面向存储安全系统的新型人工免疫算法*)

蔡 涛 鞠时光 仲 巍 牛德姣
(江苏大学计算机学院 镇江 212013)

摘 要 提出了新型人工免疫算法,用于研究高效的存储安全系统。首先给出了基于免疫存储安全系统的结构和相关定义。在分析人工免疫算法中已有匹配规则的基础上,为提高安全系统的效率,提出了任意 r 连续位匹配规则,提高检测器识别非自体的能力,减少存储安全系统识别非自体所需的成熟检测器数量;为了使存储安全系统能适应不同的自体集,自动优化检测效率和准确性,避免检测存储安全系统的失效,本文提出了自适应匹配阈值机制。分析了使用不同匹配规则时检测器能识别的最大非法访问请求数量,以及对不同自体集采用静态匹配阈值和自适应匹配阈值机制时存储安全系统的检测效率和准确性。使用新型人工免疫算法实现安全原型系统,验证了算法的性能。最后通过修改开源存储区域网系统 Lustre 中智能磁盘部分的源代码,实现了基于免疫安全磁盘的原型系统,测试增加存储安全系统前后 Lustre 系统的 I/O 性能,结果表明新型人工免疫算法能高效地保护存储系统的安全。

关键词 人工免疫算法,存储安全

New Artificial Immune Algorithm for Storage Security

CAI Tao JU Shi-guang ZHONG Wei NIU De-jiao
(College of Computer, Jiangsu University, Zhengjiang 212013, China)

Abstract On the basis of analyzing demand of storage security system and current artificial immune algorithm, this paper presents new artificial immune algorithm to ensure security of storage system efficiently. Main current matching rules are introduced to analyze the efficiency of current artificial immune algorithm. Firstly the structure of storage security system and the definition of main elements in it are given. To improve the efficiency of artificial immune algorithm, this paper proposes random r-continuous matching rule to improve the number of non-self that one detector can recognize. To avoid failure of storage secure system that no detector are self-tolerance and low efficiency of inspecting, this paper presents self-adaptable threshold selection algorithm to select suitable threshold for different detectors, and then balance between efficiency and accuracy for different self sets. Analyzing how much non-self one detector can recognize and whether the selection of threshold is adaptability. Using new artificial immune algorithm to implement the prototype of secure system and verify its performance. We implement prototype of new artificial immune algorithm. The evaluating result shows the new artificial immune algorithm has higher efficiency and is more adaptability than current artificial immune algorithm. At last, we modify the source code of storage area network system named Lustre and implement prototype of the secure disk system. By evaluating its I/O performance, the result shows new artificial immune algorithm can ensure the security of storage efficiently.

Keywords Artificial immune algorithm, Storage security

存储安全系统是当前研究的热点问题,目前的研究主要包括六个方面:增加文件系统的安全功能以保护存储系统安全,典型的系统包括 CFS[1],AFS[2],SFS[3,4] 和安全 NAS 系统[5]等;研究新型磁盘结构,典型系统包括 NASD[6] 和 Self Securing Storage[7,8];研究数据可恢复机制,典型系统包括 PASIS[9,10] 和 OceanStore[11];研究高效的密钥管理机制,提高存储系统的密钥管理效率,典型系统包括 SNAD[12-14]、PLUTUS[15] 和基于 iSCSI 的附网存储安全系统[16];研究存储系统的安全中间件,典型系统包括 SiRiUS[17] 和两层安全结构[18];研究存储系统的分区通讯和掩码机制,包括在主机[19]、交换机[20] 和存储控制器中的实现[21]。现有存储安全系统主要使用加密、认证、数据冗余和入侵检测等安全技术,存储系统保存的海量数据,使得安全开销很大,严重影响了存储系统的性能。高性能是存储系统的重要目标之一,要求存储安全系统的开销尽可能小,我们引入人工免疫算法,研究高效的存储安全系统。

