基于SOM的多层Web安全审计模型及其实现*)

郭 陟[1] 赵曦滨[2,1] 顾 明[1]

(清华大学软件学院 北京100084)[1] (江苏大学计算机科学与通信工程学院 江苏镇江212013)[2]

摘 要 随着Internet的普及,Web应用系统安全问题日益严重,而安全审计是保障信息系统安全的重要措施。可视化安全审计作为一种新的安全审计手段,有助于安全管理员充分理解主体行为,但在Web系统中却因运行效率等原因导致应用受到限制。本文定义了多层用户行为模型,以此模型作为Web安全审计的基础,并针对该行为模型提出了基于SOM的可视化算法。该算法将Web审计数据转换为形象的可视化信息,并且允许安全管理员对可视信息进行交互操作,进一步探索Web审计数据的内在关联,从而在保证可视化效果的同时,还可大幅提高安全审计的效率。

关键词 安全审计,SOM,可视化,Web架构

SOM-Based Multilayer Web Security Audit Model and its Implementation

GUO Zhi[1] ZHAO Xi-Bin[2,1] GU Ming[1]

(School of Software, Tsinghua University, Beijing 100084)[1]

(School of Computer Science and Telecommunications Engineering, Jiangsu University, Zhenjiang Jiangsu 212013)[2]

Abstract With the spread use of Internet, Web applications are facing more security problems. Security audit is one of the effective ways to secure information systems. With the use of visualization technique in security audit, security administrators can audit user behaviors easier. However, it's difficult to apply visualization to large volume of Web audit data due to low-efficiency of audit systems. This paper presents a multilayer user behavior model for Web security audit, and proposes a SOM-based visualization algorithm according to the behavior model. The algorithm transforms Web audit data to visual information, and then security administrators can operate audit systems with such information. Thus administrators can explore intrinsic relationship of Web audit data with highly-fidelity visual information while audit users with highly-efficient manner.

Keywords Security audit, SOM(self-organizing maps), Visualization, Web architecture

1 引言

随着Internet的飞速发展及相关技术的日趋完备,基于Web的应用在信息系统中所占的比重不断增加,新一代电子政务和电子商务系统大多采用了Web技术,在企业信息化和金融、保险等领域的Web应用也层出不穷,Web应用系统的开发与使用为全球化电子信息共享提供了坚实基础。然而,由于Internet的开放特性,Web应用系统面临的安全问题也十分严峻,研究如何提供安全可靠的运行环境来实施Web应用无论在理论上还是实践中都具有积极意义。其中,基于Web的安全审计是Web系统安全研究中的一个重点。

安全审计就是对有关操作系统、应用软件或用户活动所产生的一系列计算机安全事件进行记录和分析的过程[1]。在计算机网络中,安全管理员借助审计系统来监视系统状态和用户活动,并对日志文件进行分析,以及时发现系统中存在的安全问题。如文[2]所述,防火墙与密码系统等预防机制可以有效预防外部用户对Web系统的渗透,但无法预防内部用户对系统的渗透和对资源的滥用,通过对计算机系统进行安全审计则可发现内部用户渗透系统与滥用资源。此外,由于Web应用系统多用于提供开放服务,对内部用户的管理不规范,因此很难通过安全管理手段规范内部用户行为。某些Web应用系统(例如,数字图书馆、C2C电子交易系统)由于必须将系统向Internet用户开放,导致了内部用户的潜在威胁超过外部用户威胁。在这种环境中,安全审计成为了保障Web应用系统安全的重要手段。

由于Web服务器的活动可反映用户在Web应用系统中的活动,所以通过检查记录Web服务器活动规律的日志可了解用户活动规律,从而进一步监控用户行为,最终达到安全审计的目的。由此可知,检查日志数据是安全审计的一项重要工作。在日常管理中,安全管理员主要通过筛选、分类与查询日志来审计Web应用系统。Web应用系统中的用户和被监控资源数目众多,因而其应用系统日志往往是海量数据,所以安全管理员往往需要在软件工具的辅助之下来进行安全审计工作。目前,安全管理员采用手工检查日志文件或利用简单的日志统计软件的手段来进行安全审计[3]。这种审计方式效率低下,获得信息有限,难以真正通过审计手段保障Web应用系统安全。为便于安全管理员理解主体行为模式,就需要将描述主体行为的数据转换成可视的信息,以帮助安全管理员理解主体行为模式,并将这些信息展现在可视化人机接口中。

