基于免疫安全存储设备IBSSD的研究与实现

蔡 涛 鞠时光 牛德姣

(江苏大学计算机学院 镇江 212013)

摘 要 基于智能磁盘的安全存储设备是当前安全存储系统研究的热点问题。为解决现有安全存储设备I/O性能低的问题,引入人工免疫算法,实现高效的访问控制模块。首先给出基于免疫安全存储设备的结构,以及基于免疫访问控制模块中主要元素的定义,针对存储设备的特点,设计了差异选择算法和混合检测算法。实现了基于免疫访问控制的原型系统,验证了系统能高效地识别非法数据访问请求。修改开源存储区域网系统-Lustre中智能磁盘模块的代码,构建基于免疫安全存储设备的原型系统,测试了I/O性能。结果验证了基于免疫安全存储设备的I/O损失较小,能构成较高性能的安全存储系统。

关键词 存储安全系统,安全存储设备,人工免疫算法

IBSSD: Immune Based Secure Storage Device

CAI Tao JU Shi-guang NIU De-jiao

(Computer Department, Jiangsu University, Zhenjiang 212013, China)

Abstract The secure storage device is a hot topic in current secure storage system researching. In order to improve the I/O performance of current secure storage device, we used artificial immune algorithm to research efficient access control system for it. The structure of immune based secure storage device and some definitions of element in immune based access control system were given. By analyzing the characters of secure storage device, we presented diversity selection algorithm and mixed checking algorithm. Realizing the prototype of immune based access control system and testing the efficiency of it, the result shows it can inspect illegal access request efficiently. Realizing the prototype of immune based secure storage device in Lustre and evaluating its I/O performance. The result proves that the immune based secure storage device has high I/O performance and can be used for efficient secure storage system.

Keywords Secure storage, Secure storage device, Artificial immune algorithm

1 引言

安全存储设备是当前安全存储技术研究中的热点问题。通过在智能存储设备中内置安全模块,使得存储设备不需依赖外部安全服务器,能主动保护存储设备中数据的安全性,识别跳过上层安全保护系统的攻击,消除安全存储系统的性能瓶颈,减少安全存储系统I/O性能的损失。典型系统包括NASD[1-6]和Self Securing Storage[7-12],其中使用多版本文件系统、入侵检测和审计等安全技术,但存在安全开销大、降低存储系统利用率和安全开销变化大等问题,给存储系统带来的I/O性能损失在25%以上。我们引入人工免疫算法,研究面向存储设备的新型访问控制机制,实现基于免疫的安全存储设备(IBSSD),用于构建高效的安全存储系统。

人工免疫算法模拟生物免疫系统抵抗病毒和细菌等病原体的机制,已被用于网络安全、模式识别和数据挖掘等领域。Forrest在2000年实现了检测Linux中系统调用合法性的pH(process Homeostasis)系统[13],实验显示仅造成Linux系统性能5%的下降,验证了人工免疫算法具有很高的效率。目前人工免疫算法主要用于网络入侵检测系统中,Dasgupta在1999年建立了第一套基于免疫的计算机安全系统[14];张衡等为提高检测器对非自体的覆盖率,提出了r可变阴性选择算法[15];清华大学的孙照焱等人使用多层免疫模型研究保护附网存储系统的方法,但未考虑存储系统的特殊性,使用通用的人工免疫算法,存在检测效率偏低等问题,同时附网存储系统存在的I/O性能瓶颈等缺陷,也使得该系统的I/O性能损失较大[16]。目前还未见到将人工免疫算法用于实现研究存储设备中访问控制机制的报道。

本文的内容组织如下:在第2节给出IBSSD的结构,第3节给出主要元素的定义,第4节介绍基于免疫访问控制模块的实现,在第5节介绍基于免疫访问控制的原型系统的设计与分析,在开源存储区域网系统-Lustre上实现基于免疫安全存储设备的原型系统,测试对存储系统I/O性能的影响。

2 基于免疫安全存储设备的结构

存储设备主要包括4个方面的功能:接收数据访问请求、分析数据访问请求、执行命令和返回数据。我们使用人工免

到稿日期:2008-01-15 本课题获得国家自然科学基金(No. 60573046),江苏省自然科学基金(No. BK2007086)的资助。

蔡 涛(1976-),男,博士研究生,讲师,CCF会员,主要研究领域为安全存储系统、存储系统,E-mail:caitao@ujs.edu.cn;鞠时光(1955-),男,博士,教授,主要研究领域为信息安全;牛德姣(1978-),女,讲师,研究方向为信息安全、存储系统。