人工免疫系统模拟生物免疫系统的运行机制,具有分布性、多层性、多样性、自治性和自适应性等特性,能高效地保护系统。目前的研究集中在非自体识别机制与免疫网络理论两大方面[22,23],其中最重要的理论之一是美国新墨西哥大学 Forrest 教授和她的小组在 1994 年提出的否定选择算法(NSA)[24],它模拟了 T 细胞成熟过程中的自体耐受机制。其中匹配规则用于自体耐受和免疫检测中判断抗体(检测器)与自体和抗原是否匹配,直接关系到人工免疫算法的效率和对非自体的识别率。因此研究高效的匹配规则是提高存储安全系统效率的关键问题。抗原与检测器有二进制字符串和向量两种表示方式,它们最终都以二进制串形式保存,因此匹配规则实际是比较两个二进制串的算法。

本文首先介绍现有人工免疫算法中的匹配规则,分析存在的问题;给出基于免疫存储安全系统的结构以及相关定义;

*)本文受到国家自然科学基金(60573046)、江苏省自然科学基金(BK2007086)资助。蔡 涛 博士生,研究方向为存储安全、存储系统;鞠时光 教授,博导,研究方向为信息安全;仲 巍 硕士生,研究方向为存储系统;牛德姣 讲师,硕士,研究方向为存储系统。

提出任意 r 连续位匹配规则和自适应匹配阈值选择机制，构成面向存储安全的新型人工免疫算法；通过理论分析和原型系统的测试，验证算法的效率和准确性；最后修改开源存储区域网系统-Lustre 的源代码，使用新型人工免疫算法构建安全系统，实现安全磁盘原型系统，测试安全系统的开销。

1 现有人工免疫算法中的匹配规则

匹配规则用于判断检测器与自体和抗原是否匹配，现有人工免疫中的匹配规则包括 r-contiguous，r-chunk，Hamming 距离和 R&T 距离等方法。下面我们以检查检测器与抗原是否匹配为例，给出它们的定义和性能分析。

1.1 r-contiguous 匹配规则

Forrest 和她的小组在 1994 年提出了 r-contiguous 匹配规则[24]，定义抗原 $x=x_1x_2\cdots x_l$ ($x_i\in\{0,1\}$)，检测器 $d=d_1d_2\cdots d_l$ ($d_i\in\{0,1\}$)，抗原 x 与检测器 d 匹配的定义如下：

d matches $x\equiv\exists i\leqslant l-r+1$ such that $x_j=d_j$ for $j=i,\cdots,i+r-1$

表示当二进制串 x 和 d 存在不少于 r 个连续相同的对应位时，抗原 x 与检测器 d 匹配；$r(1<r\leqslant l)$ 为静态匹配阈值。

r-contiguous 匹配规则能较好地保证检测准确性，但制约了检测器识别非自体的能力。每个长度为 l 的检测器包含 $l-r+1$ 个用于检测非自体的特征子串，每个特征子串最多能识别 2^{l-r} 种非自体，因此每个检测器最多能识别 $(l-r+1)2^{l-r}$ 种非自体。

r-contiguous 匹配规则中所有检测器均使用相同的静态匹配阈值 r，因此匹配阈值的选择非常重要。匹配阈值较小时，检测器的识别能力强，算法效率高；但匹配阈值过小会造成检测器与自体匹配，无检测器能通过自体耐受检查成为成熟检测器，造成系统失效。匹配阈值较大时，检测器识别非自体更准确，但匹配阈值过大使得识别非自体所需要的成熟检测器数量急剧增加，降低了系统效率。因此对不同自体集，静态匹配阈值很难同时满足对人工免疫算法检测效率和非自体检测率的要求。

