第 35 卷 第 20 期 计 算 机 工 程 2009 年 10 月
Vol.35 No.20 Computer Engineering October 2009
·软件技术与数据库· 文章编号：1000—3428(2009)20—0034—03 文献标识码：A 中图分类号：TP311.52

基于 OLE 自动化技术的 PST 文件解析

杨俊彬，曾春溪，蔡剑怀，吴顺祥
(厦门大学自动化系，厦门 361005)

摘 要：为直接解析客户端软件 Outlook 的数据文件，分析其内在编码及逻辑结构，提出一种基于 OLE 自动化技术的 PST 文件解析方法，通过启动 Outlook 自动化服务器，创建 Outlook 自动化对象，利用 Outlook 提供的相关接口函数直接获取文件中的邮件信息。实验结果表明，该方法能够获得良好的效果。

关键词：电子邮件；PST 文件；OLE 自动化技术；文件解析

PST File Parsing Based on OLE Automation Technology

YANG Jun-bin, ZENG Chun-xi, CAI Jian-huai, WU Shun-xiang
(Department of Automation, Xiamen University, Xiamen 361005)

【Abstract】In order to parse the data files kept by Outlook directly, its internal code and logic structure are analyzed, and a PST file parsing method based on OLE automation technology is proposed. By starting Outlook automatic server, the corresponding object is set up. The post information is got by using relevant interface function provided by Outlook. Experimental results show this method can get better effect.

【Key words】E-mail; PST file; OLE automation technology; file parsing

1 概述

随着互联网的发展，电子邮件的使用及基于电子邮件的应用日益广泛[1]。邮件中蕴藏的丰富信息是进行计算机调查取证的重要途径。大多数计算机犯罪案件都涉及电子邮件的调查和取证[2]。

邮件客户端软件 Outlook 是 Microsoft 推出的 Office 办公套件的重要组成部分，是一种用于日常计划和任务管理的实用软件。它对 Outlook Express 的功能进行了扩充，可以帮助收发和管理用户的电子邮件，还包括日历、活动和会议安排以及联系人管理等功能，有着非常广泛的用户群，其保存的邮件数据文件也成为电子数据调查取证的重要对象，挖掘分析其中的有用信息与线索已成为计算机取证的重要研究课题。Outlook 的数据文件为微软所研究的一种复合文档，有其内在的编码及逻辑结构形式。国内外关于该文档格式的资料较少，解析难度大，且解析效率与软件升级后的版本兼容性难保证。

本文提供一种简便、实用及有效的方法，利用 OLE 自动化技术来完成该数据文件的解析，将经过编码的二进制数据文件还原为原始的邮件信息，即从中提取出邮件的收发件人、发送时间、主题、邮件内容及附件等。经实验验证，该方法稳定可行，获得了良好的解析效果。

2 相关知识简介

2.1 PST 数据文件

在 Outlook 中，电子邮件、联系人、约会、任务和日记等所有的 Outlook 项目都保存在一个叫“个人文件夹(.pst)”(Outlook 独创的文件类型)的文件中。PST 文件对于 Outlook 来讲，就好像注册表对于操作系统那样，是 Outlook 的核心。Outlook 可为每个用户都建立自己的 PST 文件。

用户邮箱中的收件箱、发件箱、已删除邮件、已发送邮件、草稿等邮件信息均保存在这个文件中。如果使用 Windows 2000/XP 操作系统，那么该文件的默认位置为：“C:\Documents and Settings\(User Name)\Local Settings\Application Data\Microsoft\Outlook(User Name 是操作系统的登录用户名)”，文件名为 Outlook.pst，这个文件包含了用户的帐户资料与个人信息。

2.2 OLE 自动化技术

对象链接嵌入(Object Linking and Embedding, OLE)技术是应用程序之间交换数据和相互操纵的一种方式，它通过组件对象模型(COM)使得客户模块和服务器模块可以通过特殊接口进行通信，从而实现应用程序间的相互作用。

OLE 自动化技术实现了对 OLE 组件的编程式控制，克服了链接和嵌套中存在的缺点，使用户能够通过编程在一个程序中控制另一个应用程序的对象，从而实现不同应用程序间的信息共享[3]。

OLE 自动化包括 2 个部分应用程序：自动化服务器和自动化客户。自动化服务器为自动化客户暴露出各种可编程的属性和方法，使客户程序可通过某种自动化的过程直接操作这些方法和属性，从而达到控制自动化服务器的目的。OLE 自动化的出现使系统集成从根本上成为可能[4]。

