基于人工免疫机制的木马检测子系统*)

张 亮 陈雷霆

(电子科技大学计算机科学与工程学院 成都610054)

摘 要 本文分析了当今流行木马的技术和行为方面的特点,并给出了将木马检测模型从反病毒软件转向入侵发现系统(IDS)方面的建议。最后,提出了一个基于人工免疫机制的新的自适应实时木马检测模型,作为基于免疫机制IDS子系统。

关键词 网络安全,人工免疫,木马,入侵检测

Sub-System Based on Artificial Immune for Trojan Horse Detection

ZHANG Liang CHEN Lei-Ting

(School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054)

Abstract This paper analyzes the characteristics of technology and action of the popular Trojan horses, then gives a suggestion to transfer Trojan detection module from anti-virus software to intrusion detection system(IDS), finally a new self-adaptive real-time model for Trojan detection based on artificial immune is presented as the sub-system of the immunity-based IDS.

Keywords Network security, Artificial immune, Trojan horse, Intrusion detection

1 引言

随着计算机和网络技术的迅猛发展和广泛应用,信息安全问题日益得到重视。现有的计算机安全系统主要依赖于以下几个防御系统组成:第一道防线,将计算机局域网置于防火墙之后,防火墙检查网络数据包头部,防止非法的IP和端口的访问;第二道防线,将NIDS置于防火墙之后,通过检查网络数据包的数据段检测网络入侵行为;第三道防线,在局域网内的主机上分布着HIDS和反病毒软件,对内部网络数据包和病毒、蠕虫以及木马进行检测与清除。三者各有其优缺点,单独使用一种或两种无法全面地检测、防御攻击和入侵,只有将三者紧密结合,才能构建出安全可靠的计算机防御体系。

特洛伊木马,作为一种危害极大、难以检测的攻击工具,通常被反病毒软件作为病毒来处理。反病毒软件目标主要侧重于清除病毒,要求采用准确的模式匹配方法来确诊病毒的特征代码,因此决定其自适应能力的发展面临两难境地;加上木马具有特殊的隐蔽性和反检测性,对其采用病毒检测的方法效果往往不佳。近年来,基于人工免疫机制的IDS研究以其自适应性方面的优越性而成为计算机安全中的热点,因此可以考虑将木马检测系统并入到免疫型IDS中,以提高木马(尤其是未知木马)检测的智能水平。本文重点介绍当前基于人工免疫机制的IDS的现状,介绍特洛伊木马技术发展带来的挑战,并对特洛伊木马的行为特征进行分析,最后提出基于免疫机制的木马检测模型。

2 免疫型IDS

2.1 应用于IDS的生物免疫机制

生物免疫系统是一个高度复杂的分布协调自适应系统,它能自适应地识别和排除侵入机体的抗原性异物,并且具有学习、记忆和自适应调节能力。免疫型IDS就借鉴了如下的一些免疫机制[1~4]:

·自我与异己的识别:未成熟的免疫细胞在胸腺中进行自体耐受,通过阴性选择来筛选掉那些会识别自体抗原的免疫细胞,经历耐受测试的细胞才成为成熟的免疫细胞。

·抗体的多样性:抗体的基因编码都由稳定区和可变区组成,稳定区表达分子的特性,而可变区的基因则由抗原受体基因片段的随机组合连接而成,因此可识别千变万化的抗原。

·克隆选择和阴性选择:克隆选择是指免疫细胞是随机生成的多样性细胞克隆,每一个克隆的细胞表达同一抗原的受体特征,在识别抗原成功后,会导致免疫系统合成大量相同特异性的抗体。

·联想记忆:免疫系统能将与侵入抗原反应部分抗体作为记忆细胞保留下来,对于同类抗原的再次侵入时,相应的记忆细胞被激活而产生大量的抗体,缩短了免疫反应时间。

·反馈机制:当异己抗原被识别后,免疫细胞通过活化的方式刺激抗体的克隆增生,提高抗体的浓度;当异己抗原被排除后,又是通过抑制的方式使抗体凋谢或死亡,降低抗体的浓度,从而达到免疫系统的自我稳定。

·分布式自治:免疫系统中的免疫细胞和淋巴器官分布于全身,根据周围的环境自适应地确定自身的行为,整个免疫系统是一个没有控制中心的并行的分布自治系统,具有很强的鲁棒性。