1.2 r-chunk 匹配规则

Balthrop 为提高 r-contiguous 匹配规则的检测准确性，与 2002 年提出了 r-chunk 匹配规则[25]，定义抗原 $x=x_1x_2\cdots x_l$ ($x_i\in\{0,1\}$)，检测器 $d=(i,d_1d_2\cdots d_r)$ ($d_i\in\{0,1\}$)，设 $r\leqslant l$ 且 $i\leqslant l-r+1$，抗原 x 与检测器 d 匹配的定义如下：

d matches $x\equiv x_j=d_j$ for $j=i,\cdots,i+r-1$

表示当检测器 d 与抗原 x 的从第 i 位开始存在不少于 r 个连续相同的对应位时，两者匹配；$r(1<r\leqslant l)$ 为静态匹配阈值。

r-chunk 匹配规则实际是在 r-contiguous 匹配规则的基础上增加了限制条件，提高了检测的准确性。检测器中保存开始检查抗原的起始位 i，只检查抗原 x 的后 $1-i$ 位，限定了抗原 x 中的有效区域，提高了检测的准确性。这时每个检测器包含 $l-r-i+1$ 个特征子串，最多能识别 $(l-r-i+1)2^{l-r}$ 种非自体，与 r-contiguous 匹配规则相比降低了单个检测器能识别的非自体数。此外 r-chunk 匹配规则同样存在选择静态匹配阈值的困难，静态匹配阈值无法针对不同自体集优化人工免疫算法的效率和非自体的检测率。

1.3 Hamming 距离匹配规则

Farmer 于 1986 年提出了 Hamming 距离匹配规则[26]，定义抗原 $x=x_1x_2\cdots x_l$ ($x_i\in\{0,1\}$)，检测器 $d=d_1d_2\cdots d_l$ ($d_i\in\{0,1\}$)，抗原 x 与检测器 d 匹配的定义如下：

d matches $x\equiv\sum_i\overline{x_i\oplus d_i}\geqslant r$

表示当抗原 x 和检测器 d 中相同对应位的个数大于等于 r 时，两者匹配。$r(1<r\leqslant l)$ 同样为静态匹配阈值。

Hamming 距离匹配规则中，每个检测器最多能识别 $(l-r+1)2^{l-r}$ 种非自体；但仅仅统计对应位相同的个数，不考虑相同位之间的关系，存在较大的检测误差。和上述两种匹配规则相似，匹配阈值较小时，算法的检测效率高，但检测误差较大；此外当自体数较多时，存在无检测器能通过自体耐受检查生成成熟检测器所造成的系统失效。匹配阈值偏大时，检测的准确性较高，但需要的成熟检测器数量急剧增加，系统效率降低。同样静态匹配阈值无法针对不同的自体集，优化人工免疫算法对非自体的检测率和检测效率。

1.4 R&T 匹配规则

Rogers 和 Tanimoto 在 2002 年改进了 Hamming 距离匹配规则，提出了 R&T 匹配规则[27]，定义抗原 $x=x_1x_2\cdots x_l$ ($x_i\in\{0,1\}$)，检测器 $d=d_1d_2\cdots d_l$ ($d_i\in\{0,1\}$)，抗原 x 与检测器 d 匹配的定义如下：

$$d \text{ matches } x\equiv\frac{\sum_i\overline{x_i\oplus d_i}}{\sum_i\overline{x_i\oplus d_i}+2\sum_i x_i\oplus d_i}\geqslant r,$$

$r(0<r\leqslant 1)$ 为静态匹配阈值。

R&T 匹配规则与 Hamming 距离匹配规则相比提高了检测的准确性，但同样存在选择静态匹配阈值的困难。

2 新型人工免疫算法

2.1 存储安全系统的结构

存储系统负责为文件系统提供数据存储服务，响应各类数据访问请求，存储安全系统用于保证数据访问请求的合法性，包括成熟检测器生成和数据访问请求检查两大模块。存储系统运行前，安全系统首先收集存储系统中的合法数据访问请求，建立自体集，再使用检测器生成模块生成成熟检测器。存储系统运行时，由数据访问请求检查模块检查接收到的数据访问请求是否合法，再决定是否响应该数据访问请求。存储安全系统的结构如图 1 所示。

