基于时间序列图挖掘的网络流量异常检测

周颖杰 胡光岷 贺伟淞
(电子科技大学宽带光纤传输与通信网技术重点实验室 成都 610054)

摘 要 网络流量异常检测要解决的核心问题之一是获得信息的全面性和流量信息描述的准确性。针对现有网络异常流量检测方法分析多时间序列的不足,提出了一种基于图挖掘的流量异常检测方法。该方法使用时间序列图准确、全面地描述用于流量异常检测的多时间序列的相互关系;通过对项集模式进行支持度计数,挖掘各种频繁项集模式,有利于对各种异常流量的有效检测;通过挖掘各项集之间的关系,引入了项集的权重系数,解决了流量异常检测的多时间序列相互关系的量化问题。仿真结果表明,该方法能有效地检测出网络流量异常,并且对 DDos 攻击的检测效果明显优于基于连续小波变换的检测方法。

关键词 网络流量异常检测,多时间序列,图挖掘
中图法分类号 TP393.08 文献标识码 A

Network Traffic Anomaly Detection Based on Data Mining in Time-series Graph

ZHOU Ying-jie HU Guang-min HE Wei-song
(Key Laboratory of Broadband Optical Fiber Transmission and Communication Networks UESTC of China Ministry of Education, Chengdu 610054, China)

Abstract Comprehensive collection and accurate description of traffic information are core problems in network traffic anomaly detection. Aiming at the lack of traffic anomaly detection in analyzing multi-time series, we proposed a network traffic anomaly detection method based on graph mining. Our method accurately and completely described the relationship among multi-time series which are used in traffic anomaly detection by time-series graph. By mean of the support count of the patterns, our method mined all the frequent patterns, which is conducive to detecting many kinds of abnormal traffic effectively, through mining the relationship among all pattern sets, our method introduced weight coefficients of the pattern sets, which is able to solve relationship quantification issues of multi-time series in traffic anomaly detection. The simulation results show that the proposed method can effectively detect the network traffic anomaly and achieves a higher accuracy than the based CWT (Continuous Wavelet Transform) method in term of DDos attacks detection.

Keywords Network traffic anomaly detection, Multi-time series, Graph mining

1 引言

网络流量异常是指网络的流量行为偏离其正常行为的情形。网络异常流量具有发作突然、先兆特征未知的特点,有可能在短时间内给网络和网络设备带来极大的伤害。人们通常会通过对网络流量行为的描述、分析来发现网络或系统中可能出现的异常行为,并向管理员提出警告,这就是网络流量异常检测。网络流量异常检测要解决的核心问题之一是获得流量信息的全面性和流量信息描述的准确性。由于异常流量对网络的危害极大,网络中的流量异常随着网络应用的普及越来越多,快速、准确地检测网络异常流量并做出合理的响应,已成为目前国内外学术界和工业界广泛关注的前沿科学问题之一。

现有流量异常检测方法通常将流量随时间变化的信息看作一个随时间变化的一维信号(或一维时间序列),通过多种信号分析方法进行流量异常检测。Hussain 等人提出了通过信号的频谱分析不同种类的 DDoS 攻击的方法[1]。Cheng Chen-Mou 等人提出了通过流量信号的能量谱密度分析 TCP 流量的周期特征来识别 DDoS 攻击的方法[2]。V. Alarcon-Aquino 等人提出了一种基于 UDWT(undecimated discrete wavelet transform)和贝叶斯分析的算法,利用各级的小波系数检测和定位给定时间序列在方差和频率上的微弱改变[3]。P. Barford 等人利用小波过滤器发现细微异常流,通过检测本地过滤数据的急剧变化进行异常检测[4]。Gao Jun 等人提出了一种新的基于小波包分析的网络流量异常检测新机制,对高、中、低频异常流量具有同样的检测能力[5]。