2.2 免疫型IDS模型

按照检测数据的来源可以将IDS系统分为两类:一种是HIDS(基于主机的入侵检测系统),其检测数据的来源是主机上操作系统的审计日志;另一种是NIDS(基于网络的入侵检测系统),其检测数据的来源是网络上的数据流。目前的免疫型IDS主要应用在网络入侵方面,而NIDS也正朝着分布式、多级化的方向发展。Dasgupta等[5]提出过一个基于免疫的多Agent的NIDS模型,通过流动在网络上的多级Agent的相

*)本文得到国家863项目"战略预警与监管体系结构研究"(2002AA142040)支持。张 亮 硕士研究生,主要研究方向:网络安全。陈雷霆 博士,副教授,硕士生导师,主要研究方向:网络安全、网络多媒体。

互协同而实现入侵检测，但缺点是系统负载太大。Kim 等[6,7]提出了一个主从模式的免疫型 IDS 模型，该系统由两个部分组成：通往子网络的两个路由器之间的一台主机形成主 IDS，该子网中若干台主机构成从 IDS。在主 IDS 上存储着一个用于生成未成熟检测体的基因库，通过基因重组形成不成熟检测体，再通过基于小生境策略的阴性选择和克隆选择过程产生非自身抗原的成熟检测体，并传送到从 IDS。若某个从 IDS 上的成熟检测体成功检测到了入侵行为，它将成为记忆检测体，其信息将反馈给主 IDS 以实现基因库的进化。另外，还要生成其克隆并传播到其他的从 IDS 上，以使每个从 IDS 都具有针对该入侵的检测能力。主 IDS 和从 IDS 上都有通信器以便互相之间传送信息。

通常 HIDS 都是分布在网络中的每台主机上，如果和 NIDS 实现联动，其收集的主机信息可以反馈给 NIDS 并协助 NIDS 更准确地判断网络入侵行为，而 NIDS 收集的网络信息同样反馈给 HIDS 帮助 HIDS 更准确地分辨主机入侵行为，尤其是蠕虫和木马这些通过网络进行传播和通信的攻击工具。下面我们将通过对木马的分析，建立一个用于木马检测的 HIDS 模型作为分布式 IDS 的子系统。

3 特洛伊木马的分析

特洛伊木马（Trojan，简称木马），常被用作网络系统入侵的重要工具和手段，感染了木马病毒的计算机将面临数据丢失和机密泄露的危险。木马寄生在计算机中，利用用户的疏忽来窃取密码和提升权限，或利用一些操作系统自身的漏洞（如缓冲区溢出等）最终获取管理员权限。一旦获得管理员权限以后，修改审计日志，隐藏自己的踪迹，并植入后门。木马用作后门后，采用客户端/服务器端的方式与入侵者进行通信，成为入侵者扩大攻击范围和隐藏踪迹的跳板，为其他入侵活动提供可能，如攻击者植入分布式拒绝服务（DDoS）攻击的协作端程序，从而给整个网络安全造成巨大的威胁。由此看出，木马除了具有病毒的特征外，还表现出对相邻主机的网络入侵特征。

3.1 传统的木马检测技术

现在对特洛伊木马的检测和防御基本上分为两类[8]：

·通过建立木马特征字典，如反病毒软件检查被感染系统和文件是否包含木马特征，如 NIDS 通过包过滤检查网络数据包中是否包含木马特征。

·通过对文件或系统的完整性进行检查来防范木马，如利用单向的哈希函数（如 MD5、Xerox 等）来生成文件或系统的数字签名，如 HIDS 通过对操作系统的审计日志的分析来检测木马。

以上两类方法都属于被动检测的方式，对木马的入侵存在着反应迟钝和智能化不足的缺点，尤其是缺乏检测未知木马的自适应能力。随着木马反检测技术的发展，它们已经显得力不从心。

3.2 木马的反检测技术

为了加强特洛伊木马的生存能力，攻击者不断提高木马技术，出现了以下几种趋势[8,9]：

·多实例技术：所谓多实例就是将木马程序（多份）分别存放在目标机的不同目录下，或者同时运行多个木马进程，这些木马实例彼此相互监督和相互恢复，以提高木马的反清除能力。

·伪装技术：特洛伊木马的形式正向动态链接模块的形式发展，通过窗口 Hook、挂接 API 和远程线程将自己的代码嵌入到正在运行的进程中，有的甚至以核心模块或设备驱动程序的方式嵌入到操作系统内核层。