可视化审计方面的相关研究工作从九十年代初期就已展开,文[3,4]通过将可视化技术应用到安全审计领域,大大提高了安全管理员进行安全审计的效率。文[4]提出了一个可视化的日志审计系统—Tudumi,Tudumi将网络访问、用户登录等日志信息与安全审计规则可视化为直观的图像,并提供交

*)本文研究受国家863计划项目(Nos. 2001AA414220,2001AA414020,2002AA412010)资助。

互功能，从而使管理员可快速筛选、汇总网络访问、用户登录等日志数据，以利于及时发现用户的异常活动。文[4]的方法采用了按网络地址汇总的方式，适用于监控用户在网络系统中的活动，但该方法无法提供可视化的用户行为模式信息。由于审计内部用户行为是Web安全审计的主要目标，因此Web安全审计更关注从日志中筛选出用户行为模式，从而发现用户行为规律。所以该方法在Web安全审计中的应用受到限制。文[3]则关注网络活动的行为模式，将网络活动映射到可视的SOM图中。管理员可通过直接观察可视图来对网络活动进行分类，从而发现异常活动。SOM算法具有出色的聚类区分能力[5]，并易与可视化技术相结合，因此利用基于SOM的安全审计系统能有效发现相似的用户活动，总结出行为模式。文[3]的方法主要对审计事件进行分类与可视化，因此在处理Web应用系统的海量日志数据时遇到了困难：如果审计时间窗设置得太短，则无法观察长期的攻击或滥用行为，为审计造成困难；如果审计时间窗太长，则一次分类与可视化的审计事件数目太多，安全审计系统效率将受到影响。因此该方法在Web安全审计方面应用也受到限制。

本文结合SOM理论及可视化技术，并针对Web应用系统中海量日志所造成的审计效率低下等问题提出了多层安全审计模型。基于SOM的多层安全审计模型通过支持多层用户行为模型，不仅解决了可视化过程中的效率问题，而且处于不同层次的行为模型为安全管理员提供不同细节的视角。该模型使安全管理员在关注宏观用户行为模式时也能兼顾微观用户活动细节，从而高效地审计用户活动，并从中及时发现用户异常活动，达到安全审计之目的。

2 基于SOM的多层Web安全审计模型

2.1 多层安全审计模型

本文提出的安全审计模型如图1所示。

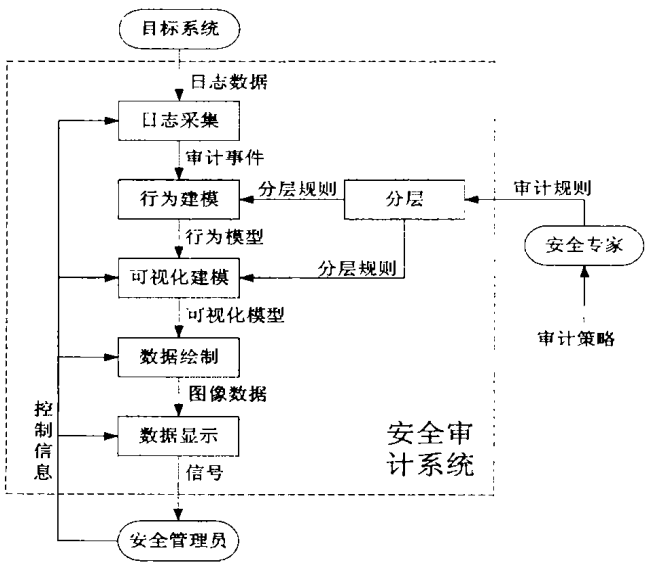

图1 多层安全审计模型

在图1的审计模型中，共有两个角色参与，即：

· 安全专家：安全专家根据经验，结合组织的安全策略，制定审计规则，由审计规则指导分层模型的建立。

·安全管理员：控制安全审计系统的运行；并根据可视化的审计信息来分析用户行为模式，从中辨识出异常用户活动。

安全审计系统的日志采集模块从目标系统获得记录用户活动的日志数据后，去除日志异构性，提取为统一的审计事件。行为建模模块根据分层规则，将审计事件转化为用户行为模型。可视化建模模块在分层规则的指导下将行为模型转换为可视化模型。最后，由数据绘制与显示模块将可视化信息传递给安全管理员。