由于异常流量检测的复杂性,一维时间序列的分析方法往往具有较高的误检率和漏检率。W. Lee 和 Xiang 提出将

到稿日期:2008-05-20 本文受国家自然科学基金(60572092),教育部"新世纪优秀人才支持计划"(NCET-07-0148)资助。
周颖杰(1984—),男,博士生,主要研究方向为网络异常检测与识别,E-mail:yjzhou@uestc.edu.cn;胡光岷(1966—),男,博士,教授,博士生导师,主要研究方向为网络行为学与网络安全;贺伟淞(1974—),男,博士生,主要研究方向为网络流量分析、时间序列数据挖掘、信号处理。

信息熵用于异常检测，结合信息增益、信息代价等来进行检测[6]。A. Lakhina 等提出使用报文特征分布(IP 地址和端口)的时间序列描述网络异常，并使用 entropy 作为工具进行分析基于特征的方法检测和识别大量异常[7]。管晓宏等人使用独立成分分析将网络流量划分为正常空间和异常空间两个独立成分，通过对异常空间的分析检测流量异常[8]。

多时间序列分析提高了流量异常检测的精度，可以有效降低误检率和漏检率。虽然许多方法认为多个时间序列之间是相互关联的，在检测过程中也大量利用时间序列之间的关联性，但现有方法对多个时间序列的描述是独立的，缺乏一种有效的手段对多时间序列的相互关系进行较为准确、全面的描述，并应用于流量异常的检测。因此流量异常检测所依据的信息在完整性、准确性方面存在一定的缺陷，降低了检测的精度。针对上述问题，本文提出可以将某一时刻的多个时间序列值及其相互关系使用一个图进行描述，每一时刻获得一个图，多个时刻的多个图构成一个时间序列图。

时间序列图可以更准确、全面地描述用于流量异常检测的多时间序列的相互关系，是流量异常检测的一种新思路。使用现有的多时间序列分析方法难以充分利用其丰富的信息。为此本文提出一种基于图挖掘的流量异常检测方法，该方法通过对项集模式进行支持度计数，挖掘各种频繁项集模式，有利于对各种异常流量的有效检测；通过挖掘各项集之间的关系，引入了项集的权重系数，解决了流量异常检测的多时间序列之间相互关系的量化问题。仿真结果表明，该方法能有效地检测出网络流量异常，并且对 DDoS 攻击的检测效果明显优于基于连续小波变换的检测方法。

2 时间序列图的构成

本文用于异常检测的数据来自 netflow 的流量信息，其中包括源/目的 IP、源/目的端口、数据包数量等属性。由于每个 netflow 数据包中含有几万行甚至几十万行的流量信息，直接对这种流量信息进行处理很困难。将 netflow 数据各个属性看作是一组随机事件，可以利用信息熵概念来有效衡量各个属性对应数据的集中和分散情况，得到海量数据的粗粒度表示。我们选取源/目的 IP、源/目的端口 4 个信息熵序列，通过建立时间序列图来表示它们及其之间关系。

2.1 NetFlow 流量数据与信息熵

NetFlow 技术可以对流经网络设备的 IP 数据流进行特征分析和测量，它是当今互联网领域公认的最主要的 IP/MPLS 流量分析和计量行业标准，被广泛应用于网络安全分析和监控中。Netflow 报文由报文头和多个流信息记录两部分构成。流信息中包含了源/目的 IP、源/目的端口、数据包数量等属性。

信息熵是对随机事件不确定性的度量。将 NetFlow 流量数据当作离散信息源，其中的各个属性作为一组随机事件，就可以用信息熵来进行分析[9]。

流量数据属性 S 的信息熵为

$$H(S)=-\sum_{i=1}^{n}P_i\ln P_i \qquad (1)$$

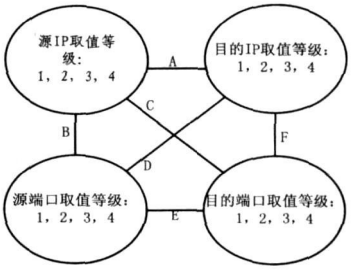
其中，P_i 为属性的某个值出现的频率，n 为属性的总实例数，$\sum_{i=1}^{N}P_i=1$。信息熵能有效地表现出同一属性上对应数据的集中和分散情况。尤其是在大规模网络流量中，数据越集中的地方熵值越小，数据越分散的地方熵值越大。