·反检测技术：通过检查反病毒软件的运行状态绕过检查或杀死反病毒软件进程，对防火墙则利用寄生端口和反弹端口技术逃避防火墙的检查。

·通信技术：木马感染系统后通过 FTP、EMAIL、ICQ 等方式实时通知木马控制端，并利用加密技术传输控制信息躲避包过滤系统的规则检查。

现有的基于病毒特征库的反病毒软件通常将木马作为病毒进行处理，但木马与病毒和蠕虫仍存在着较大差异。病毒和蠕虫的设计主要侧重于其隐蔽性和传播性的实现；而木马除此之外，还更多地强调与木马控制端通信，以及及清除与反检测能力。上述几项技术可使木马轻松摆脱反病毒软件的围剿，而其对未知木马更是束手无策。

那么依靠基于计算智能（包括人工免疫）技术的 IDS 系统能有效完成这一任务吗？让我们举个例子：如果一种新型的木马将反弹端口技术和加密通信技术相结合，就能有效躲避 NIDS 的检测。首先它利用 http 协议的80端口主动访问指定的 IP 地址（木马控制端所在），而通常出于使用需要防火墙不会禁止该端口，再将传送的信息进行加密，在这样的情况下自适应的 NIDS 系统也难以察觉木马的活动。最后木马在完成攻击行为后，将消除审计日志中自己的行踪，破坏 HIDS 的数据来源，从而越过 HIDS 的防线。

目前，对于基于计算智能尤其是人工免疫机制的 IDS 的研究，国内外已经开展了多项研究，并提出了不少模型，但是针对木马的检测还没有太多的探讨。因此研究针对木马的自适应的 HIDS 系统，是实现计算机安全的完整性的必要一环。

4 基于人工免疫机制的木马检测模型

4.1 基本原理

新墨西哥大学的 Forrest 在反病毒和主机入侵检测研究方面有着较大的影响。Forrest 等[10]运用免疫机制来检测程序和受保护数据的异常改动，其实验结果显示，这种方法能够很容易地发现病毒感染引起的数据文件的改变，并且能够检测未知的病毒。随后，Forrest 等[11]进一步利用免疫机制进行进程监视，其目的是为了检测对主机的入侵活动。她认为：根用户进程的系统调用比其他用户进程更具有潜在危险性，而且在正常情况下，每个根用户进程的系统调用在顺序上有着自己相对稳定的行为。该方法将“自我”定义为根用户进程的系统调用序列，并对一个典型的应用程序 sendmail 进行测试而取得比较好的结果。

但是 Forrest 的模型也存在一定不足：由于系统调用的类型种类繁多而且还在不断增加，定义系统调用类型成了一个和自适应要求背道而驰的问题，因此有必要将系统调用问题抽象出来。在 Forrest 的研究基础上，我们可以认为：由于进程的系统调用序列遵循着相对稳定的行为，而不同的系统调用通过对各种系统资源的占有、消耗、放弃、回收也会表现出相对稳定的资源使用行为特征，并且系统资源的类型也是有限的、非常稳定的。根据这一前提，我们提出：将进程的以时间为轴的资源使用状况曲线离散化，而得到的进程资源使用状况序列可以有效地表示为计算机系统的“自我”。此方法的特点是：系统资源信息的获取非常方便，开销极小，适用于实时检测；将多种系统资源定义为抗原的多个抗原决定基，通过对多种资源状况的综合分析可以提高检测的准确性，但缺点是带来更大的计算量，这一问题下面将通过数据结构和算法的优化来弥补。

现在的软件体积日益庞大，功能调用分支极其复杂，不可

能存在唯一的系统调用特征序列，因此需要考虑定义表达同一抗原的多个抗原特征表达序列。同时在软件的使用过程中存在着两个现象：2/8规则，即80%的用户只使用了软件20%的功能；新程序的安装和旧程序的升级和卸载。根据这些现象，我们认为在我们的模型中仍存在着相对稳定的自体抗原，但自体抗原有着比较明显的新生、演变和凋谢的基因进化过程，这是我们的木马检测模型与其他人工免疫系统不同的地方。

4.2 抗原和抗体的定义

我们的模型每天跟踪和记录主机上进程的各种系统资源的占用情况(如CPU占用时间、内存占用空间、外部存储器占用时间、IO占用时间、网络占用时间和带宽等)，并以T时间的间隔对其进行采样，建立以一天的时间为长度的资源使用状况曲线的离散序列，最后将其保存为自体抗原数据段。同时我们将木马进程和木马寄生进程的资源使用状况曲线的离散序列定义为非我抗原，将非己抗原的检测体定义为抗体。由于进程大量存在着对某种功能的反复调用，自体抗原可以通过自我"提纯"减少冗余数据量。为了进一步减少数据量，我们将采用增量值的方法记录下相邻时间之间的变化值。关于被检测进程的类型，允许灵活配置，如服务器上可只检测根用户(管理员)进程，PC机上则通常应检测所有进程。