2.3 Outlook 对象模型概述

对象模型是指应用程序的公开函数(可从代码对应用程

基金项目：国家“十一五”科技支撑计划基金资助项目(2007BAK34B04)；国家自然科学基金资助项目(60704042)；厦门大学 985 二期信息创新平台基金资助项目(00002X07204)
作者简介：杨俊彬(1984－)，男，硕士研究生，主研方向：智能信息系统，网络安全与取证；曾春溪，硕士研究生；蔡剑怀，博士研究生；吴顺祥，教授、博士
收稿日期：2009-06-10 **E-mail：**yjbly1068@163.com

序进行访问的函数)。这些函数是作为一组对象公开的，其中的每个对象都具有属性、方法和事件。对象模型包含一组可用于创建对象的定义或类。

Outlook 对象模型的核心是 Application 对象，之所以称其为根对象，是因为层次结构的其余部分都源自它。Application 对象提供对其他所有 Outlook 对象的访问。

如果需要从外部应用程序访问 Outlook 对象层次结构，那么必须先创建 Application 对象的实例，才能访问任何其他对象。

尽管 Application 对象允许访问 Outlook 中的许多基础构造块对象，但是如果要访问 Outlook 数据，则必须创建 Namespace 对象的实例。Namespace 对象是 Outlook 数据源的抽象根，这意味着，尽管不直接使用它，也可以通过它访问对象树中其下面的对象。目前，所支持的唯一数据源就是邮件应用程序编程接口(MAPI)，MAPI 允许访问存储在用户邮件文件中的所有 Outlook 数据。要获得 Outlook 应用程序的 Namespace 对象，需使用 Application 对象的 GetNameSpace 方法。

Outlook 中的信息是在文件夹中维护的。某些文件夹(如收件箱、发件箱和已发送邮件)包含邮件项；其他文件夹则包含其他类型的项目。在获取 Namespace 对象的实例以后，就可以方便地连接到 Outlook 中的任何文件夹。

Namespace 对象有一个 GetDefaultFolder 方法，该方法使用类型为 olDefaultFolders 的参数。类型 olDefaultFolders 表示某个默认的 Outlook 文件夹，并且可以是表 1 中所示的任何一个常数。

表 1　OlDefaultFolders 常数

常数	用途
olFolderCalendar	返回一个包含所有日历项的文件夹
olFolderContacts	返回一个包含所有联系人项目的文件夹
olFolderDeletedItems	返回一个包含所有已删除邮件项目的文件夹
olFolderDrafts	返回一个包含所有草稿邮件项目的文件夹
olFolderInbox	返回一个包含所有收件箱邮件项目的文件夹
olFolderJournal	返回一个包含所有日记项目的文件夹
olFolderNotes	返回一个包含所有便笺项目的文件夹
olFolderOutbox	返回一个包含所有发件箱邮件项目的文件夹
olFolderSentMail	返回一个包含所有已发送邮件项目的文件夹
olFolderTasks	返回一个包含所有任务项目的文件夹
olPublicFoldersAllPublicFolders	返回一个包含所有公用文件夹项目的文件夹

Outlook.MAPIFolder 对象表示包含电子邮件、联系人、任务及其他项的文件夹。Outlook 提供 16 个默认 MapiFolder 对象，它们由 Outlook.OlDefaultFolders 枚举值定义。例如，OlDefaultFolders.olFolderInbox 与 Outlook 中的“收件箱”文件夹相对应。

Outlook.MailItem 对象表示电子邮件。MailItem 对象通常在文件夹中，其包含了可用来创建和发送电子邮件的属性和方法。

3　基于 OLE 自动化技术的 PST 邮件文件解析

Microsoft Office 的产品都提供了一种控制其应用程序的方法(OLE 自动化)，作为同为微软公司的产品 Visual C++，根据其提供的实现自动化客户端的类，可以很方便地操作 Outlook 暴露给外部的可编程接口，并根据接口描述很方便地解析每一封邮件，提取相应的邮件信息，从而达到对 PST 邮件文件进行解析的目的。