本文只选取源/目的 IP、源/目的端口 4 个属性，它们能比较有效地表征大规模网络流量的异常。以分布式拒绝服务(Distributed Denial of Service，简称 DDoS)攻击为例，它在攻击时操纵足够数量的妥协主机在几乎同一时间向被攻击者发送大量的无用分组，这将引起源 IP 信息熵突然增大、目的 IP 信息熵突然减小。

2.2 多信息熵序列的图表示

下面我们利用源/目的 IP、源/目的端口 4 个熵时间序列，生成一个时间序列图。

如图 1 所示，用 4 个点分别表示该时间采样点上的源/目的 IP、源/目的端口这 4 个信息熵，用边表示它们之间的关系。点的不同取值决定该点信息熵的取值等级，多点间不同取值等级的组合构成不同的模式。边的权值反映该边两个端点信息熵值在该时间采样点上的变化的相似度。边权值在整个时间序列上的变化幅度反映了流量异常检测的多时间序列之间相互关系的程度：边权值的变化幅度越小，关系越紧密；边权值的变化幅度越大，关系越松散。



A, B, C, D, E, F 为各个边的边权值

图 1 多信息熵序列的图表示

具体做法如下：

1)点的表示

首先将信息熵的取值进行归一化，将其取值映射到[0, 1]区间，然后根据映射后的大小分为 4 个等级：值在[0, 0.25]区间的为 1 等级，在[0.25, 0.5]区间的为 2 等级，在[0.5, 0.75]区间的为 3 等级，在[0.75, 1]区间的为 4 等级。用 A, B, C, D 表示源/目的 IP、源/目的端口这 4 个点，用 A_j, B_j, C_j, D_j(j 表示该信息熵的取值所在等级，j= 1, 2, 3, 4)分别表示该时间采样点上的源/目的 IP、源/目的端口的信息熵模式。pq 两点(p, q= A, B, C, D 且 $p\neq q$)为一个 2-项集，两点间的模式 p_iq_j(p, q= A, B, C, D 且 $p\neq q$, i, j= 1, 2, 3, 4)为一个 2-项集模式。pqr 三点(p, q= A, B, C, D 且 $p\neq q$, $p\neq r$, $q\neq r$, i, j, k= 1, 2, 3, 4)为一个 3-项集，三点间的模式 $p_iq_jr_k$(p, q= A, B, C, D 且 $p\neq q$, $p\neq r$, $q\neq r$, i, j, k= 1, 2, 3, 4)为一个 3-项集模式。

网络流量的异常在图中通常表现为有较奇异的项集模式。

2)边的表示

记 k 时刻源/目的 IP、源/目的端口的信息熵分别为 $H^k(A)$, $H^k(B)$, $H^k(C)$, $H^k(D)$

定义连接两点的边的权值

$$Wb^k(p,q)=\frac{H^k(p)-H^{k-1}(p)}{H^k(q)-H^{k-1}(q)} \qquad (2)$$

其中，p, q= A, B, C, D 且 $p\neq q$。

边的权值反映该边两个端点信息熵值在该时间采样点上

的变化的相似程度。如果该边两端点信息熵值整个时间序列上的相似程度或变化幅度都一致，那么这两个信息熵时间序列相关性就很强，它们之间的关系就很紧密。边的两个端点属于两个不同的时间序列，边权值的变化幅度反映了这两个时间序列之间关系的紧密程度：边权值的变化幅度越小，关系越紧密；边权值的变化幅度越大，关系越松散。