抗原的结构由不变区和可变区组成，不变区包括进程的信息(如进程名、路径、文件长度等)、资源类型等，可变区则是二进制化的系统资源使用状况的离散序列，长度不定，如图1所示。在我们的模型中每天得到的数据记录作为一个抗原决定基，假如有M种资源类型，每个自体抗原会由D乘M个抗原决定基。检测体(抗体)的结构类似于抗原。在经过D天的耐受期后，我们将得到的所有自体抗原集合成系统的自体抗原基因库。抗体的结构类似于抗原。

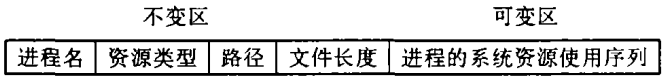

图1 抗原/抗体结构图

4.3 免疫应答模型

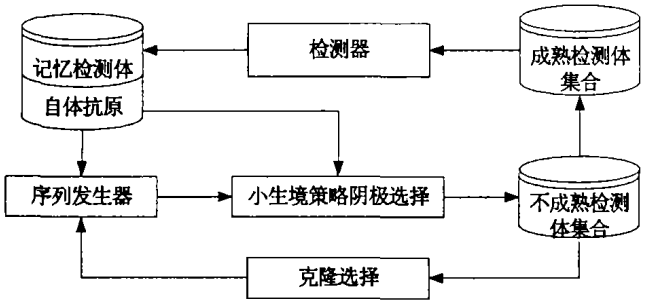

图2 免疫应答模型

如图2所示，检测体(抗体)的产生，是在阴性选择器中通过小生境策略算法生成抗体序列数据，然后和不变区构成不成熟的检测体。根据阴性选择原理，检测体在经过自体耐受(和基因库里面的自体抗原比较，和自体抗原类似的检测体被抛弃)后，被释放到检测空间中执行检测任务。在检测到非己抗原后，该抗体执行克隆选择，克隆出的抗体通过几轮的阴性选择和克隆选择后得到成熟的检测体集合。

对检测算法我们可以采用所属IDS的算法，比如连续r位匹配算法。该算法描述为：当连续匹配的位数大于等于r值时，两个序列匹配，否则不匹配。这些匹配都是部分的匹配，规则中的值，类似于生物免疫系统中免疫细胞的激活阈值。在产生算法设计中，利用连续r位的匹配规则，可实现以较小的检测器集合，检测到较大范围的"非我"行为。同时，我们对多种资源的检测结果进行综合以得出最全面的评价结果，减小误报的概率。我们定义N为资源类型数量，P_i为某种资源的检测匹配度，A_i为某种资源的加权值，从而得到以下的匹配度的评价公式：

$$P_r=\sum_{i=1}^{N}A_iP_i \tag{1}$$

在该匹配度P_r超过某一阈值Q_r后，检测模块向管理模块发出警告，由管理模块应用预定义的反应策略进行处理。如果检测模块没有及时收到回复，系统则通过通信模块和其它主机上的NIDS和木马检测HIDS进行通信，由它们加强对来自本系统的信息进行严密监视。在等待一段时间Tn以后，汇集来自其他HIDS和NIDS的反馈信息，对匹配度的评价公式修改如下：

$$P=P_r+\sum_{j=1}^{M}B_jP_j \tag{2}$$

其中M为其余HIDS和NIDS的数量，B_j为采信度，P_j为其它主机上的NIDS和木马检测HIDS得出的匹配度评价值。在该匹配度P超过某一阈值Q后，系统再次向管理模块发出警告，并写入安全日志。在得到管理模块的审查确认以后，该检测体将被录入记忆检测体集合中。

