# 刀片加密服务器的文件加脱密技术

孙秀丽
(济南大学控制科学与控制工程学院　济南 250022)

**摘　要**　刀片加密服务器可以比较全面地解决从信息传输到信息存储全过程的信息安全问题。它主要应用在建立安全公用信息平台以及电子商务、电子政务等方面。利用刀片加密服务器建立的安全公用信息平台，将能逐步改变企业传统的建网观念，使企业由独立建网逐步转变为租用公网，计算机网络将真正走向公用化。本文主要介绍了刀片加密服务器的文件加脱密技术。

**关键词**　刀片加密服务器，NFS，加脱密技术，算法

## The Encryption/ Decryption of Encryption Sever

SUN Xiu-Li
(School of Control Science and Control Engineering Jinan University, Jinan 250022)

**Abstract**　The encryption sever can solve the safe questions of information transmission and memory. It is common in use of the safe of information, such as electron commerce, electron Government. This system can transform building network into renting. This paper introduces the SSL of the system.

**Keywords**　Encryption sever, NFS, Encryption/ decryption, Arithmetic

信息安全是信息传输和存储过程中的一个关键问题，并已受到越来越多的关注和重视，只有解决好信息的安全问题，才能使网络资源得到充分的利用。刀片加密服务器就是针对这一问题的一个解决方案。利用刀片加密服务器建立的安全公用信息平台，将能逐步改变企业传统的建网观念，使企业由独立建网逐步转变为租用公网，计算机网络将真正走向公用化。在利用刀片加密服务器建立的公用信息平台中，数据信息只有授权用户通过证书认证后经过安全通道才可以看到明文，而非授权用户无论通过什么途径都得不到任何有效信息，从而保障了信息传输和信息存储的安全性，同时不改变用户的使用习惯，不影响数据库的结构，对于合法用户应保持透明。在刀片加密服务器的设计中，文件的加脱密技术在系统占有重要的地位。

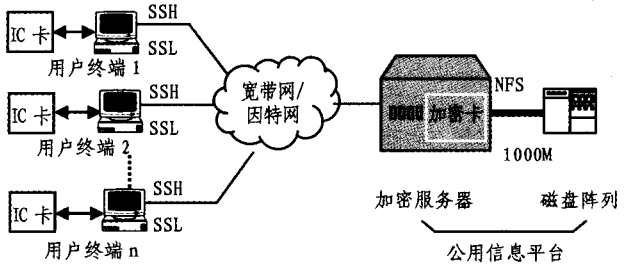


图 1　加密服务器网络应用拓扑图

## 1　刀片加密服务器设计的基本原则

刀片加密服务器设计的基本原则[1]：是在公用信息平台中的数据信息只有授权用户通过证书认证后，经过安全通道才可以看到明文，而非授权用户无论通过什么途径都得不到任何有效信息。保障信息传输和信息存储的安全性，同时不改变用户的使用习惯，不影响数据库的结构，对于合法用户应保持透明。刀片加密服务器网络的原理图如图 1 所示。

图 1 中用户终端通过宽带网或 PSTN[2] 拨号方式连到服务器，刀片加密服务器通过 1000M 以太网口连接到网络磁盘阵列[3]。刀片加密服务器和磁盘阵列构成了公用信息平台[4]。加密型刀片服务器与传统刀片服务器的不同之处有二：一是它实现了在操作系统层面上的数据加解密技术，用户不需要通过编写应用程序来实现数据的加解密；二是采用了基于安全证书的网络全过程的安全措施，数字签名、签名验证、SSL 和 SSH 技术的采用可保证用户能够安全有效地在远端控制刀片加密服务器，并同时实现了数据传输、数据存储的加解密。