解析中参考了 Microsoft 的相关技术文档和文献，具体流程如图 1 所示。

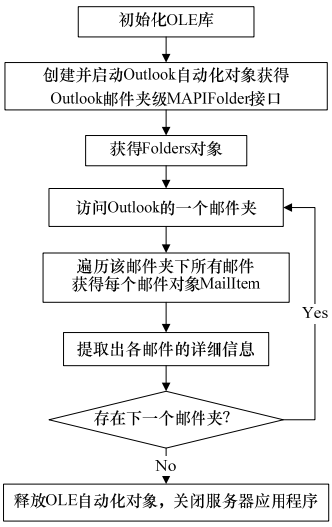

图 1　基于 OLE 自动化技术的 PST 邮件文件解析流程

详细解析步骤如下：

(1)实现任意路径下的 PST 文件解析

如果没有经过必要的预先处理，利用 OLE 自动化技术所解析的 PST 文件总是 Outlook 默认存储的数据文件。经过研究发现，Outlook 的 PST 数据文件路径的信息存储在注册表中的特定位置为在 HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows Messaging Subsystem\Profiles\Outlook 下面某子目录下的名为 01020fff 的子键中，其默认值为：C:\Documents and Settings\Administrator\Local Settings\Application Data\Microsoft\Outlook\Outlook.pst 的 Unicode 编码形式，该键值的前 54 个字节为固定字符，后面为路径信息。

需要注意的是，01020fff 子键在多个子目录下都可能出现，经研究发现，在程序所需子键的同一子目录下，必存在另一子键名为 00033009，且其键值为“02 00 00 00”，通过此特征可间接定位该 01020fff 子键。

因此，在创建 Outlook 自动化服务器对象之前，通过程序首先定位并修改注册表对应的 01020fff 键值，把默认路径替换为所要解析的 PST 数据文件所在的路径，替换时注意采用路径的 Unicode 编码，便可实现任意路径下的 PST 数据文件的解析。

在修改注册表前，需要将注册表的默认值保存起来，以便在程序结束时能还原注册表。

(2)启动 Outlook 自动化服务器，创建 Outlook 自动化对象

初始化 OLE 库，创建一个 Outlook 自动化服务器类的实例，得到一个指向 Outlook 自动化服务器对象的指针，该接口类型为_Application。接着调用该接口的 GetNamespace()方法，获得 Outlook 的“MAPI”数据存储空间接口，并通过指定 Outlook 默认收件箱文件夹，定位到 Outlook 文件夹消息应用程序编程接口 MAPIFolder(邮件夹级)。

(3)遍历 Outlook 中的邮件夹

调用 MAPIFolder 接口的 GetFolders()函数，获得 Folders 对象，然后根据子文件夹数 Folders.GetCount()按顺序遍历 Outlook 下的所有子文件夹，获得每个邮件夹对象的 m_lpDispatch 指针，程序流程如图 2 所示，其中，在按顺序遍历中，当计数为 1,2,3,4,10,11 时，分别对应已删除邮件、收件箱、发件箱、已发送邮件、草稿和垃圾邮件这 6 个邮件文件夹。

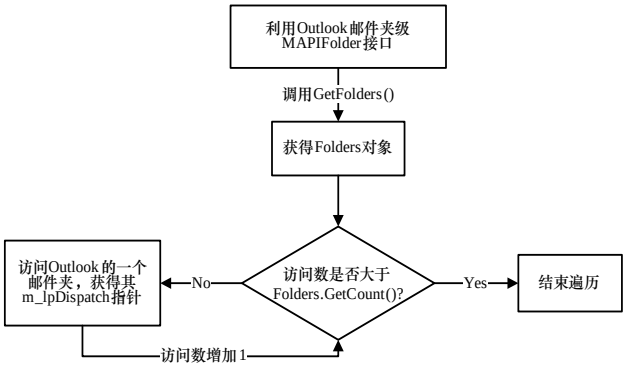

图 2　遍历 Outlook 邮件夹程序流程

(4)遍历 Outlook 某邮件夹下的所有邮件

传入邮件夹的 m_lpDispatch 指针，获得次一级 MAPIFolder 接口(邮件级)，调用 GetItems( )函数，获得 Items 对象，然后根据邮件数 FolderItems.GetCount( )按顺序遍历该邮件夹下的所有邮件，获得每个邮件对象 MailItem，并保存其 m_lpDispatch 指针，以便后面能快速定位到该邮件进行访问，程序流程如图 3 所示。