疫算法实现访问控制模块，检查数据访问请求是否合法，保护存储设备的安全性。图1给出了基于免疫安全存储设备以及现有存储设备的结构。在执行命令前，增加访问控制模块，检查数据访问请求的合法性，决定是否允许执行。

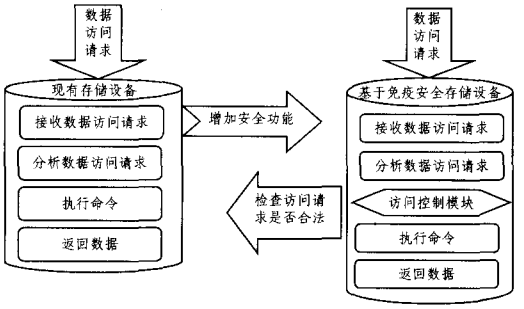

图1 基于免疫安全存储设备的结构图

基于免疫安全存储设备的关键是访问控制模块，下面我们给出其中主要元素的定义和关键算法。

3 主要元素的定义

访问控制模块中主要元素的定义如下：

定义1 论域 $X=\{0,1\}^l$，为访问控制模块中所有长度为 l 的二进制串组成的集合。

定义2 访问串 x 是对应存储设备中数据访问请求的长度为 l 的二进制串。

推论1 显然有 $x\in X$ 成立。

定义3 自体集 $S\subseteq X$，表示所有合法访问串组成的集合。

定义4 非自体集 $NS\subseteq X$，表示所有非法访问串组成的集合。

推论2 显然有 $S\cup NS=X$ 和 $S\cap NS=\phi$ 成立。

定义5 检测器 $d=(d_1,d_2,\cdots,d_l,r)$，$d_i\in\{0,1\}$，为一个长度为 l 的二进制串，r 的定义见定义6。

推论3 显然有 $d\in X$ 成立。

定义6 匹配阈值 r，$r\in N$，是判断检测器与访问串是否匹配的标准。当匹配度超过 r 时，则检测器与访问串匹配。

定义7 特征子串 m_i 是检测器 d 中长度为 i 的子串。

4 基于免疫访问控制模块的实现

我们首先分析存储设备对访问控制的要求，给出访问控制模块的组成，再介绍实现访问控制模块的主要算法。

4.1 访问控制要求的分析

现有人工免疫算法主要用于网络入侵检测等系统中，能高效保护静态系统的安全性。但存储设备中需保存的数据量非常巨大，数据会不断改变，破坏了现有人工免疫算法高效运行的基础。因此分析存储设备运行的特点，研究新型人工免疫算法是实现高效访问控制模块的关键问题。

安全存储设备中的访问控制模块在运行时只能收集到部分的合法数据访问请求，因此只能依据已知的自体选择检测器；存储设备在运行中会增加或删除部分数据，造成自体集的改变，出现新增和被删除的自体。由此可将自体分为4类：

1）完整的自体集 S：所有合法访问串的集合；

2）已知的自体集 SN：已知合法访问串的集合；

3）新增的自体集 SA：新增合法访问串的集合；

4）被删除的自体集 SD：被删除合法访问串的集合。

显然有推论4成立。

推论4 $SN\subseteq S$，$SA\subseteq S$ 且 $SD\subseteq S$

SN 是自体集 S 的抽样。根据局部性原理，SN 邻近区域存在未知合法访问串的几率较大。为减小误检率，挑选出的检测器与 SN 间的匹配度应小于匹配阈值，我们定义选择阈值 r_s 作为选择标准（见定义8）。

定义8 选择阈值 r_s，$r_s\in N$ 且 $1\leqslant r_s\leqslant r$。

4.2 访问控制模块的组成

基于免疫访问控制模块包括安全信息提取、安全信息转换、检测器挑选和检测访问串等子模块，结构如图2所示。安全信息提取子模块提取与检查数据访问请求合法性相关的信息，安全信息转换子模块则负责将提取出的安全信息转换成二进制字符串。这两个子模块实现数据访问请求中安全信息的提取和转换功能，构建与数据访问请求对应的二进制访问串。