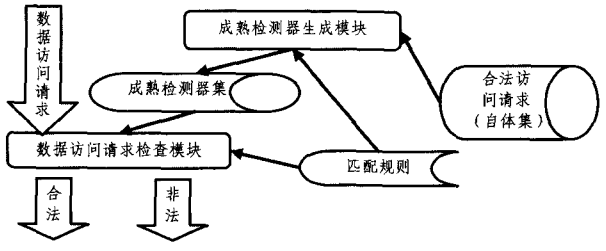

图 1 存储安全系统的结构

成熟检测器生成模块负责构建成熟检测器集，用于识别非法数据访问请求；数据访问请求检查模块负责检查每个数据访问请求的合法性，它们的算法过程在 2.4 节中给出。

2.2 存储安全系统中的定义

下面我们给出存储安全系统中有关概念的定义。

定义 1(论域) $U=\{0,1\}^l$ 表示所有长度为 l 的二进制字符串。

定义 2(自体集) $S\subseteq U$ 表示存储系统中所有合法数据访问请求的集合。

定义 3(非自体集) $NS\subseteq U$ 表示存储系统中所有非法

数据访问请求的集合。

推论 1 显然有 $S \cup NS = U$ 和 $S \cap NS = \phi$ 成立。

定义 4(数据访问请求) $x = x_1 x_2 \cdots x_l (x_i \in \{0,1\})$表示存储系统中的一个数据访问请求。

推论 2 显然有 $x \in U$ 成立。

定义 5(匹配阈值) r 是判断抗原 x 与检测器 d 是否匹配的标准。

定义 6(检测器) $d = (d_1 d_2 \cdots d_l, r)$ $(d_i \in \{0,1\})$为长度为 l 的二进制串,用于检查数据访问请求。r 为该检测器的匹配阈值。

推论 3 显然有 $d \in U$ 成立。

定义 7(特征子串) 检测器 d 中用于检查是否与数据访问请求匹配的 r 位子串。

2.3 任意 r 连续位匹配规则

存储安全系统用检测器包含的特征子串检查是否与数据访问请求匹配,判断其合法性,因此增加每个特征子串能识别的非法数据访问请求数量是提高存储安全系统效率的重要手段。r-contiguous 和 r-chunk 匹配规则比较在对应位置是否存在特征子串,Hamming 距离和 R&T 距离比较对应相同位的个数,两类方法均限制了特征子串能识别的非自体数量。本文提出任意 r 连续位匹配规则,成熟检测器能识别在任意位置出现特征子串的非法数据访问请求。对检测器 d 与数据访问请求 x,两者匹配的定义如公式(1)。

$$d \text{ matches } x \equiv \exists i \leqslant l-r+1 \text{ and } \exists j \leqslant l-r+1 \text{ such that } x_k = d_l \text{ for } k = i, \cdots, i+r-1 \text{ and } l = j, \cdots, j+r-1 \quad (1)$$

如图 2 所示,当匹配阈值为 4 时,数据访问请求 1 的前 4 位和数据访问请求 2 的后 4 位均与检测器 A 中的特征子串—0011 相同,根据任意 r 连续位匹配规则,判断检测器 A 与数据访问请求 1 和 2 均匹配。