这样，在每个时间采样点上都形成了一幅无向赋权图，在整个时间序列上就得到了一个时间序列图。

3 时间序列图的挖掘与流量异常检测

基于图挖掘的异常检测是异常检测中一个新兴的领域。Caleb C. Noble 和 Diane J. Cook 提出了两种在基于图的数据中发现异常模式的方法[10]：异常子结构检测和异常子图检测，但它们都没有应用图元之间的关系来分析。应用图元之间的关系进行分析由于引入了更加丰富的信息，有助于提高网络异常检测的准确度。所以本文首先通过图挖掘分析图元及其之间的关系，得到各项集模式的支持度[11]和各项集的权重系数，然后对它们进行分析，得到异常判定准则。

3.1 时间序列图的挖掘

时间序列图的挖掘包括：项集模式的支持度计数，得到各种频繁项集模式，为判断网络流量异常提供依据；各项集之间关系的挖掘，引入项集的权重系数，以解决流量异常检测的多时间序列之间相互关系的量化问题。

3.1.1 项集模式的支持度计数

支持度反映了一个项集模式出现的频繁程度。支持度越小，说明该模式出现得越不频繁，则网络流量越可能发生异常。对于某时间采样点上的一个图，它的 6 个 2-项集模式和 4 个 3-项集模式的频繁程度可以反映该时间采样点上出现异常流量的可能性。该点上有越多的模式是频繁的模式，则该时间采样点上出现异常流量的可能性越小；反之，则该时间采样点上出现异常流量的可能性越大。

项集模式的支持度为项集模式在项中出现的次数与项集包含的总实例数之比。使用支持度计数的方法挖掘图中的 2-项集模式和 3-项集模式，得到所有 2-项集模式的支持度 $\mathrm{Sup}_t(p, q)$ 和 3-项集模式的支持度 $\mathrm{Sup}_t(p, q, r)$（p, q, r 分别为图中的不同节点，t 表示图所在的时间采样点）。应该认识到，频繁 2-项模式比频繁 3-项模式对检测异常更具有价值，因为其反映的多时间序列之间相互关系是最直接、最基本的。

3.1.2 边权值的挖掘

为了解决流量异常检测的多时间序列之间相互关系的量化问题，对边权值进行挖掘，引入项集（本文中为 2-项集或 3-项集）的权重系数来进行描述。

项集的权重系数代表了项集中元素的联系程度。它不仅确定了该项集的关系强度，也确定了该项集所处模式的关系强度。项集中元素的联系程度越强，对该项集支持度的影响就越大，项集的权重系数越大；反之，对该项集支持度的影响就越小，项集的权重系数越小。

对边权值进行挖掘可以得到项集的权重系数。每条边权值 $Wb^k(p, q)$（$p, q = A, B, C, D$ 且 $p \neq q$）在整个时间序列上的样本二阶中心矩 $S(p, q) = \frac{1}{n}\sum_k (Wb^k(p, q) - E(Wb^k(p, q)))^2$。边权值在整个时间序列上的变化幅度反映了流量异常检测的多时间序列之间相互关系的程度。某条边的样本二阶中心矩越小，该边两个端点信息熵值之间联系越稳定，说明其联系越强，对该节点对的支持度影响就越大。

随着样本二阶中心矩的增大，其继续增大对项集的权重系数的影响会减小，因此可以近似地用单增的指数函数从趋势上来描述样本二阶中心矩和项集的权重系数之间的关系。定义项集的权重系数如下：

2-项集的权重系数 $W^2_{p,q} = 10^{-S(p, q)}$（$p, q$ 分别为图中的不同节点）。式中使用指数函数使项集的权重系数取值在 0 到 1 之间。当某条边的样本二阶中心矩为 0 时，该边两个端点信息熵值正相关，项集的权重系数为 1；当某条边的样本二阶中心矩为 $+\infty$ 时，该边两个端点信息熵值不相关，项集的权重系数为 0。