## 2　刀片加密服务器设计的工作原理

用户通过因特网、刀片加密服务器来访问磁盘阵列中的数据。在刀片加密服务器中装有改进的 L 操作系统，用户的数据通过加密型刀片服务器进行加密后存储在磁盘阵列中。用户在访问磁盘阵列中的数据时必须通过安全认证和建立 SSL 或 SSH 后才能访问通过加密型刀片服务器解密后的数据。以上的加解密过程均是在操作系统层次上完成的，因此对于用户的应用来讲加解密过程是透明的。

加密型刀片服务器实现了以下功能：基于 SSL 和 SSH[5] 的密传；基于安全证书的安全认证；在操作系统层上并基于对称算法的密存；用户通过 SSL 和 SSH 安全通道在远端对系统进行管理；所有的算法均符合国家密码管理委员会的有关规定等。

加密型刀片服务器的研制成功不仅解决了信息的按权限分级管理的难题，还将对计算机网络和安全公用信息平台的建设、推广产生巨大的推进作用。加密型刀片服务器能够通

*)基金项目：国密办字[2002]133 号文。**孙秀丽**　讲师，硕士，主要研究方向：控制理论与控制工程。

过数字签名[2]和签名验证来确认用户的身份，然后根据用户的身份确定是否对相应的信息进行解密。由于存放在硬盘上的信息是密文存储，即使是超级用户也只能看到密文，无法得到有效的信息。

文件的加脱密操作是在网络文件系统 NFS 上实现的，由算法芯片 SSP02-A 完成加脱密运算。由于刀片加密服务器系统主要是面向公共信息平台用户，而公共信息平台是由若干个刀片加密服务器和公用磁盘阵列构成，数据集中存储在磁盘阵列上。每个刀片加密服务器可以通过 NFS 方式访问磁盘阵列，将磁盘阵列上的某一部分空间挂载到刀片加密服务器上，让刀片加密服务器方便地使用磁盘阵列上的文件，就像本地操作一样，同时便于实现集中管理。

Linux 操作系统通过虚拟文件系统 VFS 实现对多种不同类型文件系统的支持。每一种文件系统都有自己的组织结构和文件操作函数，相互之间差别很大，VFS 的作用就是屏蔽掉这些差异，给用户、应用程序、甚至 Linux 的其他管理模块提供一个统一的界面。

## 3 改造网络文件系统 NFS 的优点

### 3.1 改造后的网络文件系统对系统和应用程序保持透明

首先我们分析一下网络文件系统的文件结构。网络文件系统和 ext、ext2、minix 等文件系统结构类似，是由超级块、inode、文件操作函数入口等部分组成。超级块主要用来描述目录和文件在磁盘上的静态分布（包括目录、文件的大小和结构）；文件系统由子目录和文件构成，每个子目录或文件只能有唯一的 inode 描述，inode 是管理文件系统的最基本单位，也是文件系统连接任何子目录、文件的桥梁；文件操作函数是指对文件读写操作的函数，我们对网络文件系统 NFS 的改造是在文件操作函数入口上进行的。

应用程序对文件的访问过程是从应用程序向系统发布一个调用请求开始的。当某个进程发布了一个面向磁盘阵列文件的系统调用请求时，操作系统内核将调用 VFS 中的相应函数，这个函数将处理一些与物理结构无关的操作，并且把它重定向为网络文件系统中的函数调用，网络文件系统则处理与物理结构相关的操作。

通过对以上过程的描述可以看出，如果在网络文件系统层进行改造，那么存储在磁盘阵列上的文件对于系统和应用程序都是透明的。

### 3.2 对于通过网络文件系统访问的文件，具有对文件的加脱密功能，加脱密处理由 SSP02-A 算法芯片完成

当网络文件系统进行文件写入操作时，把需要写入的文件数据加密，并将密文数据写入到磁盘阵列上；相反，当网络文件系统进行文件读取操作时，把读取到的密文数据脱密，并将脱密后的明文返回给文件系统。通过以上两个过程完成对存储在磁盘阵列上的文件数据的加脱密操作。

