个别木马的个性化控制的控制端。这种木马的系统架构如图 1 所示。

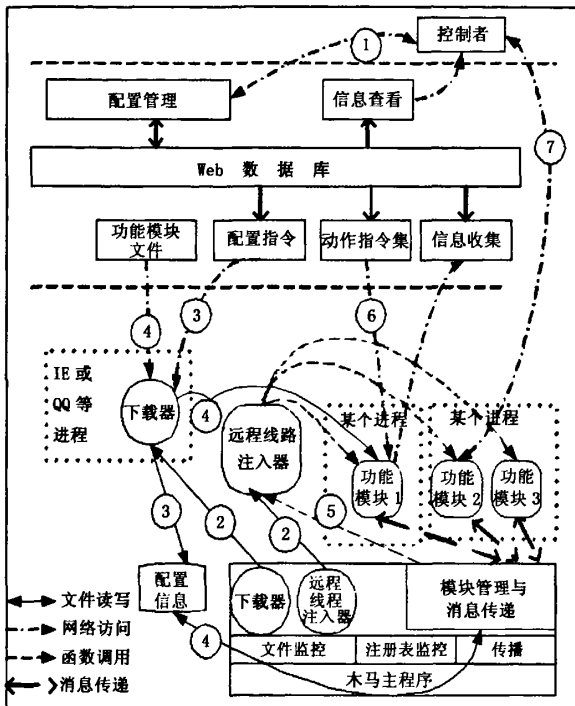


图1 多功能组合木马系统架构

Fig. 1 Architecture of multifunctional assembled Trojan horse

### 2.1 系统流程

多功能组合木马系统的流程如下:

- 1) 木马的控制者通过浏览器设置木马系统配置指令,比如增加新模块、升级旧模块、自杀等功能。针对具体的功能模块配置相应的动作指令,各个功能模块与木马主程序由控制者上传到 Web 服务器。
- 2) 木马主程序携带压缩后的下载器与远程线程注入器模块,主程序运行后,将之释放,然后根据宿主计算机的具体情况,将下载器注入 IE 或 QQ 等本地防火墙允许访问的网络进程内。
- 3) 一般情况下,下载器使用 http 协议从 Web Server 下载当前配置指令,并以二进制的形式存入宿主计算机。
- 4) 木马主程序读取配置信息,指示下载器下载新模块或升级旧模块,并根据本地目录结构与配置指令要求,将模块存入宿主计算机某目录下,若实际存入位置与配置信息不符,则改写配置信息文件。
- 5) 当一个新模块下载成功后,主程序根据本地进程情况使用注入器将模块注入到某个进程,如果需要独立自启动,则主程序为其添加相应的功能。

6) 模块启动后,使用 http 协议访问针对本功能模块的动作指令集,根据指令指示,在本地执行相应的任务,比如刷流量、DDOS 攻击等,当然不是每个模块都需要配置指令。如果功能模块搜集到 QQ 帐号、游戏帐号等信息,若该模块具备信息上报功能,则将信息上报到 Web Server,木马的所有者可以随时查看。

7) 如果木马在宿主计算机上安装了控制模块,则控制模块一般被注入 IE、QQ 或 lass 进程。控制模块一般每隔 5 s 左右访问一次指令集,一旦发现指令要求其某个 IP 地址链接,则主动与控制者的计算机建立链接。

另外,每个模块都携带下载器,但功能模块只对木马主进程进行监控,一旦发现主进程停止,就启动下载器,主程序文件被删除就重新下载,并变名储存<sup>[2]</sup>。

## 2.2 模块管理

木马主程序的模块管理部分主要完成模块监控与信息传递两个功能。模块监控实现各模块被删除后的重新下载及模块停止运行后的重新启动。信息传递部分完成主进程和其他各木马进程的消息交互以及木马内部各模块间的信息传递。由于木马各功能模块各自完成不同的任务,多数情况位于不同的进程之内,它们之间是一种松耦合的关系,为使其能协同工作,必须实现进程内或进程间的通信。通信采用消息传递与事件分发两种方式。