3 个节点之间的关系可以用它们两两关系之和来表示。3-项集的权重系数 $W^3_{p,q,r} = 10^{-(S(p, q)+S(p, r)+S(q, r))}$（$p, q, r$ 分别为图中的不同节点）。

3.2 异常判定准则

某时间采样点上项集模式的频繁程度或支持度可以反映该时间采样点上出现异常流量的可能性。该时间采样点上越多的模式是频繁的模式或各模式的支持度越大，出现异常流量的可能性越小；反之，出现异常流量的可能性越大。支持度的大小是检测网络异常流量的一个有效手段。

项集的权重系数解决了流量异常检测的多时间序列之间相互关系的量化问题，它量化了各项集所处模式的支持度对该时间采样点是否存在异常的贡献程度。

基于以上思想以及对时间序列图的挖掘结果，定义异常系数 W_t 来衡量某个时间采样点上的单点路由器的网络流量异常程度。

$$W_t = -\min_{0<l<N}\{W_l\} - \log_{10}\Big(\sum_{1\leqslant p, q\leqslant 4,\ p\neq q} W^2_{p,q} \circ Sup_t(p, q) + 0.6\times \sum_{1\leqslant p, q, r\leqslant 4,\ p\neq q且p\neq r且q\neq r} W^3_{p,q,r} \circ Sup_t(p, q, r)\Big) \quad (3)$$

其中，p, q, r 分别为图中的不同节点，t 表示图所在的时间采样点，N 为总的时间采样点数。

异常系数 W_t 越大，说明该时间采样点上越可能出现异常。

4 仿真试验及分析

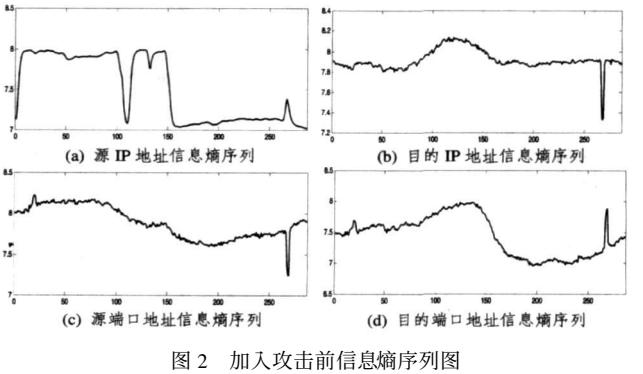


图 2 加入攻击前信息熵序列图

本文在仿真中使用来自 Abilene[12] 骨干网的采样数据。Abilene 是 Internet2 骨干网，连接着美国 200 所大学。Abilene 包含 12 个节点，横跨美国大陆。我们搜集了 2006 年 12 月 13 日一天的 Abilene 的 IP 级采样（以 100: 1，每隔 5min

进行周期采样)流量数据,将每 5min 采集到的数据作为一个时间采样点,一天 288 个时间采样点。

对上述 Abilene 骨干网络数据进行转换、计算后,可得到 4 个信息熵序列:源 IP 地址信息熵序列、目的 IP 地址信息熵序列、源端口地址信息熵序列、目的端口地址信息熵序列,如图 2 所示。

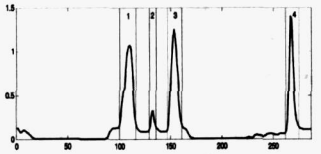
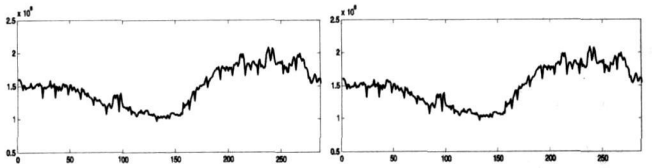