### 3.3 过网络系统保存的文件，用块加密的方式实现

块加密比流式加密具有更好的安全性，但是块加密时对于不到一块的数据需要进行填充，所以需要改变文件的大小。

下面我们通过图 1 所示的文件定位示意图来描述文件加脱密实现过程。

如果网络文件系统 NFS 对图 1 中 $L$ 段的数据进行随机读写操作时，对称加密算法对 $L$ 段数据信息的加密处理是将 $L$ 长的明文加密后得到 $L$ 长的密文，解密处理是将 $L$ 长的密文脱密后得到 $L$ 长的明文，该过程要求 $L$ 段的数据长度（字节数）为加密长度 $F$（$F=16$）的整数倍。下面我们以文件写操作过程为例介绍 NFS 对文件的读写操作处理。文件读操作过程和文件写操作过程类似。

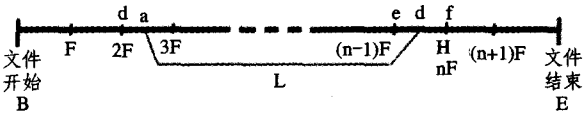


图 2 文件定位示意图

在进行文件写操作时，假设被写入的明文数据段的起点——即该段数据在文件中的起始偏移量为 $a$、结束点——即结束偏移量为 $b$、长度为 $L$，NFS 首先判断 $a$ 是否是加密长度 $F$ 的整数倍：

① 如果不是，那么需将起始偏移量 $a$ 向前移动到加密长度 $F$ 的整数倍的位置（假设向前移动了 $c$ 个位置，到达 $d$ 处，$c<F$），并从文件的该位置（即偏移量为 $a-c=d$ 处）开始读取 $F$（加密长度）个数据进行脱密，得到 $F$ 个明文数据，将这 $F$ 个明文数据中的前 $c$ 个与 $L$ 段长的明文数据一起构成写入数据 $L1$，即：$L1$ 的长度 $=c+L$ 的长度，$c$ 个新脱密的数据在 $L$ 个原数据的前面，$L1$ 的起点——即起始偏移量为 $d$，$L1$ 的结束点——即结束偏移量与 $L$ 的结束点相同，为 $b$。

② 如果是，则新的写入数据 $L1$ 的起点——即起始偏移量与原数据 $L$ 的起点一样，为 $d=a$，$L1$ 的长度 $=L$ 的长度，$L1$ 的结束点——即结束偏移量与 $L$ 的结束点相同，为 $b$。

通过以上操作确定出新的写入数据 $L1$ 的起点为 $d$。然后再确定结束点 $s$，首先判断需要写入的数据 $L1$ 的长度是否为加密长度 $F$ 的整数倍：

③ 如果不是（假设少 $g=f-b$ 个数据），可能会有三种情况：

如果从 $b$ 点再向后移动 $g$ 个数据到达 $f$ 点，仍未超出文件结束位置（即 $f$ 点在文件内），则需要从（$b-b\%F$）位置——即图 2 中 $e$ 处开始读取 $F$（加密长度）个数据（到达 $f$ 处）进行脱密，得到 $F$ 个明文数据，将这 $F$ 个明文数据中的后 $g$ 个与 $L1$ 段长的明文数据一起构成写入数据 $L2$，即：$L2$ 的长度 $=L1$ 的长度 $+g$，$g$ 个新脱密的数据在 $L1$ 个原数据的后面，$L2$ 的起点——即起始偏移量与 $L1$ 的相同，为 $d$，$L2$ 的结束点 $s$——即结束偏移量为 $s=f$。

如果从 $b$ 点再向后移动 $g$ 个数据到达 $f$ 点，已超出文件结束位置（即 $f$ 点在文件外，$b$ 点后面只有 $x$ 个数据，$x<g$），则将结束偏移量 $b$ 向前移动至（$b-b\%F$）位置——即图 2 中 $e$ 处，将 $e$ 点作为新的写入数据 $L2$ 的结束点——即结束偏移量为 $s=e$，$L2$ 的长度 $=L1$ 的长度 $-(F-g)$，$L2$ 的起点——即起始偏移量与 $L1$ 的相同。对剩余的（$F-g+x$）个明文数据进行填充然后再加密处理（见后面说明）。