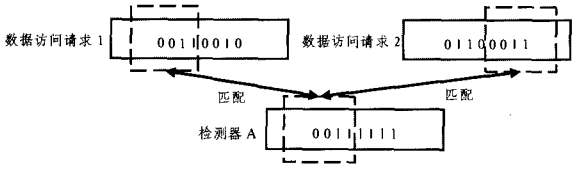

图 2 任意 r 连续位匹配规则示意图

2.4 自适应匹配阈值机制

现有匹配规则使用静态匹配阈值,不能根据自体集优化检测效率和对非自体的检测率,我们提出自适应匹配阈值机制解决这一问题。

首先改变成熟检测器共用同一静态匹配阈值的方法,给每个成熟检测器选取专用的匹配阈值,保存于成熟检测器中;在存储安全系统中保存初始匹配阈值 $r_s (r_s \in (1, l))$和最大匹配阈值 $r_m (r_m \in (1, l))$。修改成熟检测器生成算法,增加匹配阈值的选取功能。先使用穷举法[28]、线性法[29,30]或贪心法[31]等算法生成初始检测器集,用初始匹配阈值 r_s 和任意 r 连续位匹配规则挑选与自体不匹配的初始检测器,保存到成熟检测器集;若获得的成熟检测器数量少于系统设置值且匹配阈值小于最大匹配阈值 r_m,则增大匹配阈值,再次挑选成熟检测器,直到获得足够数量的成熟检查器或匹配阈值超过 r_m。算法流程如下。

```
Function r-generation(r_s, r_m)
begin
    for (i=r_s; i<=r_m; i++)
    begin
        while (能产生不重复的初始检测器)
        begin
            产生不重复的初始检测器;
            if (使用任意 r 连续位匹配规则和 i 检查该初始检测器
                是否与自体匹配)
                设置该初始检测器的匹配阈值为 i,保存到成熟检
                测器集;
            else
                continue;
            if (成熟检测器数量达到系统设定值)
                break;
        end
    end
end.
```

增加自适应匹配阈值选择机制后,存储安全系统首先使用较小的匹配阈值生成成熟检测器,优先生成具有较强检测能力的成熟检测器,保证了存储安全系统的效率;同时能自动根据不同的自体集,调整匹配阈值,防止出现无初始检测器能通过自体耐受所造成的存储安全系统失效,保证存储安全系统高效、准确地识别非法数据访问请求。

此外我们修改检查数据访问请求的流程,检查时依次取出成熟检测器集中的检测器,提取保存的匹配阈值,使用任意 r 连续位匹配规则判断是否与数据访问请求匹配;如匹配则判断该数据访问请求为非法,拒绝相应的操作;如数据访问请求不与任何成熟检测器匹配,则判断其为合法数据访问请求,提交给存储系统执行。

3 性能分析

面向存储安全系统的新型人工免疫算法用于提高现有人工免疫算法的效率和适应性,从而实现高效灵活的存储安全系统。下面我们比较使用不同匹配规则时单个检测器能识别的非自体个数,分析不同匹配规则的效率;同时比较静态匹配阈值机制和自适应匹配阈值机制适应不同自体集,优化检测效率和准确性的能力。

表 1 采用不同匹配规则时单个检测器识别的最大非自体数量

匹配规则	任意 r 连续位匹配规则	r-contiguous 匹配规则	r-chunk 匹配规则	Hamming 距离匹配规则
单个检测器能识别的最大非自体数	$(l-r+1)2^{l-r+1}$	$(l-r+1)2^{l-r}$	$(l-r-i+1)2^{l-r}$	$(l-r+1)2^{l-r}$

采用任意 r 连续位匹配规则时,长度为 l 的检测器包含 $l-r+1$ 个特征子串,每个特征子串能识别$(l-r+1)2^{l-r}$种非自体,每个检测器共能识别$(l-r+1)2^{l-r+1}$种非自体。表 1 是使用不同匹配规则时,单个检测器能识别的最大非自体数量,从中可以发现采用任意 r 连续位匹配规则使单个检测器能识别的最大非自体数量增加了至少一倍,大大减少了存储安全系统检测非法数据访问请求所需要的成熟检测器个数,提高了安全系统的效率。

检测器的匹配域值 r 越小,识别非法数据访问请求的能力越强,因此为提高存储安全系统的效率应使用较小的匹配阈值。但使用过小的静态阈值时,存在无检测器能通过自体耐受生成成熟检测器,造成存储安全系统失效的问题。使用自适应匹配阈值机制时,首先用较小的匹配域值生成成熟检测器,保证了生成高效的成熟检测器;一旦无检测器能通过自体耐受或生成的成熟检测器数量不足,则增大匹配阈值补充成熟检测器,保证存储安全系统能准确的识别非法数据访问请求。因此自适应匹配阈值机制能适应不同自体集,自动优化存储安全系统的检测效率和检测率,而静态匹配域值只能针对某一特定自体集优化存储安全系统的检测效率和检测率。

4 使用新型人工免疫算法实现安全原型系统的实现与测试

我们在 Linux 平台上使用新型人工免疫算法实现了安全原型系统，测试比较不同人工免疫算法的性能。原型系统使用 8 位二进制串表示数据访问请求和检测器，用两个文本文件保存自体和数据访问请求，采用穷举法生成初始检测器，不限制能通过自体耐受的成熟检测器数量；输出对数据访问请求的检查结果，统计识别全部非法数据访问请求所需的成熟检测器数量。