2.2 多层用户行为模型

用户在Web应用系统中的活动记录在Web日志中。Web日志包括访问日志、错误日志等形式。在Web安全审计中，访问日志被广泛用于检测滥用资源、攻击网站等异常情况。以CLF(Common Log Format)[6]这种被广泛使用的Web访问日志格式为例，它包括远程主机名、用户远程登录名、用户名等主要条目，从上述格式日志数据的远程主机名、请求时间、请求内容中可提取出审计事件。

在明确审计对象后，存在如下集合

$$Sets: Subject, Objects, Timestamp \tag{1}$$

其中，*Subjects* 代表主体标识。在Internet应用程序日志采集过程中，该标识一般取远程用户的网络地址。*Objects* 代表客体标识。在Internet应用程序的入侵检测过程中，该标识一般取被访问的资源名或被执行的命令名。*Timestamp* 代表审计事件发生的时间戳。在Internet应用程序中，一般采用访问资源或执行命令的时间，经常取日志中的访问请求时间。

则审计事件 *Events* 形式具体为

$$Events \subseteq Subjects \times Objects \times Timestamp \neq \Phi \tag{2}$$

在审计事件基础上，可确定审计事件构成的数据集，记为

$$Dataset \in 2^{Events} \tag{3}$$

在审计数据集基础上，可对用户行为进行建模，建模过程如下：

·确定行为建模涉及的主体范围，即确定需要审计用户范围。

·根据审计事件的客体信息与安全策略，确定采用何种审计测度(Measure)。

·根据主体与时间戳，辨识出用户会话(User session)。

· 针对每一主体，建立行为模型。

本文采用的审计测度选取事件或频率来度量，采用滑动时间窗作为用户会话的划分标准。为达到提高Web安全审计系统效率的目的，本文提出如下多层用户行为模型作为Web安全审计的基础。

从组织的安全审计策略，可提取出审计规则，记为

$$Filter \subseteq Timestamp \times Timestamp \tag{4}$$

$$Interval \subseteq Timestamp \times Timestamp \tag{5}$$

其中，*Filter* 表示当前的滑动时间窗口。*Interval* 表示划分会话的时间间隔。

依据审计规则中的滑动时间窗口，对审计数据集进行处理，可得过滤后的数据集，记为

$$dataset(Filter) = \{x \mid x \in Dataset \wedge x = (s,o,t) \wedge \exists \langle t_1, t_2 \rangle \in Filter \wedge t_1 \leqslant t \leqslant t_2 \wedge s \in subjects \wedge o \in Objects \wedge t \in Time\} \tag{6}$$

在过滤完数据集后，依据审计规则中的划分会话标准，可得用户会话，记为

$$Session(dataset(Filter), Inteerval) = \{x \mid x \in adataset(Filter) \wedge x = (s,o,t) \wedge \exists \langle t_1, t_2 \rangle \in Interval \wedge t_1 \leqslant t \leqslant t_2 \wedge s \in Subjects \wedge o \in Objects \wedge t \in Time\} \tag{7}$$

根据上述数据，可得单层用户行为模型，记为

$$\psi = f(dataset(Filter), Interal) = \{S_i \mid S_i \in Session(dataset(Filter), Interval) \wedge 1 \leqslant i \leqslant n\} \tag{8}$$

其中，n 为行为模型中用户会话的数目，n 为自然数。

上述单层行为模型还可表示为如下矩阵形式，记为矩阵 M

$$M=\begin{array}{c} \\ \text{用户会话} \end{array}\begin{array}{c} \text{审计测度} \\ 1\ \cdots\ j\ \cdots\ p \\ \begin{matrix}1\\ \vdots\\ i\\ \vdots\\ n\end{matrix}\begin{bmatrix} & & \\ & \vdots & \\ \cdots & x_{ij} & \\ & & \end{bmatrix} \end{array}$$

在上述矩阵 M 中，每一个 p 维行向量代表主体的一次会话，每一个 n 维列向量代表一种审计测度。

2.3 多层可视化映射算法

文[7]中提到，可采用矩阵 M 行向量间的关联关系来描述会话或审计事件间的背离程度。关联关系越强。会话间的相似程度越高；会话间关联关系越弱，相似程度越低。数学上，可采用关联系数 r_{ij} 来度量关联程度，计算方法见式(9)。

$$r_{ij}=\frac{1}{p}\sum_{k=1}^{p}\frac{(x_{ik}-\bar{x}_i)(x_{jk}-\bar{x}_j)}{\sigma_i\sigma_j} \tag{9}$$