图 3　加入攻击前异常系数时序图

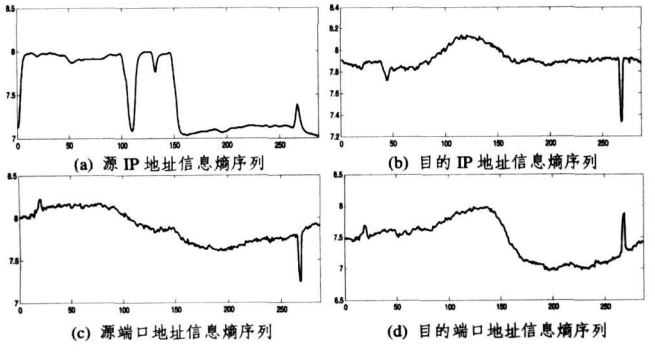
图 3 绘制了该天中通过单节点流量的异常系数时序图。图中共有 4 个尖峰,对应检测出的 4 个异常。通过手工分析原始 IP 包数据可知,4 个时间采样点均为异常,其中异常 1,2,3 为点到多点异常流量,异常 4 为 DDos 攻击。



(a) 加入攻击前　　(b) 加入攻击后

图 4　加入攻击前后的包数据流量图

为了进一步检验该方法的有效性,我们在 IP 级采样的流量数据中注入攻击流,便于准确地知道"攻击"发生的时间。仿真模拟了 200 个 Agent 攻击一台主机的过程,在骨干路由器节点搜集数据包,将每 5min 采集到的攻击数据作为一个时间采样点,总共采集 10 个时间采样点的 Netflow 流量数据,这就构成了一个攻击包序列。我们将攻击流注入到第 40~49 个时间采样点中。这样就模拟了一个在时间采样点 40~49 之间,由 200 个 Agent 攻击一台主机的 DDoS 攻击。图 4 为加入攻击前后的包数据流量图。图 5 为加入攻击流量后的 4 个信息熵序列图。图 6 中绘制了加入攻击流量后通过单节点流量的异常系数时序图。



(a) 源 IP 地址信息熵序列　　(b) 目的 IP 地址信息熵序列

(c) 源端口地址信息熵序列　　(d) 目的端口地址信息熵序列

图 5　加入攻击后信息熵序列图

直接从加入攻击后的流量信号来看,虽然攻击数据的加入引起了流量在时间采样点 40~49 之间有一定程度的加强,但是与其他时刻的流量比较起来,其强度并不明显。从加入攻击后的 4 个信息熵序列来看,仅目的 IP 地址信息熵序列在时间采样点 40~49 之间的值有一个小幅度的减小。而通过本文方法分析后,根据最终得到的如图 6 所示的异常系数时序图,可以很容易地判断出异常并确定其发生时间(图中标号 a 所示时间采样点位置)。从图 6 我们发现时间采样点 40~49 之间有一个异常,这就是人为注入的异常。

为了将本文方法与 Alberto Dainotti 的基于小波的方法做一个比较,此处用基于小波的方法对同样的数据(原始流量和注入攻击均相同)进行了实验仿真,根据 Netflow 中流量数据的 IP 包数据信息得到的结果,如图 7 所示。该仿真的实验程序、参数设置等完全参照文献[13] 中介绍的基于小波的流量异常检测法。为了尽可能多地检测出异常,与本文的方法相比较,我们将检测模块中的所有输出全部报警,对包数据流量直接进行精检测。从仿真结果图可知,人为注入的异常(40~49 之间)虽然引起了包数据流量一定程度的变化,但是变化并不显著;而且,从整个时间范围来看,根本无法判断该处存在异常。

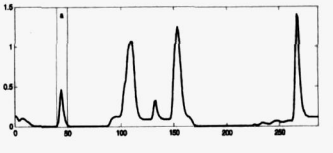
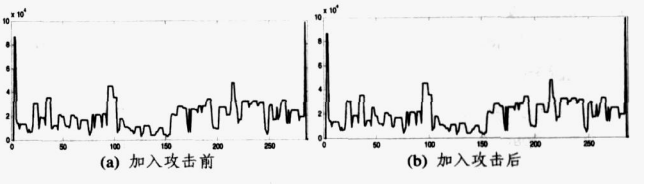