4.1 检测效率的测试与分析

分别使用不同匹配规则测试原型系统的性能，使用 R&T 匹配规则时初始匹配阈值为 0.1，最大值为 1，增量为 0.1，使用其它匹配规则时初始匹配阈值为 1，最大值为 8，增量为 1。测试前使用枚举法生成全部 256 个二进制 8 位串，顺序选取一定数量的 8 位串写入自体文件中，其余的放入数据访问请求文件中。我们建立二进制串数为 0,8,16,32,64,128 和 192 的自体集文件，测试原型系统识别全部非法数据访问请求所需要的最少成熟检测器数量，测试结果如图 3 所示。

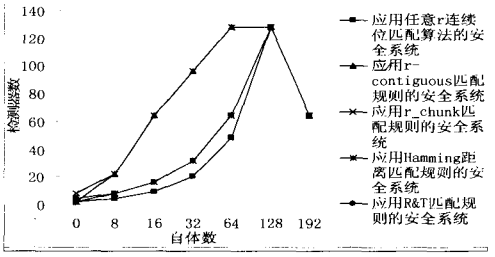

图 3 不同原型系统识别出全部非自体所需的最少成熟检测器数

从图 3 可以看出，采用新型人工免疫算法能明显减少识别全部非法数据访问请求所需的成熟检测器数量，提高了系统效率。所需的成熟检测器数量远小于存储系统中非法数据访问请求的数量，因此存储安全系统能高效的保证存储系统的安全。

4.2 适应性的测试与分析

分别使用静态匹配阈值和自适应匹配阈值机制测试原型系统的性能，建立二进制串数为 0,8,16,32,64,128 和 192 的自体集，设置静态匹配阈值为 4,5,6,7 和 8，测试识别全部非法数据访问请求所需的成熟检测器数量，结果如图 4 所示。

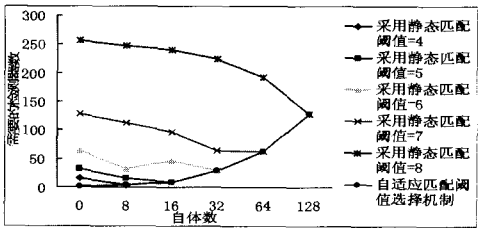

图 4 采用自适应与静态匹配阈值机制识别非自体所需的检测器数

从图 4 中可以发现使用静态匹配阈值时，当自体数达到一定值后，无初始检测器能通过自体耐受，生成成熟检测器，造成存储安全系统的失效。使用自适应匹配阈值机制则始终能生成成熟检测器，同时针对不同自体集，始终只需要最少数量的成熟检测器就能识别全部非法数据访问请求。因此自适应匹配阈值机制能针对不同自体集，优化检测效率和非法数据访问请求的检测率，有效地保证存储安全系统的效率和检测精度。

5 基于免疫安全磁盘原型系统的实现与测试

Lustre 是开源存储区域网系统，由 client，MDS 和 OST 三类模块组成。其中 OST 模块是面向对象的存储目标器，实现智能磁盘的功能。OST 模块使用 portals 协议与其它模块通讯，完成数据的读写操作，功能简单，但保存的数据量非常庞大，使用现有安全保护方法会带来很大的安全开销，严重影响存储系统的性能。我们修改 Lustre 系统中 OST 模块的源代码，使用新型人工免疫算法实现安全系统，检查 OST 接收到的数据访问请求的合法性，保护磁盘数据的安全，实现基于免疫的安全磁盘原型系统，修改后的 Lustre 系统结构如图 5 所示。