4.4 基因库的进化

在我们的木马检测模型中，抗体有着类似其他人工免疫系统的进化机制，自体抗原也经历着相对缓慢的进化过程(如定义进化周期为一天)，如图3所示。

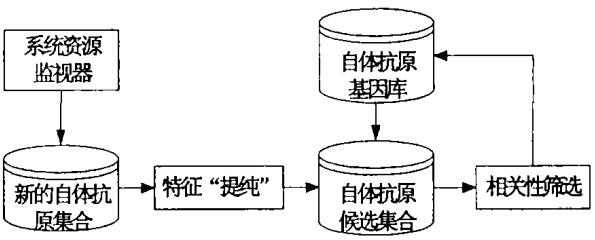

图3 自体抗原进化模型

首先在系统资源监视器中产生新的自体抗原，通过特征"提纯"，清除掉数据序列中冗余特征序列段。"提纯"算法可以将数据序列分成几个子序列，将相同的数据片段进行合并，清除掉冗余的数据。算法如何保留最多的特征和最少的数据有待进一步的研究。然后"提纯"后的自体抗原和基因库中的自体抗原进行比较，如果类似则被抛弃，反之则进一步筛选。我们利用以下的相关系数公式来计算自体抗原之间的相关性：

$$\rho^2=\frac{(\sum_{i=1}^{N}(X_i-\overline{X})(Y_i-\overline{Y}))^2}{\sum_{i=1}^{N}(X_i-\overline{X})^2\sum_{i=1}^{N}(Y_i-\overline{Y})^2} \tag{3}$$

相关系数产生一个(0,1)的数字，该数字关系到两个输入序列的相似性，其定义为：$X,Y\in\{0,\cdots,255\}^N$，$N=L/8$，L为序列长度。筛选步骤如下：首先合并基因库中的自体抗原和今天的自体抗原候选集合，然后计算所有自体抗原的数据段之间的相关性，可以得到一对相关性最大的自体抗原，继续选择两者各自相关性集合的最大值进行比较，依次类推，最终筛选出被淘汰的自体抗原。

4.5 特殊的安全策略

拥有根用户(系统管理员)权限的木马对 HIDS 及木马检测系统本身也有着巨大的威胁,因此我们的模型需要特殊的安全策略:

·自我保护机制:数字签名技术对保护数据完整性起着重要的作用,我们的模型将引入该技术对本系统的程序和关键数据进行保护。同样,多实例的技术也可以用于保护本系统,我们采用启动多个监视进程只激活其中一个的策略,一旦其中一个进程被木马删除,其他的进程可以激活和恢复被删除的进程。

·分布式数据存储:安全日志和检测数据对于本系统的检测和分析极为重要,不仅需要检测其是否受到破坏,还要进行恢复。因此我们采用分布式的数据存储方法,将数据的副本传送给其他的 HIDS,保存在其他的主机上。

4.6 模型的总体设计

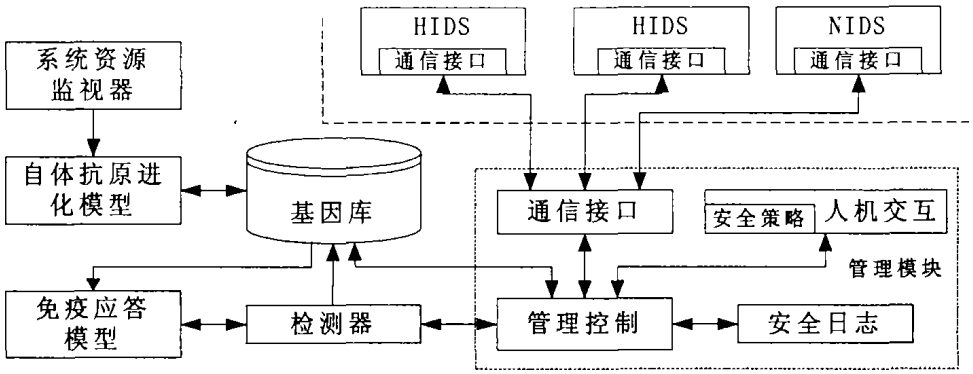

图4 基于人工免疫机制的木马检测系统模型

本系统作为免疫型 IDS 的子系统,必须和 IDS 中的其他模块紧密结合,相互通信。因此本系统设计了一个管理模块,分别由管理控制、通信接口、安全日志和人机交互四个子模块组成。在管理模块的控制下,检测器执行检测并经过管理控制模块审查提交记忆检测体;管理控制模块根据需要访问基因库,和其他 HIDS 交换基因信息(如自体抗原和记忆检测体等)。管理控制模块把对检测数据的分析结果和入侵检测的警告信息保存到安全日志中,同时和其他 HIDS 和 NIDS 交换日志以实现分布式数据存储;而所有的外部通信都由通信接口模块负责。人机交互模块是和管理员进行交互的接口,它包含一个安全策略子模块,其中定义了一些默认的安全策略,在管理员不在的时候指导管理控制处理相应的事务。

