基于安全机制的文件共享和传输的优化设计

崔善童,李曼珍
(东华大学 信息科学与技术学院,上海 201600)

摘 要:分析了现有的文件共享和传输的相关技术,针对存在的某些安全隐患进行了相应的优化设计。在文件共享部分,通过采用基于 XML 的间接文件共享方式,即只公开共享信息,而非直接挂载共享文件,避免了因共享路径产生的安全漏洞;在文件传输部分加入了多重安全机制,保证了文件传输的安全性。分析表明,本设计在不影响文件共享传输效率的情况下,使得局域网内文件共享和传输更加安全可靠。

关键词:文件共享和传输;间接共享;可拓展标记语言(XML);安全机制;局域网
中图分类号:TP393 **文献标识码**:A **文章编号**:1674-7720(2013)20-0094-04

Optimal design of file sharing and transmission based on the security mechanism

Cui Shantong, Li Manzhen
(College of Information Science and Technology, Donghua University, Shanghai 201600, China)

Abstract: The paper analyzes the existing file sharing and transmission of relevant technology. In view of the existence of some security risks for which the corresponding optimized design. In file sharing section, by using indirect file sharing based on XML, which only open sharing of information, but not directly to mount the shared file, avoids security vulnerabilities due to the shared path; in the file transfer section, adding multiple security mechanisms to ensure the security of the file transfer. Analysis shows that this design does not affect the file-sharing transmission efficiency, making the LAN file sharing and transfer more secure and reliable.

Key words: file sharing and transmission; indirect file sharing; eXtensible markup language (XML); security mechanism; local area networks(LAN)

随着网络技术的飞速发展,利用网络实现文件的共享和传输,给人们带来了极大的便利。然而安全问题却日益突出,尤其在局域网范围内,由于共享和传输十分频繁,可能导致单位或企业的机密信息泄露。此外,通过传输病毒或木马,可能导致整个局域网的瘫痪,因此局域网的安全文件共享和传输技术至关重要。

本文针对现有技术的安全隐患,设计了一套基于安全机制的文件共享和传输系统。在文件共享部分,对局域网只公布本机的共享信息,而不直接共享文件,以防止设置共享路径而产生的安全漏洞;在文件传输部分,引入多重安全机制,保证了文件传输的安全。

1 文件共享和传输

1.1 现有技术分析

NFS 网络文件系统[1]实现了 Linux 之间的文件共享,具有良好的安全性。但是 NFS 服务器管理复杂且维护成本高。

Windows 的网上邻居存在共享信息更新缓慢,且设置共享路径容易成为网络病毒或木马的突破口,使得本机甚至局域网受到非法攻击。

Samba 文件系统[2-3]以及 FTP 文件传输协议[4-5]的身份认证过于单一,存在着中间人攻击、密码嗅探等固有缺陷。

PKI 和 SSL 传输协议依赖第三方认证机构 CA[6],需要定期对过期证书重新认证和更新,管理复杂,极大地影响文件传输的速度和用户体验。

此外,对共享文件授权都只分为“用户本身”、“组”、“其他”三种身份,权限开放容易造成过大或者过小。例如,A 文件只共享给所在“组”的部分人,却只能选择分组共享,使得小组其他成员也能获取共享文件,不能指定到具体共享人,缺乏共享文件和传输的针对性。

1.2 设计要求

本设计在局域网环境下,解决上述技术存在的不足

或安全隐患。优化设计的要求如下：

(1)避免设置特定共享路径，不直接挂载共享文件；

(2)实现点对点文件传输，不设置专用服务器进行集中式管理；

(3)采用多重安全机制；

(4)保证共享信息及时更新；

(5)不依赖于第三方证书认证机构 CA；

(6)对文件共享人的权限精确化。

2 系统软件设计

本系统设计包含文件共享和传输两个子系统。共享子系统中包括共享信息的配置、共享信息查询与响应两个功能模块；传输子系统中包含共享方和请求方两个功能模块。

根据设计要求，采用以下方法来实现：

(1)只显示共享信息，采用间接共享方式，防止共享路径的挂载点被攻击。

(2)文件共享时，不设置专门服务器管理共享文件，采用点对点进行共享信息交互。

(3)在文件传输过程中，引入差别身份认证、可信 IP 认证、文件加密传输、加密密钥 TPM 保护、文件完整性校验和审计等多重安全机制，保证文件共享和传输的安全性。