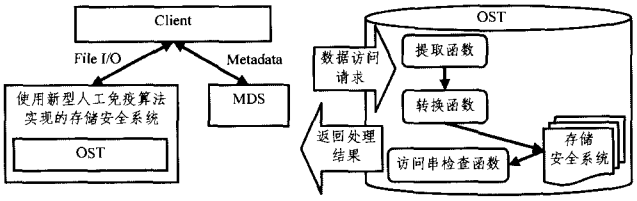

图 5 修改后的 Lustre 系统结构 图 6 基于免疫安全磁盘的结构图

基于免疫安全磁盘原型系统的结构如图 6 所示。首先修改 OST 模块的初始化代码，建立 PROC 通道，读取外部程序输入的自体集信息。在 OST 模块中增加安全系统头文件，实现安全系统的初始检测器生成、成熟检测器选择和访问串检查等函数，负责构建成熟检测器集和判断访问请求的合法性；设置初始匹配阈值为 12，最大值为 16，保存 32 个成熟检测器，由第 3 节中给出的计算公式可知安全系统最多能识别 5120 个非法访问串，能高效地保护系统的安全性。我们分析 OST 的通讯信息，实现访问请求提取和转换函数，负责提取访问请求中的操作命令、主机标识和数据对象标识等信息，转换成 16 位的二进制访问串。修改 OST 模块中接收访问请求函数 OST_Handle 的代码，在执行访问请求前调用提取函数和转换函数，提取访问信息，构建访问串；再调用访问串检查函数，判断访问请求的合法性，决定是否允许执行。

使用 Iozone 测试增加存储安全系统前后 Lustre 系统的 I/O 性能，检验存储安全系统的开销。采用大小为 4k，8k，16k，32k，64k，128k，256k，512k 和 1024k 的操作块，测试写 1M 大小文件时，Lustre 系统的 I/O 性能。测试结果如图 7 所示。我们可以发现，存储安全系统造成的 I/O 性能损失在总性能的 8%以内，能高效地保护磁盘数据的安全性。

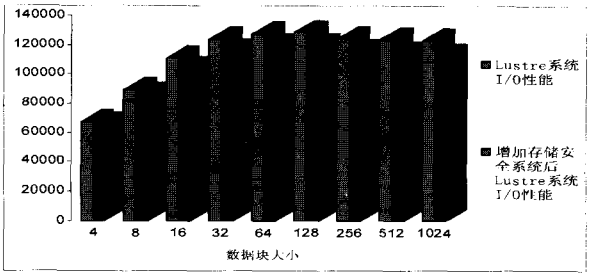

图 7 基于免疫安全磁盘的性能测试结果

结束语 本文提出新型人工免疫算法，用于研究高效的存储安全系统。给出了任意 r 连续位匹配规则，增强检测器识别非自体的能力，提高了存储安全系统的效率；为了解决选

择匹配阈值的困难，提出了自适应匹配阈值机制，能针对不同自体集，动态优化存储安全系统的检测效率和准确性，避免存储安全系统的失效。通过分析使用不同匹配规则时，检测器能识别的最大非法访问请求数量，证明任意 r 连续位匹配规则能很好地提高人工免疫算法的检测效率；分析了采用静态匹配阈值和自适应匹配阈值机制，针对不同自体集时，系统的检测效率和准确性，证明了自适应匹配阈值机制能动态优化安全系统的检测效率和准确性。通过使用新型人工免疫算法建立安全原型系统，测试使用不同匹配规则和不同匹配阈值机制时识别全部非法访问请求所需的成熟检测器数，结果表明新型人工免疫算法能较好地提高检测效率，优化安全系统的检测效率和准确性。最后我们修改开源存储区域网系统 Lustre 中智能磁盘部分的源代码，实现基于免疫安全磁盘的原型系统，测试增加安全存储安全系统前后 Lustre 系统的 I/O 性能，结果表明应用新型人工免疫算法能高效地保护存储系统的安全。

参 考 文 献

[1] Blaze M. A cryptographic file system for UNIX// Proceedings of 1st ACM Conference on Communications and Computing Security. 1993

[2] Howard J, Kazar M, Menees S, et al. Scale and performance in a distributed file system. ACM TOCS, 1988, 6 (1)

[3] Fu K, Kaashoek M, Mazieres D. Fast and secure distributed read-only file system. OSDI, October 2000