图 6　加入攻击后异常系数时序图



(a) 加入攻击前　　(b) 加入攻击后

图 7　基于小波方法的检测结果

从图 6 和图 7 的仿真结果可知,本文方法对 DDoS 攻击的检测效果明显优于基于连续小波变换的检测方法。

结束语　本文提出一种基于时间序列图挖掘的流量异常检测方法,它利用时间序列图准确、全面地描述了用于流量异常检测的多时间序列之间的相互关系;通过挖掘频繁项集模式和引入项集的权重系数,充分利用了流量异常检测的多时间序列之间相互关系的丰富信息,解决了流量异常检测的多时间序列之间相互关系的量化问题,提高了检测的精度。通过对采集 1 天数据的仿真实验结果表明,该方法能有效地检测出网络流量异常,并且对 DDoS 攻击的检测效果明显优于基于连续小波变换的检测方法。下一步的工作中,我们将结合网络流量特征参数和网络状态特征参数及它们之间的相互关系,研究新的、更有效的时间序列图的构建方式与相关的图挖掘方法,获得全面、准确的网络异常行为特征,并在此基础上进一步对异常进行预测,对网络安全态势进行评估。

参 考 文 献

[1] Hussain A, Heidemann J, Papadopoulos C. A Framework for Classifying Denial of Service Attacks //Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. Karlsruhe, Germany, 2003

[2] Cheng C M, Kung H T, Tan K S. Use of Spectral Analysis in Defense Against Dos Attacks //Proceedings of IEEE GLOBE-

COM. 2002

[3] Alarcon-Aquino V, Barria J A. Anomaly Detection in Communication Networks Using Wavelets. IEEE Proc-Commun, 2001, 148(6)

[4] Barford P, Kline J, Plonka D, et al. A Signal Analysis of Network Traffic Anomalies //Proc. of ACM SIGCOMM Internet Measurement Workshop. Marseilles, France, November 2002: 412-423

[5] Gao Jun, Hu Guangmin, Yao Xingmiao. Anomaly Detection of Network Traffic Based on Wavelet Packet //APCC' 06. Asia-Pacific Conference on Communications. 2006

[6] Wenke L, Xiang D. Information-Theoretic Measures for Anomaly Detection //Proc. of IEEE Symposium on Security and Privacy. Oakland, CA, May 2001: 130-143

[7] Lakhina A, Crovella M, Diot C. Mining Anomalies Using Traffic Feature Distributions //Proc. of ACM SIGCOMM 2005. Philadelphia, Pennsylvania, USA, August 2005: 9-20

[8] http://www.apng.org/9thcamp/matbdfs.ppt

[9] 杨岳湘, 王海龙, 卢锡城. 基于信息熵的大规模网络流量异常分类. 计算机工程与科学, 2007, 29(2): 40-43

[10] Noble C C, Cook D J. Graph-based Anomaly Detection //SIGKDD' 03. Washington, DC, USA, August 2003

[11] Han Jiawei, Kamber M. Data Mining-Concepts and Techniques. Morgan Kaufmann Publishers, 2000

[12] [EB/OL]. http://www.internet2.edu/network/

[13] Dainotti A, Pescape A, Ventre G. Wavelet-based Detection of DoS Attacks //Proceedings of IEEE GLOBECOM. 2006

(上接第 28 页)

namic and fault-Tolerant service invocation //The 2nd Annual International Workshop of the Working Group on Web and Databases of the German Informatics Society. Germany, 2002

[13] Alwagait E, Ghandeharizadeh S. DeW: A dependable Web services framework //The 14th International Workshop on Research Issues on Data Engineering. USA, 2004

[14] Aghdaie N, Tamir Y. Implementation and Evaluation of Transparent Fault-Tolerant Web Service with Kernel-level Support //The IEEE International Conference on Computer Communications and Networks. Miami, Florida, 2002