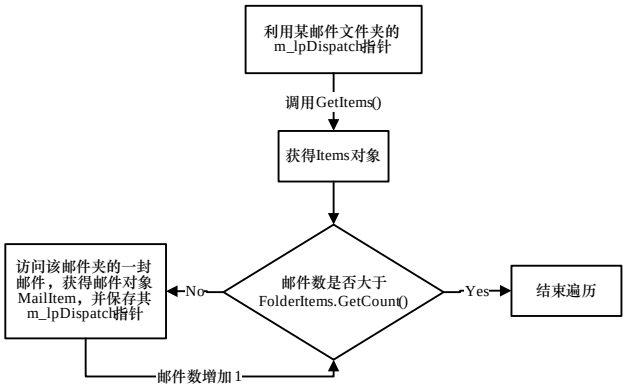

图 3　遍历 Outlook 某邮件夹下所有邮件程序流程

(5)提取 Outlook 每封邮件的信息

根据每个 MailItem 邮件对象，调用其中定义的各种属性、方法及函数，便可提取每封邮件的详细信息，包括邮件的收发账户 Email 地址、主题、发送时间、邮件内容和附件等信息，程序流程如图 4 所示。

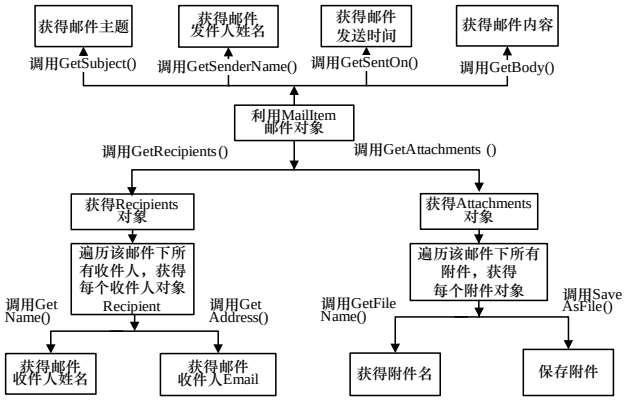

图 4　获取 Outlook 邮件信息程序流程

需要强调的是：在提取一封邮件的收件人的姓名及 E-mail 地址相关信息之前，需要调用 MailItem 对象的 GetRecipients()函数，获得 Recipients 对象，然后根据收件人数 Recipient.GetCount()按顺序遍历，便可获得该封邮件所有收件人的相关信息。另外，提取每封邮件的附件信息之前，需要调用 MailItem 对象的 GetAttachments()函数，获得 Attachments 对象，然后根据附件数 Attachment.GetCount()按顺序遍历，便可获得该封邮件所有附件的相关信息，包括通过其下的 API 函数导出附件。

(6)解析结束

释放 OLE 自动化对象，关闭 Outlook 服务器应用程序，并将注册表中“01020fff”键值还原成默认的值。

4　实例验证

本文在 Windows XP+SP2 操作系统平台上，用 Microsoft Visual C++6.0 开发环境编程实现了上述方法。在预先安装 Outlook 2003 的前提下，通过在 Windows 2000,XP,2003 和 Vista 4 个不同版本的 Windows 操作系统下，用多组数据反复进行验证，均能得到正确的解析结果，运行界面如图 5 所示，结果表明，基于 OLE 自动化技术的 PST 邮件文件解析方法是可行的，具有较高的稳定性。

图 5　PST 邮件数据文件的解析界面

5　结束语

本文针对直接解析 PST 邮件数据文件难度较大的问题，提出基于 OLE 自动化技术的解析方法，利用 Outlook 提供的相关的接口函数直接获取文件中的邮件信息，避免了研究分析复杂的 PST 邮件文件格式，为该数据文件的解析找到一种高效的解决方法，也为深入分析及进一步挖掘邮件关键信息提供了重要的素材。

参考文献

[1] 张学旺, 汪林林. 一种安全 Web 电子邮件客户端设计[J]. 计算机工程, 2008, 34(14): 171-173.

[2] 刘浩阳. 电子邮件的调查与取证[J]. 辽宁警专学报, 2007, 27(5): 27-31.

[3] Kraig B. Inside OLE[M]. [S. l.]: Microsoft Press, 1995.

[4] 杨　华, 高克昌. 用 OLE 自动化技术开发应用型地理信息系统[J]. 计算机应用研究, 2004, 21(1): 191-193.

编辑　陈　文