[4] Mazieres D, Kaminsky M, Kaashoek M, et al. Separating key management from file system security. SOSP, December 1999

[5] Li Xiangguo, Yang JiangHua, Wu Zhaohui. An NFSv4-Based Security Scheme for NAS. Parallel and Distributed Processing and Applications 2005. NanJiang, China

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

[7] 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

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

[9] Wylie J, Bigrigg M, Strunk J, et al. Survivable information storage systems. IEEE Computer, August 2000

[10] Ganger G R, Khosla P K, Bakkaloglu M, et al. Survivable Storage Systems // DARPA Information Survivability Conference and Exposition (Anaheim, CA, 12-14 June 2001). IEEE, 2001, 2: 184-195

[11] Kubiatowicz J, Bindel D, Chen Y, et al. OceanStore: An Architecture for Global-Scale Persistent Storage. ASPLOS, December 2000

[12] Freeman W, Miller E. Design for a decentralized security system for network-attached storage // Proceedings of the 17th IEEE Symposium on Mass Storage Systems and Technologies. College Park, MD, March 2000: 361-373

[13] Miller E L, Long D D E, Freeman W, et al. Strong security for distributed file systems// Proceedings of the 20th IEEE international Performance, Computing and Communications Conference (IPCCC '01). Phoenix, IEEE, April 2001: 34-40

[14] Miller E L, Long D D E, Freeman W E, et al. Strong Security for Network-Attached Storage // Proceedings of the 2002 Conference on File and Storage Technologies (FAST). January 2002: 1-13

[15] Kallahalla M, Riedel E, Swaminathan R, et al. PLUTUS: Scalable secure file sharing on untrusted storage// Conference on File and Storage Technology (FAST'03). (31 Mar-2 Apr San Francisco, CA, Published by USENIX, Berkeley, CA, 2003: 29-42

[16] 韩德志，傅湘林，黄建忠. 基于 iSCSI 的附网存储安全系统的研究与实现. 小型微型计算机系统，2004(7)

[17] Goh E-J. Shacham H, Modadugu N, et al. SiRiUS: Securing Remote Untrusted Storage// the proceedings of the Internet Society (ISOC) Network and Distributed Systems Security (NDSS) Symposium. 2003

[18] Azagury A, Cabetti R, Factor M, et al. A Two Layered Approach for Secuting an Object Store Network. SISW, 2002

[19] Company H-P. HP OpenView storage allocator. www. openview. hp. com, October 2001

[20] Brocade Communications Systems, Inc. Advancing Security in Storage Area Networks. White Paper, June 2001

[21] Hewlett-Packard Company. HP SureStore E Secure Manager X-P. www. hp. com/go/storage, March 2001

[22] Dasgupta D. An overview of artificial immune systems and their applications// D. Dasgupta, ed. Artificial immune systems and their applications, Springer-Verlag, Inc., 1999: 3-23

[23] de Castro L N, Timmis J. Artificial Immune Systems: A New Computational Approach. Springer-Verlag, London, UK, 2002

[24] Forrest S, Perelson A, Allen L, et al. Self-nonself discrimination in a computer// Proceedings IEEE Symposium on Research in Security and Privacy. Los Alamitos, CA, IEEE Computer Society Press, 1994: 202-212

[25] Balthrop J, Esponda F, Forrest S, et al. Coverage and generalization in an artificial immune system// W. B. Langdon, E. Cantú-Paz, K. Mathias, eds. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO). San Francisco, CA, 9-13 Morgan Kaufmann Publishers, July 2002: 3-10

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

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

[28] 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

[29] Helman P, Forrest S. An efficient algorithm for generating random antibody strings. Technical Report, CS-94-07. The University of New Mexico, Albuquerque, NM, 1994

[30] D'haeseleer P, Forrest S, Helman P. An immunological approach to change detection: algorithms, analysis and implications // J. McHugh and G. Dinolt, eds. Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy. USA. IEEE Press, 1996: 110-119

[31] D'haeseleer P. Further efficient algorithms for generating antibody strings. Technical Report CS95-3. The University of New Mexico, Albuquerque, NM, 1995