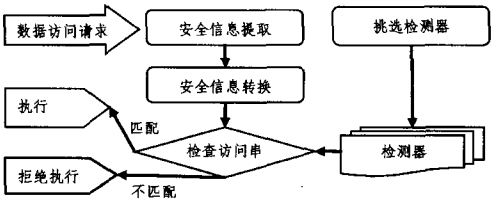

图2 基于免疫访问控制模块的组成图

检测器挑选子模块负责挑选用于检查访问串合法性的检测器，构建检测器集。检测访问串子模块负责检查有无检测器与访问串匹配：如有，则判断该访问串对应的数据访问请求为非法，拒绝执行，保护存储设备中数据的安全性；否则判断访问串对应的数据访问请求为合法，执行相应的数据访问请求。

检测器选择算法和访问串检测算法是访问控制模块中的两个关键算法。下面我们首先分析现有的算法，针对存储设备的特点，设计新型算法。

4.3 差异选择算法

目前主要的检测器选择算法有 Forrest 提出的否定选择算法[17]、Seidan 和 Celada 提出的肯定选择算法[18]、Leandro 提出的克隆选择算法[19]等，需要在运行前收集完整的自体集，通过与自体集的比较选择检测器，这种方式能很好地在自体集不变或很少变化的情况下构建检测器集，一般用于入侵检测等领域。存储设备中，基于免疫访问控制系统在运行前无法获得全部自体数据，挑选出的检测器存在较大误差。存储设备中数据的改变会造成自体集变化，带来检查误差。现有研究中使用重新生成检测器集和免疫反馈两种方法更新检测器集，适应系统的变化。前者在每次自体集变化后都需要重新构建检测器集，开销大；后者在自体改变后需要较长时间才能更新检测器集，系统存在较长时间的检测盲区，影响了系统的安全性。我们设计差异算法，由 SN 挑选检测器，生成能识别新增和被删除自体的差异检测元，提高访问控制模块适应自体集变化的能力。下面我们分别给出由 SN 挑选检测器和生成差异检测元的流程。

4.3.1 挑选检测器的流程

差异选择算法根据 SN，使用选择阈值 r_s 挑选检测器，构

建检测器集，算法流程如图 3 所示。

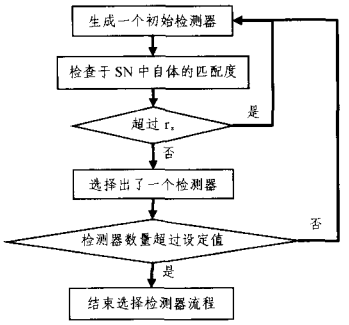

图 3　选择检测器流程

4.3.2　*生成差异检测元的流程*

当存储设备中的数据发生变化时，将新增和被删除自体（总称为差异自体）分别保存于 *SA* 与 *SD* 中，生成能识别差异自体的差异检测元，快速适应自体的变化，避免重新选择检测器的大量开销，缩短检测漏洞和误检存在的时间。识别新增和被删除自体的，仅需特征子串和起始位置两个因素，我们用三元组给出差异检测元的定义如下。

定义 10　差异检测元 $(m_i, position, type)$，m_i 表示一个长度为 i 的特征子串，$position$ 为 m_i 用于识别差异自体时的起始位置，$type$ 表示差异检测元的类型（0 表示用于识别新增的自体、1 表示用于识别被删除的自体）。

差异检测元要高效地识别差异自体，它们之间应具有较高的匹配度，我们定义生成阈值作为判断它们之间是否匹配的标准。

定义 8　生成阈值 $r_n, r_n \in N$ 且 $r \leqslant r_n \leqslant l$，差异检测元与新增和被删除自体之间的最小匹配度。