如果 $b$ 点已超出文件结束位置，则将结束偏移量 $b$ 向前移动（$F-g$）个到达 $e$ 点，将 $e$ 点作为新的写入数据 $L2$ 的结束点——即结束偏移量为 $s=e$，$L2$ 的长度 $=L1$ 的长度 $-(F-g)$，$L2$ 的起点——即起始偏移量与 $L1$ 的相同。对剩余的（$F-g$）个明文数据进行填充然后再加密处理（见后面说明）。

④ 如果是，则新的写入数据 $L2$ 的起点——即起始偏移量与原数据 $L1$ 的起点一样，为 $d$，$L2$ 的长度 $=L1$ 的长度，$L2$ 的结束点——即结束偏移量 $s$ 与 $L1$ 的结束点相同，为 $s=b$。

通过以上操作确定出最终的写入数据 $L2$ 的起点为 $d$、终

点为 $s$、长度为 $L2$，$L2$ 的起点、终点偏移量均为加密长度 $F$ 的整数倍。

最后将 $L2$ 段明文加密后写入磁盘阵列。

现在讨论一下对可能剩余的 $F-g+x(0<(g-x)<(F,x)$ 0 或 $x=0$)个明文数据进行填充后加密：由于这部分数据的长度小于加密长度 $F$，也就是说，如果 $F$ 为 16 个字节，那么这部分数据最多为 15 个字节，这时需要填充 $(g-x)$ 个数据，填充后进行加密处理，并将 $(g-x)$ 写入文件结束的一个字节。无论需不需要填充，文件都需要增加一个字节保存填充的个数，不填充时为 0，填充时为填充的个数，脱密时根据填充的个数去除填充数据。

**结束语** 刀片加密服务器使用户免除后顾之忧；通过服务器加脱密技术使人们开发、应用因特网的商机成为可能；数据加密技术的应用非常广泛。可以预见，随着时代的发展，刀片加密服务器的加脱密技术将会朝着更快速、更安全、更经济的方向前进。

## 参 考 文 献


1 马琳茹. 基于 Linux 平台加密服务器设计与实现[D]. 北京：中国人民解放军国防科学技术大学，2002

2 谭华，沈建军. PSTN 网络智能优化的思考[J]. 电信网技术，2005(5)：19～22

3 王刚. 网络磁盘阵列结构和数据布局研究[D]. 天津：南开大学，2002

4 乐奕平，王辉球. 我国中小企业电子商务发展新模式[J]. 商业研究，2004(22)：179～181

5 夏恒，徐向阳，李仁发. 一种安全实用的密钥管理系统的研究与实现[J]. 计算机应用与软件，2005(12)：113～115


(上接第 76 页)

2)先运行服务器端程序，再运行客户端程序，利用 sniffer 软件监视来自客户机的所有 TCP 包，客户端记录发送时间，根据 sniffer 得到的接收时间和客户端的发送时间得到每次的传输时间值，统计得到平均延迟时间。

3)利用 sniffer 软件解析 TCP 包，得到窗口值，并统计往返时间($RTT$)，接受端通知最大窗口值($W_{max}$)。利用这些参数代到式(13)，(14)计算出平均延迟时间。