(4)在共享信息查询中，利用 UDP 广播，对在线 PC 的共享信息进行更新，保证共享信息的实时性。

(5)采用差别身份认证方式，对非敏感共享文件基于静态口令认证；敏感共享文件基于口令和证书双重认证。差别认证保证了易用性和安全性的统一。

(6)对共享文件指定文件共享人，将共享人与对应的 IP 进行绑定，实现对共享人的精确指定。

2.1 文件共享设计

文件共享主要包含共享信息配置、共享信息的查询与应答两个功能模块。

2.1.1 共享信息配置

共享信息配置是对共享文件的绝对路径、文件名、共享人(IP)、文件夹属性等信息进行配置，构建本机共享信息。利用 XML(eXtensible Markup Language)技术[7]将共享信息写入到 XML 文件中。

共享信息 XML 文件格式如图 1 所示。

```
<?xml version="1.0" encoding="UTF-8"?>
<sharelist>
  <list fullname="/root/sharefile">
    <Ip>192.168.1.1</Ip>
    <Ip>192.168.1.2</Ip>
    <Name>sharefile</Name>
    <Node>Y</Node>
  </list>
  <list fullname="/home/test.c">
    <Ip>192.168.1.10</Ip>
    <Ip>192.168.1.21</Ip>
    <Name>test.c</Name>
  </list>
</sharelist>
```

图 1 共享信息 XML 文件

其中第一行为 XML 版本和格式，共享信息以<sharelist>为开始，以</sharelist>结束。每个共享文件以<list>父结点为开始，再以</list>结束。List 结点中包含<fullname>、<Ip>、<Name>和<Node>子结点，分别代表共享文件的路径、共享人 IP、文件名和文件夹。

利用 XML 实现共享信息配置的流程如图 2 所示。

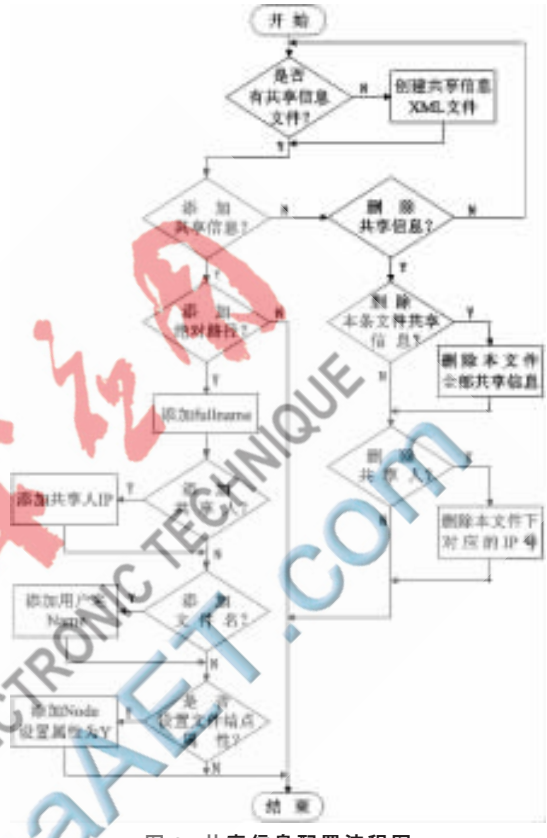

图 2 共享信息配置流程图

共享信息配置的流程如下：

(1)查看本地共享信息。若有，则显示本地共享信息；若没有，则创建 XML 文件。

(2)选择共享文件配置的功能选项：添加还是删除。选择添加转步骤(3)；选择删除转步骤(7)。

(3)是否添加共享路径。若是，添加共享文件，并添加共享路径<fullname>主结点；若否，则结束配置。

(4)是否有添加共享人。若有，在该共享文件下的<Ip>结点添加共享人 IP；若没有，直接下一步。

(5)是否有添加的文件名。若有，在<Name>结点添加文件名；若没有，直接下一步。

(6)若共享为文件夹，则设置<Node>结点为 Y；若共享为文件，则直接结束。

(7)是否删除本文件共享。若是，则删除本条共享信息下的所有结点信息；若否，则直接下一步。

(8)是否只删除本文件共享人。若是，则删除本条共享信息的<Ip>结点信息；若否，则结束配置。

2.1.2 共享信息查询和应答

共享信息查询和应答采用 UDP 协议[8]。程序都设置为多线程，以满足多请求和多响应的情况，即：

(1)请求方利用广播机制，向在线 PC 发出请求；

(2)共享方监听固定的端口号,如果在固定端口监听到有请求信息,则发出应答信息;