生成差异检测元的算法流程如图 4 所示。

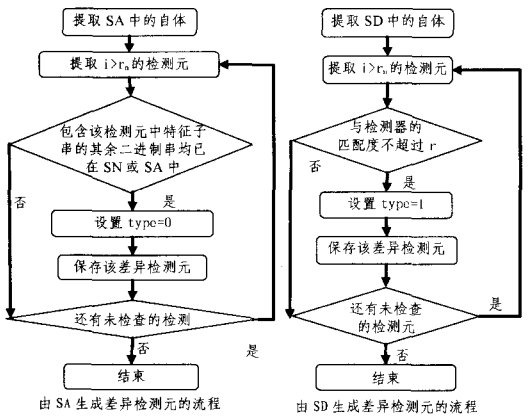

图 4　生成差异检测元的流程

当访问控制系统空闲后，使用 *SA* 和 *SD* 更新 *SN*，清空检测器集，按照 4.3.1 的流程重新选择检测器。

4.4　混合检测算法

检查访问串时，使用匹配算法计算检测器与访问串间的匹配度。如超过系统设置的匹配阈值，则判断该访问串对应的数据访问请求为非法。目前主要的匹配算法有 1994 年 Forrest 提出的 r-contiguous 匹配算法[20]、2002 年 Balthrop 提出的 r-chunk 匹配算法[21]、Farmer 在 1996 年提出的 Hamming 距离匹配算法[22]及其多种变形等，Harmer 在 2002 年分析了匹配算法的检测效率和检测率等问题[23]。存储设备中的数据访问请求通常包含设备标识、操作命令和数据标识等信息，现有匹配算法使用单一匹配值计算方法，无法适应数据访问请求的数据特点和检测要求；单一以对应相同子串或对应相同位作为判断条件，不能保证全面检查访问串中的各类信息，检测效率偏低。检查访问串时关键是计算检测器与访问串间的匹配度，现有匹配算法存在匹配度计算方法单一、无法适应存储设备中访问串的特点和检测要求，检测效率和准确性偏低。我们引入组分区分方法，分析访问串中不同组分的数据特点和检测要求，设计混合匹配度计算法，满足对不同组分的不同检测要求，提高检测的准确性和效率。

存储设备中需要保存大量数据，对应的数据标识数量庞大。为了提高基于免疫安全存储设备的 I/O 性能，要求检查该部分时有较高的检测效率。存储设备中设备个数有限，对应设备标识的数据量不大，但对判断访问串的合法性很重要，因此检查设备标识时应有很高的准确性。存储设备中操作命令的种类有限，但对于判断访问串的合法性很重要，需要较精确地检查访问串中的操作命令部分。我们将访问串分成动作和对象两个组分，动作组分包含访问串的设备标识和操作命令两个部分，检查时应具有很好的准确性；对象组分包含访问串中的数据标识，检测时应具有较高的检测效率。我们定义混合检测算法中，当检测器 d 与访问串 x 满足式(1)时，两者匹配：

$$d \text{ matches } x \equiv r_{p1} + r_{p2} \geqslant r \text{ and } r_{p1} \geqslant r * \alpha \text{ and } r_{p2} \geqslant r * \beta \tag{1}$$

其中 α 和 β 是两个参数，满足 $\alpha+\beta=1$。

式(1)包含两个条件：最小匹配度条件和最小比重条件，当两者同时成立时判断检测器与访问串匹配。

最小匹配度条件：$r_{p1}+r_{p2} \geqslant r$，所有组分子匹配度之和不小于系统设置的匹配阈值。

最小比重条件：$r_{p1} \geqslant r * \alpha$ 且 $r_{p2} \geqslant r * \beta$，组分子匹配度所占匹配阈值的最小比重。

r_{p1} 是访问串中动作组分与检测器对应子区间的子匹配度，为动作组分与检测器对应子区间对应相同子串的最大长度，r_{p1} 由式(2)计算。

$$\max\{r_k \mid \forall r_k\}\ r_k \equiv \exists i \leqslant l - r_k + 1 \text{ such that } x_j = d_j \text{ for } j = i, \cdots, i + r_k - 1 \tag{2}$$

r_{p2} 是访问串中对象组分与检测器对应子区间的子匹配度，为对象组分与检测器对应子区间相同子串的最大长度，r_{p2} 由式(3)计算。

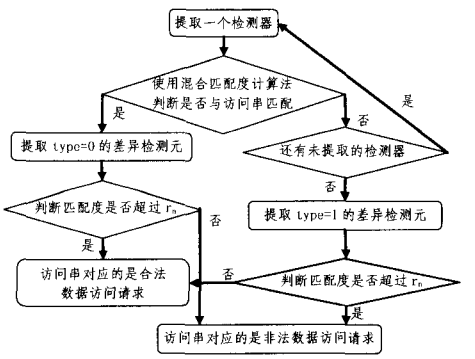

图 5　混合检测算法的流程

$$\max\{r_k \mid \forall r_k\}\ r_k \equiv \exists i \leqslant l - r_k + 1 \text{ and } \exists j \leqslant l - r + 1 \text{ such that } x_m = d_l$$

for $m=i,\cdots,i+r_k-1$ and $l=j,\cdots,j+r_k-1$ (3)