其中，

$$\bar{x}_i=\frac{\sum_{j=1}^{p}x_{ij}}{p}\quad 1\leqslant i\leqslant n \tag{10}$$

$$\sigma_i=\sqrt{\frac{\sum_{j=1}^{p}(x_{ij}-\bar{x}_i)^2}{p}}\quad 1\leqslant i\leqslant n \tag{11}$$

对矩阵 M 按行进行完归一化计算后，2个归一化的 p 维行向量在欧氏空间的距离与 r_{ij} 成正比，即有式(12)。

$$d^2(x_i,x_j)=2(1-r_{ij})\quad 1\leqslant i,j\leqslant n \tag{12}$$

其中，x_i 表示第 i 个归一化的行向量。

基于统计假设[8]，正常的主体行为有相似的行为模式，体现为归一化的行向量在空间中的聚类(Cluster)；异常的主体行为往往背离正常主体行为模式；在用户活动中，正常活动处于主导地位。基于该假设，本文采用 SOM 距离矩阵来为安全管理员提供可视化的聚类信息，并利用多层行为模型与可视化模型来减少可视化建模的效率问题。

假设构造 L 层用户行为模型，记为

$$\Psi=\{\psi_k\mid 1\leqslant k\leqslant L\} \tag{13}$$

其中，ψ_k 表示第 k 层行为模型。考虑将多层可视化模型建立在网格图形基础上。

$$G_{r\times c,k}=\{grid_{(i,j)}\mid 1\leqslant i\leqslant r,1\leqslant j\leqslant c\} \tag{14}$$

其中，$G_{r\times c,k}$ 表示第 k 层的网格面。$grid_{ij}$ 表示网格面中第 i 行第 j 列的单个网格点，r 为网格面的行数，c 为列数。

在可视化建模过程中，用户行为模型被映射到网格面中，并转换为图形方式。该过程定义为如下函数：

$$\Gamma:2^{Session}\rightarrow G_{r\times c} \tag{15}$$

在多层可视化映射模型中，k 层第 α 行 β 列的网格点所对应的行为模型 ψ_k 向 $k+1$ 层映射记为

$$G_{r\times c,k+1}=\Gamma(\psi_k) \tag{16}$$

$$\Gamma(\psi_k)_{(\alpha,\beta)}=\{S_{i,j}^{k+1,t}\mid 1\leqslant t\leqslant T\} \tag{17}$$

其中，Γ 是自组织映射网络的学习算法 SOM[9,10]，具体算法本文就不再赘述。$S_{i,j}^{k+1,t}$ 代表第 $k+1$ 层中的第 t 个用户会话，该会话位于第 i 行第 j 列网格点内，T 为位于 $k+1$ 层中的会话总数。

根据上述映射结果，可获得 $k+1$ 层的行为模型，记为

$$\psi_{k+1}=f(\bigcup_{t=1}^{T}S_{i,j}^{k+1,t},Filter_{k+1},Interval_{k+1}) \tag{18}$$

其中 $Filter_{k+1}$ 在第 $k+1$ 层的滑动时间窗口，$Interval_{k+1}$ 表示第 $k+1$ 层划分会话的时间间隔。

3 实验结果

基于本文提出的 Web 安全审计模型，笔者构造了原型系统，并利用某大学图书馆 Web 服务器的日志数据进行了实验。在实验中，采用了3层安全审计模型。第1层的用户行为模型以天为单位划分用户会话，将2周内39个用户的日志数据(含134,072个审计事件)可视化后得到如图2所示实验结果。

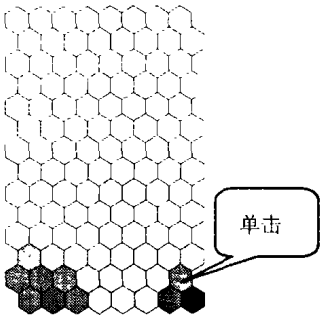

图2 以天为单位划分用户会话的实验结果

图2中标识的网格点介于正常与异常之间，需要深入调查。单击图2中标识的网格点，可展开第2层可视化 SOM 图观察细节。第2层的用户行为模型以小时为单位划分用户会话，将3个用户的日志数据(含22,010个审计事件)可视化后得到如图3所示实验结果。

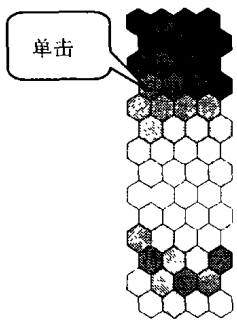

图3 以小时为单位划分用户会话的实验结果

单击图中标识的网格点，可展开第3层可视化 SOM 图。第3层的用户行为模型以单个审计事件为单位划分用户会话，将2个用户的日志数据(含32个审计事件)可视化后得到如图4所示实验结果。