(3)请求者验证应答信息后,接收来自共享方的共享信息。

2.2 文件传输设计

2.2.1 文件传输流程

文件传输采用 TCP 协议[9],请求方请求文件共享,得到共享方的响应后,接受共享方的安全认证和共享文件;共享方负责对文件共享请求的响应,在传输前对客户端进行安全认证,在认证通过后,传输共享文件。具体流程如图 3 所示。

图 3 文件传输流程图

文件传输的流程如下:

(1)请求者查询共享信息列表,选择共享信息,进行共享请求。

(2)共享者对请求者进行身份认证。若身份认证失败,结束响应;若身份认证成功,进入下一步。

(3)请求者发送所需共享文件名,共享者获取该共享文件的共享人 IP 列表。

(4)对请求者进行 IP 认证。如果 IP 列表中不存在请求者 IP,则 IP 认证不通过,结束响应;反之,进入下一步。

(5)共享方获取该共享文件路径,得到共享文件的 Hash 摘要值。

(6)利用随机生成的密钥加密共享文件,同时加密密钥,发送加密后的密钥和摘要值给请求者。

(7)发送共享密文给请求者。

(8)请求者解密密钥,利用密钥解密共享密文,得到共享文件。

(9)对接收的共享文件进行 Hash,对比文件传输前后的摘要值,校验文件的完整性。

(10)在文件传输过程中,对整个传输过程进行审计,记录主要操作信息。

2.2.2 安全机制

(1)差别身份认证

身份认证包含静态口令认证和证书认证。对共享文件敏感标记,非敏感文件采用用户名、口令身份认证;敏感文件还需要基于证书的重认证。

(2)共享人 IP 认证

解析请求者 IP 号,与共享信息的共享人 IP 列表进行对比,如果为共享人中的 IP,则 IP 认证成功,请求者可以共享;否则认证失败,拒绝请求者共享。

(3)密钥产生和保护

随机产生一组 2 048 bit 密钥,并把该密钥存储于 TPM 的 NV 空间内进行保存。

TPM(可信计算模块),是安全存储的芯片,TPM 的 NV 空间是 TPM 的独立内存空间[10],存储敏感信息。密钥存储于 NV 空间,保证了密钥的安全性。

(4)文件加解密

文件加密是读取共享文件,利用随机产生的加密算法进行加密,记录算法 ID 生成密文头文件,并依次存储加密内容;文件解密利用密钥作为解密口令,读取密文头文件算法 ID,完成对应算法解密。

(5)完整性校验

客户端获取解密文件摘要值,与接收的原摘要值进行对比。如果相同,则共享文件完整;反之,则文件在传输过程中遭到了破坏或篡改。

3 性能及安全性分析

3.1 性能分析

由于文件传输过程中,加入了安全机制保证传输的安全性,因此还要对其性能影响进行分析,利用与未加入安全机制的文件传输协议进行对比分析。

为保证性能分析的适用范围,测试文件大小从 4 KB 一直到 100 MB 不等。对每个文件测试 10 次,差分取平均值,避免偶然误差。测试结果如表 1 所示。

表 1 传输时间比较

文件大小	传输时间/ms		安全机制的时间消耗率/%
	未采用安全机制	采用安全机制	
4 KB	2.3	3.7	37.8
52 KB	7.0	10	30.0
112 KB	14.8	23.3	36.4
600 KB	149.1	207.0	28.0
1.2 MB	238.4	313.1	23.9
9.21 MB	1 101.8	1 561.7	29.4
24 MB	2 100.5	2 786.5	24.6
72 MB	9 000.2	11 320.1	20.5
102 MB	12 040.9	13 992.4	13.9

通过表 1 可以得出：

(1)未加入安全机制的传输时间都小于加入安全机制的传输时间。

(2)采用安全机制的共享传输，随着文件大小的增加，时间消耗率逐渐降低。(时间消耗率=(安全机制下传输时间-非安全机制下传输时间)/安全机制下传输时间×100%)。如图 4 所示。