检查访问串时，不仅要判断是否有检测器与访问串匹配，还要判断是否有差异检测元与访问串匹配。我们设计混合检测算法，算法流程如图 5 所示。

5 原型系统的实现与分析

我们首先实现基于免疫访问控制模块的原型系统，测试检查非法访问串的效率。并在开源存储区域网系统 Lustre 上，实现基于免疫安全存储设备的原型系统，测试基于免疫访问控制模块对存储系统 I/O 性能造成的影响。

5.1 基于免疫访问控制模块原型系统的实现与分析

我们在 Linux 平台上实现了基于免疫访问控制模块的原型系统，使用 8 位二进制串表示访问串、自体和检测器，将访问串分成长度相等的动作和对象组分，设 r_{p1} 和 r_{p2} 的值相同为匹配阈值的一半，α 和 β 均为 0.5。使用枚举法生成 256 个各异的 8 位二进制串，分别顺序选取数量为 0，8，16，32 ，64，128，192，224 和 240 个二进制串作为自体。将其余的二进制串作为访问串写入访问串输入文件中，使得自体文件与访问串输入文件中的二进制串互补，此时所有访问串均为非法。原型系统输出每个访问串的检查结果，统计非法访问串的数量，与识别所需的检测器数量相比较，结果如图 6 所示。

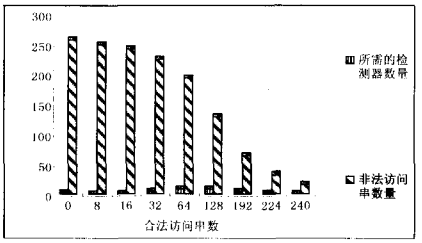

图 6 非法访问串数量与识别所需检测器数量的比较

图 6 的测试结果表明，应用数量不同的自体文件时，基于免疫访问控制原型系统识别非法访问串所需检测器数量远小于非法访问串的数量，保证了基于免疫访问控制原型系统的效率；所需的检测器数量变化较小，保证了系统开销的稳定。

5.2 安全存储设备原型系统的实现与分析

Lustre 是开源存储区域网系统，运行于 Linux 平台，由主机(Client)、元数据服务器(MDS)和面向对象存储目标器(OST)3 个部件组成。原型系统中 3 个模块运行于同一台计算机中，配置如表 1 所示。

表 1 原型系统的配置

操作系统	Red Hat Linux 企业版 4
Lustre 版本	1.4.8
CPU	PIV2.0G(双核)
内存	512M
硬盘	SATA 10000 转
测试工具	Iozone

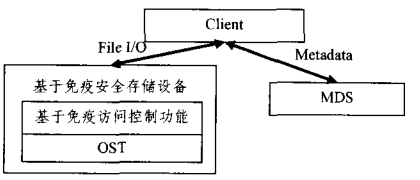

图 7 实现基于免疫安全存储设备后的 Lustre 系统结构

我们修改 Lustre 中 OST 模块的代码，增加基于免疫的访问控制模块，实现基于免疫安全存储设备的原型系统，此时 Lustre 系统结构如图 7 所示。

首先建立 PROC 通道，读取已知的部分自体信息。在 OST 模块中增加实现访问控制模块的头文件，设置匹配阈值为 8，α 和 β 的值均为 0.5，r_{p1} 和 r_{p2} 相同为 4，系统中保存 32 个检测器；实现安全信息提取和转换函数，提取数据访问请求中的操作命令、主机标识和数据对象标识等信息，分别转换成动作组分和对象组分两个 8 位二进制串，构成 16 位的二进制访问串。最后修改 OST 模块中接收数据访问请求函数 OST_Handle 的代码，在执行数据访问请求前调用安全信息提取函数和转换函数，构建访问串，使用基于免疫的访问控制模块判断访问串的合法性，决定是否允许对应的数据访问请求执行。

使用 Iozone 做为测试工具，使用大小为 4k，8k，16k，32k，64k，128k，256k 和 512k 数据块读 512k 的文件，使用大小为 4k，8k，16k，32k，64k，128k，256k，512k 和 1024k 的数据块写 1M 的文件，测试实现基于免疫安全存储设备前后，Lustre 系统的 I/O 性能，测试结果如图 8 所示。