测试中，传感器数据采用周期发送，周期为 2 秒。

### 5.2 实验结果及分析

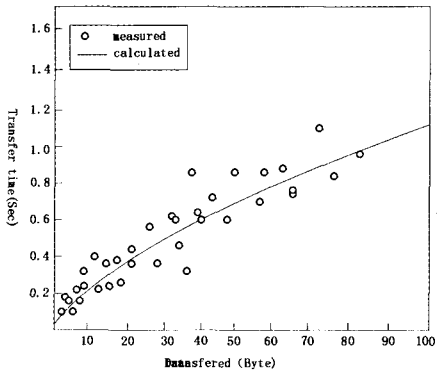


图 4 报警数据传输延迟与模型对比曲线

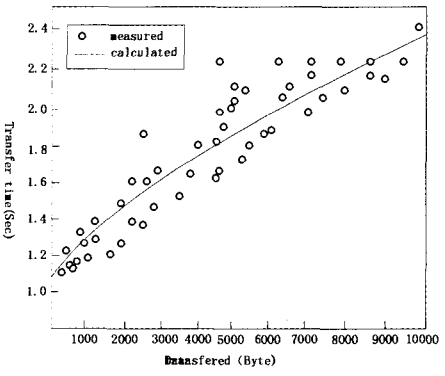


图 5 传感器数据传输延迟与模型对比曲线

图 4～6 分别对应报警数据、传感器数据、编程数据的测量值和模型计算值的曲线。图 4 中测量的数据很好地跟踪了模型曲线，图 5 中数据也较好地跟踪了模型曲线，图 6 中测量数据在较大数据流时，测量数据和模型曲线产生了较大的误差，原因是模型当中没有考虑到慢启动阶段的数据丢失情况，如果数据丢失，则根据图 1，TCP 应该进入快速重传或超时重传阶段，这时执行的拥塞控制算法将不再是慢启动算法，因此影响到了模型的准确性。在下一阶段的建模研究过程中，对于较大数据流将考虑数据丢失情况下的 TCP 模型。总之，可以看出，本文提出的 TCP 延时模型能够较好地反映网络化控制中的主要数据流类型的传输时间，可以作为仿真模型和实际模型使用。

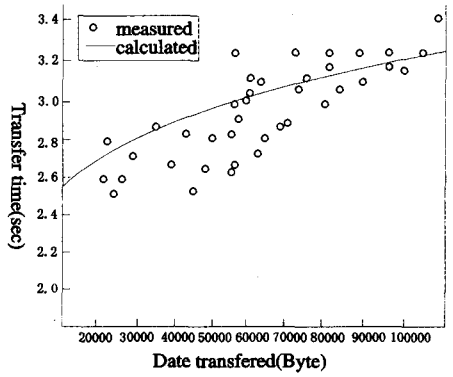


图 6 编程数据传输延迟与模型对比曲线

**结论** 本文从网络化控制系统中数据流特点入手，利用解析的方法建立了数据流传输的延迟模型，通过实验的方法，检验了模型的准确性，同时也指出了模型对长流量数据需要改进的方向。本文给出的模型和方法可以作为研究网络化控制系统中 TCP 建模的的参考模型，为进一步研究网络化控制中的 TCP/IP 通信建立了理论基础。

## 参 考 文 献


1 Walsh G C, Beldiman O, Bushnell L. Error encoding algorithms for networked control systems. In: Proc. of the 38th Conf. on Decision & Control Phoenix, Arizona USA, 1999. 4933～4938

2 杨丽曼，李运华，袁海斌. 网络控制系统的时延分析及数据传输技术研究[J]. 控制与决策，2004，(4)：361～382

3 谢希仁. 计算机网络[M]. 北京：电子工业出版社，2003

4 Cardwell N, Savage S, Anderson T. Modeling TCP Latency. IEEE INFOCOM 2000, Tel Aviv, Israel, 2000. 1724～1751

5 Mellia M, Stoica I, Zhang H. TCP Model for Short Lived Flows. IEEE Communicaion Letters, 6(2): 85～87

6 Pack S, Ahn S, Choi Y. Wireless TCP Model for Short-Lived Flows. IEEE, 1725～1729

7 赵炯，周其刚，张树京. TCP 启动阶段的建模研究. 计算机工程，2003，29(8)：19～21

8 Floyd S, Headerson T. The NewReno modification to TCP's fast recovery algorithm. RFC 2582, April 1999