### 2.2.1 模块监控

主程序在资源节内共携带 3 个模块:进程隐藏模块、下载器、远程线程注入器。监控线程运行后,释放出下载器与远程线程注入器,在本地计算机搜索本地防火墙允许访问的网络进程,比如 IE、QQ、MSN 或 lass 等。然后,将下载器注入目标进程,以绕过防火墙访问网络。下载器每隔 5 min 左右从 Web Server 下载配置信息,模块管理部分读取配置信息指示下载器下载各功能模块,并根据本地计算机的具体情况存为指定文件夹下的某个文件。各功能模块下载后,需要独立运行的由监控模块将它启动,需要注入其他进程的,由监控模块搜索合适的进程将它注入。各模块运行后都需要向模块管理部分注册。监控线程每隔 30 s 左右运行一次,对各模块文件及运行情况进行检查。如果某个模块被用户删除,则重新下载并改变储存位置和文件名,然后重新启动。其中,下载器与注入器也具备开机自动运行的功能,下载器每隔一定时间也对主程序文件与运行状况进行检查,一旦发现文件被删除或者停止运行就重新下载变名储存,然后延迟启动。这样,下载器与主进程实现了互相监控,而主程序与每个功能模块也具备相互监控能力,提高了木马的生存能力<sup>[3]</sup>。

### 2.2.2 信息处理

所有模块运行后,首先调用自己的 register 函数将自己的模块文件名以及信息处理函数向管理模块注册,信息传递部分主要实现如下功能:①监听信息,处理消息队列中的消息;②轮询处理所有模块之间的事件,将 main\_event\_queue 队列中的事件分发到相应的模块处理函数 do\_event();③轮询处理其他进程发来的消息,对 message\_queue 队列中的消息由其分发到相应的模块的相应处理函数 do\_message();④如果消息目标不是内部模块,就向其他进程发送消息。

### 2.2.3 模块注册

所有模块运行后首先调用自己的 register 函数将自己的模块文件名以及信息处理函数向管理模块注册<sup>[4]</sup>。注册时根据需要功能模块可注册以下几个函数的子集:模块启动(start)函数、模块停止(stop)函数、模块消息处理(do\_message)函数、模块事件处理(do\_event)函数。

### 2.2.4 消息格式与传递

模块之间传递的消息,必须遵守相同的格式,具体如下所示:

```
<message><to>module</to><from>module</from><key>MSG</key></message>
```

其中:<to>module</to>指出消息接收模块名;<from>module</from>指出消息发送模块名;<key>MSG</key>指出消息内容。模块将构造好的消息送入主进程的消息队列,主程序对消息队列中的消息轮询处理,根据消息的指示,将之分发到相应的模块的消息处理函数 do\_message()。

### 2.2.5 事件处理

模块注册时须向主程序登记它需处理的事件,当事件发出,将被插入到 `main_event_queue` 队列中,主程序轮询处理这些事件,根据事件 ID 查找目的模块名字,然后将其分发到相应模块的事件处理函数。

## 3 系统实现

结合文中提出的新型木马架构,以及木马在加载、隐蔽、反清除、信息采集和网络通信等方面所采用的技术方法,作者构建了一个完整的自组合木马系统,该系统在传播期间能成功自动升级、自由添加功能、瞬时自杀。经近半年的测试,该木马可以成功突破瑞星、金山等公司的防御系统,至今没有任何杀毒软件可以将其清除,下面对关键技术做一些简要介绍。

### 3.1 木马加载

经过测试发现最好的技术不一定可以达到最好的效果,针对主流反病毒软件在 Windows 系统下采用内核级 HOOK 技术,即 HOOK 函数 `ZwSetValueKey` 对注册表实施监控;笔者则采用 `RegRestore` 函数将 `hiv` 文件写回注册表的方法,来覆盖所有具有自启动功能的键值,确保在安全模式下木马进程仍可开机自动运行。要注意的是,在使用 `RegRestoreKey` 前,需将木马进程的权限提升到 `SE_RESTORE_NAME`,然后使用 `RegRestoreKey(key, FileName, 0x00000008L)` 将指定的 `hiv` 文件写回到系统注册表。这种方式几乎可以逃避所有主流的反病毒软件的监控。

### 3.2 文件隐藏

木马文件的隐藏实现方式多种多样,目前主流技术为 Rootkit。而笔者采用总线级文件隐藏技术,即 HOOK `atapi.sys` 中的 `scsi dispatch`,得到 IRP 后,向下进行 IRP 堆栈搜索,当搜索到的 `fileobject` 为需要隐藏的目录时,返回错误,那么该目录下所有的文件都将被隐藏,这种方式对付 `icesword`, `darkspy` 等软件都是有效的。当然,随着反木马技术的发展,总线级隐藏也将失去效果。

### 3.3 进程隐藏

对于进程隐藏,在设计木马时,仅 HOOK `ZwQuerySystemInformation` 函数来隐藏进程;而对其他几个较敏感的 `Zw` 函数则不做处理,经测试除卡巴斯基 7.0 之外;其他任何杀毒软件均不能检测到。