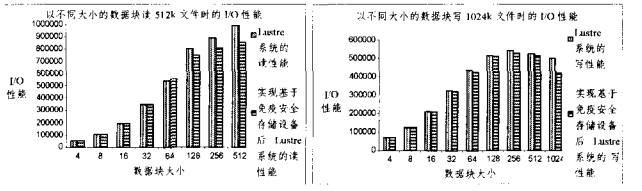

图 8 实现基于免疫安全存储设备前后 Lustre 系统的 I/O 性能

从图 8 可以发现，实现基于免疫访问控制模块后，Lustre 系统的读性能下降在 10%以内，写性能下降在 8%左右，表明基于免疫访问控制模块具有很高的效率，能保持较高的存储系统 I/O 性能。

结束语 本文引入人工免疫算法，实现存储设备中的访问控制模块，构成安全的存储设备。首先给出基于免疫安全存储设备的结构，针对存储设备的特点，设计差异选择算法，快速适应存储设备的数据变化，避免检测漏洞和误检，缩短存在检测盲区的时间；设计混合检测算法，针对存储设备中数据访问请求的特点，提高检查数据访问请求的效率和准确性。实现了基于免疫访问控制模块的原型系统，验证了基于免疫的访问控制模块能高效地识别非法数据访问请求，完成存储设备的访问控制功能。最后在开源存储区域网系统-Lustre 上，实现了基于免疫安全存储设备的原型系统，测试对存储系统 I/O 性能的影响。结果表明，基于免疫安全存储设备能保持较高的存储系统 I/O 性能。

目前所实现的基于免疫安全存储设备中的访问控制模块，未优化检测器的生成机制，易造成较大的检测漏洞和误差。下一步我们准备引入数据挖掘技术，根据已知的合法数据访问请求优化检测器的生成，保证检测器对非法数据访问请求的识别能力。

参 考 文 献

[1] Gibson G A, Van Meter R. Network Attached Storage Architecture. Communications of the ACM, 2000, 43(11)

(下转第 137 页)

立，然而，根据式(5)，可得 $I^4(A,B)=(0.6,0.4)$，因此，$I^4(A,B)$不是直觉模糊包含度，也不是模糊包含度。

结束语 包含度在模糊集理论中有着重要应用。在应用模糊集理论处理实际问题时，模糊集的包含关系过于苛刻，一般必须用模糊包含度加以代替。直觉模糊包含度是对模糊包含度的直觉化扩展，本文针对文献[7]中 Vague 包含度定义仍然取值于区间[0,1]这一问题，重新定义了直觉模糊集的包含度，并验证了 4 类直觉模糊包含度公式 I^0-I^3。其中，由于直觉模糊 R-蕴含的良好性质，使得直觉模糊 R-蕴含可以生成一类直觉模糊包含度，而直觉模糊 S-蕴含则不能生成直觉模糊包含度，文中给出了反例验证。课题下一步计划将直觉模糊包含度引入模糊推理与基于粗糙集的决策分析，对其应用进行拓展研究。

参 考 文 献

[1] 张文修，徐宗本，梁怡，等. 包含度理论[J]. 模糊系统与数学，1996，10(4)：1-9

[2] 张文修，梁怡，徐萍. 基于包含度的不确定推理[M]. 北京：清华大学出版社，2007

[3] 曲开社，翟岩慧. 偏序集、包含度与形式概念分析[J]. 计算机学报，2006，29(2)：219-226

[4] Atanassov K. Intuitionistic fuzzy sets. Fuzzy Sets and Systems，1986，20：87-96

[5] Atanassov K. Intuitionistic Fuzzy Sets：Theory and Applications. Heidelberg，Germany：Physica-Verlag，1999

[6] Burillo P，Bustince H. Vague sets are intuitionistic fuzzy sets [J]. Fuzzy Sets and Systems，1996，79(3)：403-405

[7] 黄国顺，刘云生. 基于包含度的 Vague 集相似度量[J]. 小型微型计算机系统，2006，27(5)：873- 877

[8] Cornelis C，Deschrijver G，Kerre E E. Implication in intuitionistic fuzzy and interval-valued fuzzy set theory：construction，classification，application[J]. International Journal of Approximate Reasoning，2004，35(1)：55-95