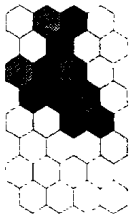

图4 以事件为单位划分用户会话的实验结果

由上述实验结果，可以将异常会话定位到事件级别，这样将显著提高安全审计效率。事实证明，该审计结果所表现的异

(下转第99页)

3 基于幂零泛与运算模型的模糊逻辑命题演算系统

前面已经证明泛逻辑的零级泛与运算 $T(x,y,h)=(\max(0,x^m+y^m-1))^{1/m}$ 任意 $x,y\in[0,1]$，其中 $m=(3-4h)/(4h(1-h))$，$m\in(0,+\infty)$，$h\in(0,0.75)$ 是幂零三角范数，而且与有界与算子 $T_1(x,y)=\max(0,x+y-1)$ 同构。设三角范数 $T(x,y,h)(h\in(0,0.75))$ 是命题联结词"&"的真值函数，$T(x,y,h)(h\in(0,0.75))$ 的剩余 $I(x,y,h)(h\in(0,0.75))$ 是命题联结词"→"的真值函数。

定义3.1 由泛与运算模型 $T(x,y,h)(h\in(0,0.75))$ 确定的命题演算系统记为 PC(T)。$S=\{\underline{0},P_1,P_2,\cdots\}$ 是可数集，¬，→分别是一元运算与二元运算，由 S 生成的{¬，→}型自由代数记作 $F(S)$。$F(S)$ 中的元素叫命题、句子或公式，S 中元素叫原子命题或原子公式。

在 PC(T)中，命题联结词¬，→的真值函数分别是 $N(x,k)=(1-x^m)^{1/m}$，$I(x,y,h)=(\min(1,1-x^m+y^m))^{1/m}(h\in(0,0.75))$，命题$\underline{0}$的真值是0。另外我们可定义其它命题联结词：

$P\&Q:\neg(P\to\neg Q)$　　$P\wedge Q:P\&(P\to Q)$

$P\vee Q:(P\to Q)\to Q$　　$\neg P:P\to\underline{0}$

$P\equiv Q:(P\to Q)\&(Q\to P)$

定义3.2 命题演算系统 PC(T)的一个赋值是：映射 $e:F(S)\to[0,1]$，且满足 $e(0)=0$，$e(P\to Q)=e(P)\to e(Q)$，$e(P\&Q)=e(P)*e(Q)$，其中等式右边的"→"，"*"分别代表泛逻辑中的泛蕴涵运算与泛与运算。

命题3.1 对任意公式 $P,Q\in F(S)$，$e(P\wedge Q)=\min(e(P),e(Q))$，$e(P\vee Q)=\max(e(P),e(Q))$。

定义3.3 设 $P\in F(S)$，如果对任意一个赋值 e，均有 $e(P)=1$，称 P 是 PC(T)的1-重言式。

定理3.1 下列公式都是 PC(T)的1-重言式：(1)$P\to(Q\to P)$；(2)$(P\to Q)\to((Q\to R)\to(P\to R))$；(3)$(\neg P\to\neg Q)\to(Q\to P)$；(4)$((P\to Q)\to Q)\to((Q\to P)\to P)$；(5)若 P，$P\to Q$ 都是 PC(T)的1-重言式，则 Q 也是 PC(T)的1-重言式。

证明：任意 $P,Q,R\in F(S)$，设 e 为任意一个赋值函数，$e(P)=x$，$e(Q)=y$，$e(R)=z$。

(1)$x\to(y\to x)=x\to(\min(1,1-y^m+x^m))^{1/m}=1$；

(2)由命题2.4，$x*(x\to y)=\min(x,y)\leqslant y$，$y*(y\to z)\leqslant z$，所以，$((x\to y)*(y\to z))*x=(x*(x\to y))*(y\to z)\leqslant y*(y\to z)\leqslant z$。再由命题2.2(2)，$(x\to y)*(y\to z)\leqslant x\to z$，$x\to y\leqslant(y\to z)\to(x\to z)$。从而，$1\leqslant(x\to y)\to((y\to z)\to(x\to z))$，所以，$(x\to y)\to((y\to z)\to(x\to z))=1$；