[15] Jayasinghe D. FAWS for SOAP-based Web services- A client-transparent fault tolerance system for SOAP-based Web services. http://www-128.ibm.com/developerworks/webservices/library/ws-faws/, 2005

[16] Deron L, Fang C, Chen C, et al. Fault-tolerant Web service //The 10th Asia-Pacific Software Engineering Conference. Thailand, 2003

[17] Object Management Group. The Common Object Request Broker: Architecture and Specification, Chapter25: Fault Tolerant CORBA Specification. 2002

[18] Santos G T, Lung L C, Montez C. FTWeb: A Fault Tolerant Infrastructure for Web Services //The 2005 Ninth IEEE International EDOC Enterprise Computing Conference (EDOC' 05). Enscheda, the Netherlands, 2005

[19] Tan S, Vellanki V, Xing J, et al. Service Domains. IBM System Journal, 2004, 43(4): 734-755

[20] Birman K, van Renesse R, Vogels W. Adding High Availability and Autonomic Behavior to Web Services //The 26th International Conference on Software Engineering. Edinburgh, Scotland, United Kingdom, 2004

[21] 徐伟, 金蓓弘, 李京, 等. 一种基于移动 Agent 的复合 Web 服务容错模型. 计算机学报, 2005, 28(4): 558-567

[22] Dialani V, Miles S, Moreau L, et al. Transparent fault tolerance for Web services based architecturesEighth International Europar Conference (EUROPAR' 02). Germany, 2002

[23] Fabre J C, Perennou T. A metaobject architecture for fault-tolerant distributed systems: the FRIENDS approach. IEEE Transactions on Computers, 1998, 47(1): 78-95

[24] Beedubail G, Karmarkar A, Gurijala A, et al. An algorithm for supporting fault tolerant objects in distributed object-oriented operating systems //Fourth International Workshop on Object-oriented Operating Systems. Sweden, 1995

[25] Elnozahy M, Alvisi L, Wang Yi-Min, et al. A survey of rollback recovery protocols in message passing systems. ACM Computing Surveys, 2002, 33(3): 375-408

[26] Cristian F. Understanding fault-tolerant distributed systems. Communications of ACM, 1991, 34(2): 57-58

[27] Schneider F B. Implementing fault-tolerance services using the state machine approach. ACM Computing Surveys, 1990, 22(4): 299-320

[28] Budhiraja N, Marauller K, Schneider F B, et al. Optimal primary-backup protocols //Sixth International Workshop on Distributed Algorithm. Haifa, Israel, 1992

[29] Chen I R, Bastani F B. Warm standby in hierarchically structured process-control programs. Trans. on Software Engineering, 1994, 20(8): 658-663

[30] Veríssimo P, Rodrigues L, Rufino J. Delta4: a generic architecture for dependable distributed computing. Powell D. ESPRIT Research Reports. Berlin: Springer, 1991: 295-305

[31] Baldoni R, Marchetti C, Piergiovanni S T. Asynchronous active replication in three-tier distributed systems //The 2002 Pacific Rim International Symposium on Dependable Computing. Japan, 2002

[32] Chandra T D, Toueg S. Unreliable failure detectors for reliable distributed systems. Journal of the ACM, 1996, 43(2): 225-267

[33] Birman K P. Building secure and reliable network applications. Greenwich: Manning Publications, 1996

[34] Bertier M, Marin O, Sens P. Implementation and performance evaluation of an adaptable failure detector //IEEE Conference on Dependable Systems and Networks (DSN' 02). Washington D. C., 2002

[35] Stelling P, Foster I, Kesselman C, et al. A Fault Detection Service for Wide Area Distributed Computations //The 7th IEEE Symp. on High Performance Distributed Computing. Chicago, USA, 1998

[36] Tanenbaum A S, Van Renesse R. Distributed operating systems. ACM Computing Surveys, 1985, 17(4): 419-470