[9] 路艳丽，雷英杰，田野. 直觉模糊逻辑算子研究[J]. 计算机科学，2008，35(11)：151-153

(上接第 104 页)

[2] Amiri K S. Scalable and Manageable Storage Systems. Ph. D. Dissertation，CMU-CS-00-178. Carnegie Mellon，December 2000

[3] Gobioff H. Security for a High Performance Commodity Storage Subsystem. Ph. D. Dissertation，CMU-CS-99-160. Carnegie Mellon，July 1999

[4] Amiri K，Gibson G A，Golding R. Highly Concurrent Shared Storage// Proceedings of the International Conference on Distributed Computing Systems. Taipei，April 2000

[5] Gobioff H，Nagle D，Gibson G. Embedded Security for Network-Attached Storage. technical report CMU-CS-99-154. CMU SCS，June 1999

[6] Gobioff H，Gibson G，Tygar D. Security for Network Attached Storage Devices. technical report，CMU-CS-97-185. CMU SCS，1997

[7] GoodsonG R，WylieJ J，GangerG R，et al. The Safety and Liveness Properties of a Protocol Family for Versatile Survivable Storage Infrastructures. Technical Report CMU-PDL-03-105. Carnegie Mellon University Parallel Data Laboratory，March 2004

[8] Pennington A，Strunk J，Griffin J，et al. Storage-based Intrusion Detection：Watching Storage Activity For Suspicious Behavior// 12th USENIX Security Symposium. Washington，D. C.，Aug. 2003

[9] Soules C A N，Goodson G R，Strunk J D，et al. Metadata Efficiency in Versioning File Systems//2nd USENIX Conference on File and Storage Technologies. San Francisco，CA mar 31-Apr 2，2003

[10] Strunk J D，Goodson G R，Pennington A G，et al. Intrusion Detection，Diagnosis，and Recovery with Self-Securing Storage. Technical Report，CMU-CS-02-140. CMU SCS，May 2002

[11] Ganger G R，Nagle D F. Better Security via Smarter Devices// HotOS-VIII (IEEE Workshop on Hot Topics in Operating Systems). May 2001

[12] Strunk J D，Goodson G R，Sheinholtz M L，et al. Self-Securing Storage：Protecting Data in Compromised Systems//4th Symposium on Operating System Design and Implementation. San Diego，CA，Oct. 2000

[13] Somsysji A，Forrest S. Automated Response Using System-Call Delays// Proceedings of the 9th USENIX Security Symposium. Denver，Colorado：USENIX ASSOC，SUITE 215，2560 NINTH ST，BERKELEY，CA94710 USA，2000：185-197

[14] Dasgupta D. Immune-based intrusion detection system：a general framework// Proceedings of the 22nd National Information Systems Conference. Virgina，USA，1999

[15] 张衡，吴礼发，张毓森，等. 一种 r 可变阴性选择算法及其仿真分析. 计算机学报，2005，28(10)

[16] 孙照焱. 基于生物免疫机制的附网存储关键技术研究. 博士学位论文. 北京：清华大学精密仪器与机械学系，2004

[17] Forrest S，Perelson A S，Allen L，et al. Self-Nonself Discrimination in a Computer// Proceeding of IEEE Symposium on Research in Security and Privacy. Los Alamitos，CA：IEEE Computer Society Press，1994：202-212

[18] Seiden P E，Celada F. A Model for Simulating Cognate Recognition and Research in the Immune System. J. theor. Biol，1992，158：329-357

[19] de Castro L N，Von Zuben F J. Learning and Optimization Using the Clonal Selection Principle. IEEE Transaction on Evolution Computation，2002，6(3)

[20] Forrest S，Perelson A S，Allen L，et al. Self-Nonself Discrimination in a Computer// Proceeding of IEEE Symposium on Research in Security and Privacy. Los Alamitos，CA：IEEE Computer Society Press，1004：202-212

[21] Balthrop J，Forrest S，Glickman M R. Revisting LISYS：Parameters and normal behavior// Proceedings of the 2002 Congress on Evolutionary Computation CEC2002. USA：IEEE Press，2002：1045-1050

[22] Farmer J D，Packard N H，Perelson A S. The immune system，adaptation，and machine learning. Physica D，1986，22：187-204

[23] Harmer P，Williams G，Gnusch P D，et al. An Artificial Immune System Architecture for Computer Security Applications. IEEE Transactions on Evolutionary Computation，2002，6(3)：252-280