### 3.4 变名技术

变名技术是对隐藏失败的补救措施,一般通过利用 HOOK 技术来实现。如果木马进程或者文件被用户发觉,当清除失败时,则多数用户都产生一种逆反心理,一般都会通过寻求帮助来解决,甚至报案,这样其实违反了木马的隐藏性原则。所谓的变名技术即通过 HOOK 技术捕获用户清除事件,然后完成以下 3 个动作:变名备份、自动退出、延迟重启。这样一般用户会认为已经将该木马进程与文件清除,就不再纠缠,木马也就间接地实现了隐藏。

### 3.5 管理界面

通过控制页面控制者可以浏览木马在线情况,如果控制者对某个宿主计算机感兴趣,可以在控制端 IP 地址中输入自己计算机的 IP 地址,并运行控制端软件,木马接收到指令后,则主动与控制端链接。

通过指令配置页面,可以使用宿主计算机刷流量、提高关键字排名、弹广告、DDOS 攻击等。刷流量提高关键字排名、DDOS 攻击可以使用隐藏显示模式,在宿主计算机用户不察觉的情况下执行。而弹广告则需要在宿主计算机弹出广告页面,所以需要使用不同的显示模式。

总之,当需要对木马增加某种功能时,只需按照模块的设计要求实现,然后在配置页面上添加,并将模块文件上传到服务器,则可以充分利用木马已经大面积传播的优势,使新功能模块迅速得到扩散。

## 4 结束语

本文提出了一种新型的木马结构,并根据研究结论,结合当前木马在加载、隐蔽、反清除、信息采集和网络通信等方面的主流技术,实现了一个多功能木马,并在其传播期间,对其进行跟踪研究,发现多功能组合木马可以适应不同的宿主环境,其健壮性成倍提高,木马的升级扩展更加容易。本文的内容希望能为反病毒技术提供参考。

### 参考文献(References):

- [1] 梅剑峰. Linux下木马技术研究[J]. 微计算机信息, 2006, 3(9): 31-33.  
MEI Jian-feng. The research of the Trojan horse technology in Linux OS [J]. Microcomputer Information, 2006, 3(9): 31-33. (in Chinese)
- [2] 蔡洪民. 特洛伊木马攻击分析与检测技术研究[J]. 计算机安全, 2007(4): 65-67.  
CAI Hong-min. Research of detection technology and attacks analysis of Trojan horse [J]. Network & Computer Security, 2007(4): 65-67. (in Chinese)
- [3] 潘 勉. 基于 DLL 技术的特洛伊木马植入新方案[J]. 计算机工程, 2004, 18(30): 111-112.  
PAN Mian. New Scheme for the injection of Trojan horse based on DLL [J]. Computer Engineering, 2004, 18(30): 111-112. (in Chinese)
- [4] 刘志都, 程新党, 崔 蕊. 软件体系结构模式应用探析[J]. 南阳师范学院学报, 2007, 6(9): 69-72.  
LIU Zhi-du, CHENG Xin-dang, CUI Rui. Application for software architecture module [J]. Journal of Nanyang Normal University, 2007, 6(9): 69-72. (in Chinese)

## 论文预告

# 改进的考虑缓发中子在内的中子倍增公式

王子义

(海军工程大学 船舶与动力学院, 武汉 430033)

作者对先前发表在《核动力工程》1987 年第 6 期上的研究成果“考虑缓发中子在内的中子倍增公式”进行了分析与研究,认为该公式将每一代核裂变产生的缓发中子的先驱核单独存放在其中,公式的项数太多,计算极不方便。在此基础上,采用将缓发中子先驱核浓度累积的方法,建立了“改进的考虑缓发中子在内的中子倍增公式”。这个“公式”简单明了,计算方便。下面列出此公式:

$$\begin{cases} N_1 = k(1-\beta)N_0 + \sum_{j=1}^6 (1-e^{-\lambda_j t})C_{j_0} + S \\ C_{j_1} = e^{-\lambda_j t}C_{j_0} + k \cdot \beta_j N_0 \end{cases} \quad (1)$$

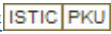
$$\begin{cases} N_2 = k(1-\beta)N_1 + \sum_{j=1}^6 (1-e^{-\lambda_j t})C_{j_1} + S \\ C_{j_2} = e^{-\lambda_j t}C_{j_1} + k \cdot \beta_j N_1 \\ \vdots \end{cases} \quad (2)$$

$$\begin{cases} N_m = k(1-\beta)N_{m-1} + \sum_{j=1}^6 (1-e^{-\lambda_j t})C_{j_{(m-1)}} + S \\ C_{j_m} = e^{-\lambda_j t}C_{j_{(m-1)}} + k \cdot \beta_j N_{m-1} \end{cases} \quad (m)$$