结束语 本论文首先指出了当前反病毒软件在检测未知木马方面的不足,介绍了人工免疫系统在自适应性方面的优点,并指出了人工免疫机制在木马检测方面的可行性;然后通过对一些木马新技术的分析,举出一个木马模型证明了现在的计算机安全体系的不足,并提出将木马检测从反病毒软件中迁移到免疫型 IDS 中作为子系统,利用其免疫机制来提高木马检测的自适应能力;最后在 Forrest 研究的基础上,提出了依据进程的系统资源使用状况来映射进程的系统调用的行为模式,并以此建立了基于人工免疫机制的木马检测模型。本系统是基于主机的检测模型,而且提出了自体抗原的进化问题,无疑会增加主机的负担,因此如何提高自体抗原的进化算法和其它免疫算法的效率,如何确定模型中定义的耐受时间、采样间隔、激活阈值等是下一步重点研究的方向。而如何和 IDS 实现良好的联动,如何防御和清除木马、恢复被感染的系统也有待我们进一步探索和研究。

参 考 文 献

1 赵俊忠,黄厚宽,田盛丰. 免疫机制在计算机网络入侵检测中的应用研究[J]. 计算机研究与发展,2003(9):1293～1299
2 焦李成,杜海峰. 人工免疫系统进展与展望[J]. 电子学报, 2003(10):1540～1548
3 左兴权,李士勇,李远贵. 人工免疫系统研究的新进展[J]. 计算机测量与控制, 2002(10):701～705
4 莫宏伟. 人工免疫系统原理与应用[M]. 哈尔滨:哈尔滨工业大学出版社, 2002
5 Dasgupta D. Immunity-based intrusion detection system: a general framework [C]. In: Proc. of the 22nd National Information Systems Security Conf. Crystal City: NIST Publishers, 1999
6 Kim J, Bentley P. The artificial immune model for network intrusion detection [C]. In: 7th European Conf. on Intelligent Techniques and Soft Computing. Aachen, Germany, 1999. 13～19
7 Kim J, Bentley P. Negative selection and niching by an artificial immune system for network intrusion detection [C]. GECCO'99, Orlando, Florida, 1999
8 王锐,等译. 网络最高安全技术指南[M]. 北京:机械工业出版社, 1998
9 朱明,徐骞,刘春明. 木马病毒分析及其检测方法研究[J]. 计算机工程与应用,2003,28:176
10 Stephanie F, Lawrence A, Alan P, Rajesh C. Self-nonself discrimination in a computer [C]. In: Proc. of the IEEE Computer Society Symposium on Research in Security and Privacy, 1994. 202～212
11 Stephanie F, Steven H, Anil S, Thomas L. A sense of self for unix processes [C]. In: Proc. of the IEEE Computer Society Symposium on Research in Security and Privacy, 1996. 120～128

(上接第52页)

储广域网的系统性能提供了理论依据。理论分析和测试结果表明,虽然影响 IP 存储广域网性能的因素有很多,当网络带宽小于存储设备性能时,IP 广域网网络带宽成为决定 IP 存储广域网最大系统吞吐率的决定因素。

参 考 文 献

1 Chang F, et al. Myriad: cost-effective disaster tolerance. Conference on File and Storage Technologies. Monterey, CA, Jan. 2002
2 Cancio G, et al. Tierney, The DataGrid Architecture: [Data Grid TechReport DataGrid-ATF-01]. July 2001
3 Yoshida H. SWAN: Storage Wide Area Network: An Overview of Today's Capabilities and Futrue Directions, Hitachi Data Systems Corp. technical paper, Aug. 2001
4 SNIA Technical Council. Shared Storage Model: A framework for describing storage architectures. draft SNIA TC Proposal document . June , 2001
5 Markatos E P. Speeding up TCP/IP: faster processors are not enough. In: the 21st IEEE Intl. Performance, Computing, and Communication Conf. (IPCCC'02), Phoenix, Arizona, April, 2002
6 Saroiu S, Gummadi P, Gribble S. A Measurement Study of Peer-to-Peer File Sharing Systems. In: Multimedia Computing and Networking Conf. San Jose, CA, Jan. 2002
7 Ng W T, et al. Obtaining High Performance for Storage Outsourcing, FAST 2002, Jan. 2002
8 Hwang K, Jin H, Ho R S C. Orthogonal Striping and Mirroring in Distributed RAID for I/O-Centric Cluster Computing. IEEE Transactions on parallel and distributed systems, 2002, 13(1)