(3)因为 $(x\to 0)=(1-x^m)^{1/m}$，$(y\to 0)=(1-y^m)^{1/m}$，则 $(x\to 0)\to(y\to 0)=(\min(1,1-y^m+x^m))^{1/m}$，又 $x\to x=1$，所以，$((x\to 0)\to(y\to 0))\to(y\to x)=1$；

(4)由命题2.4直接可证。

(5)因为 $e(P)=1$，$e(P\to Q)=e(P)\to e(Q)=1\to e(Q)=1$，而 $1\to e(Q)=e(Q)$，从而，$e(Q)=1$。

定义3.4 由泛与运算模型 $T(x,y,h)(h\in(0,0.75))$ 确定的模糊命题演算系统 PC(T)由下面两部分组成：

(Ⅰ)PC(T)中公理：设 $S=\{\underline{0},P_1,P_2,\cdots\}$ 是可数集，$F(S)$ 是由 S 生成的{¬，→}型自由代数，$F(S)$ 中以下形式的公式称为公理：

(1)$P\to(Q\to P)$；(2)$(P\to Q)\to((Q\to R)\to(P\to R))$；(3)$(\neg P\to\neg Q)\to(Q\to P)$；(4)$((P\to Q)\to Q)\to((Q\to P)\to P)$。

(Ⅱ)PC(T)中的推理规则为分离规则 MP：由公式 P，$P\to Q$ 可推得 Q。

定理3.2 由泛逻辑的零级幂零泛与运算模型 $T(x,y,h)(h\in(0,0.75))$ 确定的模糊命题演算系统 PC(T)与 Lukasiewicz 逻辑命题演算系统是等价的。

关于泛逻辑的零级泛运算模型中，h 在其它区间的情形，我们另文讨论。

参考文献

1 何华灿. 泛逻辑学原理[M]. 北京：科学出版社，2001

2 Klement E P, Mesiar R, Pap E. Triangular Norms [M]. Kluwer Academic Publishers, 2000

3 Hajek P. Matamathematics of Fuzzy Logic [M]. Kluwer Academic Publishers, 1998

4 Hajek P. Observations on the Monoidal t-norm Logic [J]. Fuzzy Set and Systems, 2002(132): 107～112

5 Burris S. Sankappanavar H P. A Course in Universal Algebra [M]. Springer-verlag New York Heidelberg Berlin, 1981

(上接第53页)

常是由于某校外 IP 开设了针对图书馆的代理服务所引起的。

结论 本文针对现有可视化审计方法在 Web 应用环境中的效率问题，定义了多层用户行为模型，并针对该行为模型提出了基于 SOM 的可视化算法，用以为安全管理员提供可视化的审计信息。实验表明，采用本文方法有利于安全管理员高效地进行安全审计工作，并且在可视化算法中采用多层行为模型可大大提高安全审计系统的效率。利用本文的安全审计模型，为 Web 安全审计提供了一条新的途径。

依据本文定义的分层模型可有效提高交互速度，提高审计效率，但如何划分行为模型的层次需依据审计规则。因此，在今后工作中，笔者将进一步研究根据组织安全策略制定审计规则 Filter 与 Interval 的方法。

参考文献

1 戴英侠，连一峰，王航. 系统安全与入侵检测. 北京：清华大学出版社，2002

2 Anderson J. P. Computer security threat monitoring and surveillance: [Technical Report]. James P. Anderson Co., Fort Washington, PennsyIvania, 1980

3 Girardin L, Brodbeck D. A Visual Approach for Monitoring Logs. In: Proc. of the Twelfth Systems Administration Conference (LISA'98), 1998. 299～308

4 Takada T. Koike H. Tudumi: Information Visualization System for Monitoring and Auditing Computer Logs. In: Proc. of the Sixth Intl. Conf. on Information Visualisation (IV'02), 2002

5 de Backer S, Naud A. Scheunders P. Non-linear dimensionality reduction techniques for unsupervised feature extraction. Pattern Recognition Letters, 1998, 19: 711～720

6 Logging Control In W3C httpd. http://www.w3.org, 1995

7 Lam K-Y, Hui L, Chung S-L. Multivariate Data Analysis Software for Enhancing System Security. J. Systems Software, 1995, 31: 267～275

8 Denning D E. An intrusion detection model. IEEE Trans. on Software Engineering, 1987, SE-13: 222～232

9 Kohonen T. Self-Organizing Maps. Berlin: Springer, 2001

10 边肇祺，张学工，等. 模式识别. 北京：清华大学出版社，1999