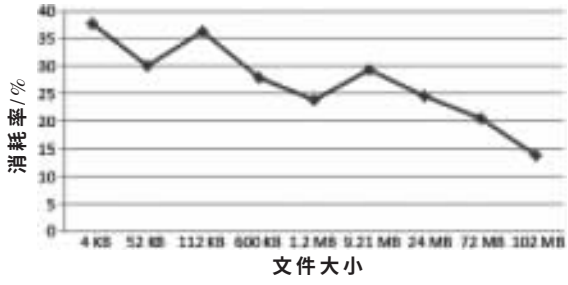

图 4 安全机制时间消耗率变化

通过以上分析，可以得出，随着文件的增大，因安全机制消耗的时间百分比越来越小，即对传输的影响在降低。同时，小文件传输虽然时间消耗率大，但是传输时间基数小。因此，加入安全机制的共享传输对传输效率的影响很小。本设计不影响共享文件传输的传输效率。

3.2 安全性分析

本文提出的文件共享和传输的具体安全性比较如表 2 所示。

表 2 方案比较

	NFS	Samba	本系统
身份认证机制	单一	单一	灵活
IP 认证	有	部分	有
共享授权准确	部分	部分	准确
是否密钥保护	否	是	是
完整性校验	有	无	有
集中式管理	需要	需要	不需要
数据加密	无	无	有
抗重放攻击	无	无	有
依赖第三方 CA	依赖	依赖	不依赖

由表可知，本设计具有以下安全特性：

(1)差别身份认证

基于用户名、口令的认证较单一，但快速易用；基于证书的认证安全性高，但认证复杂；采用差别身份认证，根据文件敏感度不同差别认证，保证了易用性和安全性的双重要求。

(2)去除设置共享路径的安全漏洞

通过间接文件共享方式共享请求时，只对请求者公布共享信息，而不直接共享。必须通过共享方的多重安全认证后，请求方被动接收共享文件。解决了因直接挂载共享文件而产生的共享路径安全漏洞。

(3)共享授权具体化

通过设置共享人，指定共享人的个数和对应 IP，保证了共享授权的具体化。

(4)共享文件保密性

在文件传输前，随机产生加密算法，增大了破解难度。实现密文传输，防止文件在传输中被截取。

(5)抗重放攻击

每个文件在不同时期加密密钥不相同，可以防止利用过期的随机值进行重复请求，即防止重放攻击。

(6)文件完整性

对比文件传输前后的摘要值，防止文件在传输过程中被篡改或者破坏，保证了文件的完整性。

本文主要针对现有的安全共享和传输技术在局域网内应用方面可能存在的安全隐患，采用基于 XML 文件的间接文件共享，并加入了多重安全机制，在不影响共享文件传输效率的基础上增强了文件共享和传输的安全性。

参考文献

[1] RUSSEL S, DAVID G, STEVE K. Design and implementation of the sun network filesystem[C]. Proc of the Summer 1985 USENIX Conf. El Cerrito, CA: USENIX Association, 1985: 119-130.

[2] MOSKOWITZ J, BOUTELL T. Windows and Linux integration: hands-on solutions for a mixed environment[M]. New Jersey: Wiley Publishing Inc. 2007.

[3] TERPSTRA J H, VERNOOIJ J R. The official samba-3HOWTO and reference guide(2nd ed)[M]. NJ: Prentice Hall PTR, 2005.

[4] POSTEL J, REYNOLDS J. File transfer protocol, RFC 959[C]. Menlo Park, CA: SRI International, Network Information Center, 1985.

[5] 许君，王朝坤，李瑞. 基于内容的分布式 FTP 搜索引擎的设计与实现[J]. 计算机研究与发展，2011，48(S3)：430-435.

[6] 颜海龙，闫巧，冯纪强. 基于 PKI/CA 互信互认体系的电子政务[J]. 深圳大学学报(理工版)，2012，29(2)：113-117.

[7] 郭艳艳，吴扬扬. 一种基于 XML Schema 的 XML 索引[J]. 华侨大学学报(自然科学版)，2011，32(1)：43-47.

[8] 孙瑞锦，徐博，周玉明. 一种实时监测基于 UDP 的 Skype 语音流的算法[J]. 解放军理工大学学报(自然科学版)，2008，9(5)：507-511.

[9] 王雷. TCP/IP 网络编程技术基础[M]. 北京：清华大学出版社，2012.

[10] Trusted Computing Group. TPM main specification version2.0 [EB/OL]. http://www.trustedcomputinggroup.org. [2013-03].

(收稿日期：2013-07-28)

作者简介：

崔善童，男，1989 年生，硕士研究生，主要研究方向：计算机网络通信安全。