式中:  $j=1, 2, \dots, 6; m=1, 2, 3, \dots, m$ 。

以上公式即为“改进的考虑缓发中子在内的中子倍增公式”。详细内容见本学报 2008 年第 5 期《核裂变链式反应过程中裂变中子密度计算公式建立的研究》一文。

# 多功能组合木马架构的研究

作者: 刘志都, 程新党, 廖湖声, [LIU Zhi-du](#), [CHENG Xin-dang](#), [LIAO Hu-sheng](#)  
作者单位: 刘志都, 程新党, [LIU Zhi-du](#), [CHENG Xin-dang](#) (南阳师范学院, 计算机与信息技术学院, 河南, 南阳, 473061), 廖湖声, [LIAO Hu-sheng](#) (北京工业大学, 计算机学院, 北京, 100022)  
刊名: 海军工程大学学报   
英文刊名: [JOURNAL OF NAVAL UNIVERSITY OF ENGINEERING](#)  
年, 卷(期): 2008, 20 (4)  
被引用次数: 2次

## 参考文献(4条)

1. 梅剑峰 [Linux下木马技术研究](#) [期刊论文]-[微计算机信息](#) 2006 (09)
2. 蔡洪民 特洛伊木马攻击分析与检测技术研究 [期刊论文]-[计算机安全](#) 2007 (04)
3. 潘勉 基于 DLL 技术的特洛伊木马植入新方案 [期刊论文]-[计算机工程](#) 2004 (30)
4. 刘志都;程新党;崔蕊 软件体系结构模式应用探析 [期刊论文]-[南阳师范学院学报](#) 2007 (09)

## 本文读者也读过(10条)

1. 贺小伟. 余景景. 王淼. [HE Xiao-wei](#). [YU Jing-jing](#). [WANG Miao](#) 基于SPI技术漏洞的新型木马的防范方法 [期刊论文]-[西北大学学报 \(自然科学版\)](#) 2006, 36 (6)
2. 陈雷霆. 张亮. [CHEN Lei-ting](#). [ZHANG Liang](#) 人工免疫机制在木马检测系统中的应用研究 [期刊论文]-[电子科技大学学报](#) 2005, 34 (2)
3. 康治平. 向宏. [Kang Zhiping](#). [Xiang Hong](#) 特洛伊木马隐藏技术研究及实践 [期刊论文]-[计算机工程与应用](#) 2006, 42 (9)
4. 赵吉刚. 舒辉. 董卫宇. [ZHAO Ji-gang](#). [SHU Hui](#). [DONG Wei-yu](#) 基于驱动的通用木马结构研究与设计 [期刊论文]-[计算机工程与设计](#) 2008, 29 (16)
5. 梅登华. 林耀通. [MEI Deng Hua](#). [LIN Yao Tong](#) 基于Multi-Agent的木马模型设计 [期刊论文]-[电子技术应用](#) 2008, 34 (5)
6. 汪玉美. 刘萍. 李云. 王海涛. [WANG Yu-mei](#). [LIU Ping](#). [LI Yun](#). [WANG Hai-tao](#) 基于Web Services的木马通信模型研究 [期刊论文]-[计算机工程与设计](#) 2010, 31 (19)
7. 田磊. 李振海. 陈琳. 李世超. [Tian Lei](#). [Li Zhenhai](#). [Chen Lin](#). [Li Shichao](#) 基于局域网渗透的木马技术研究与实现 [期刊论文]-[计算机应用与软件](#) 2007, 24 (10)
8. 钟明全. 李焕洲. 唐彰国. 张健. [ZHONG Ming-quan](#). [LI Huan-zhou](#). [TANG Zhang-guo](#). [ZHANG Jian](#) 基于网络驱动技术的木马通信检测系统 [期刊论文]-[计算机工程](#) 2010, 36 (9)
9. 许建真. 许强. [XU Jian-zhen](#). [Xu Qiang](#) 基于中心环旋转木马的应用层组播模型 [期刊论文]-[计算机应用](#) 2009, 29 (2)
10. 李晓东. 罗平. 曾志峰. [LI Xiao-dong](#). [LUO Ping](#). [ZENG Zhi-feng](#) 利用木马的自启动特性对其进行监控 [期刊论文]-[计算机应用研究](#) 2007, 24 (5)

## 引证文献(2条)

1. 张哲. 孙军安 基于心理的木马反清除技术研究 [期刊论文]-[南阳师范学院学报](#) 2009 (3)
2. 胡波. 曹玖新. 孙学胜. 姚臻. 刘永生 功能原子化的自适应木马模型研究 [期刊论文]-[计算机工程与设计](#) 2010 (12)