基于模式挖掘的 IDS 警报分析[1]

梅海彬　龚俭

（东南大学计算机系 江苏省计算机网络技术重点实验室 南京 210096）

摘　要　目前 IDS 报告的警报过多且包含大量误报，这给管理员的分析和系统的自动响应带来困难，因此对警报的自动分析和误报消除已变得非常重要。本文观察到警报数据中存在具有规律性大量频繁出现的警报，并且该类警报多为误报或噪音的特性，因而提出了一种基于模式挖掘来发现警报中这些规律，进行警报分析的新方法。利用该方法可挖掘警报中的频繁模式，辅助管理员进一步分析和处理警报。试验结果表明，管理员利用该方法来分析和处理警报后，能减少警报数目 50%以上，从而有效降低误报率。

关键词： 入侵检测系统；网络安全；警报分析；数据挖掘；频繁模式

Pattern Mining Based Analysis of IDS Alarms

Mei Haibin　Gong Jian

(Department of Computer Science and Engineering, Southeast University Computer Network Technology Key Laboratory of Jiangsu Province, Nanjing 210096, China)

Abstract: It is a well-known problem that intrusion detection systems tend to generate a large amount of alerts, and usually mixes with many false alarms. As a result, it is difficult for operator or intrusion response systems to understand the intrusions behind the alarms and take appropriate actions. How to analyze alarms automatically and eliminate the false alarms has become very important. This paper observes that a lot of alarms occur regularly in the whole alarms which an intrusion detection system triggers and these alarms are often false alarms or noise. Therefore this paper proposed a novel approach to find these rules and analyze IDS alarms based on pattern mining. By mining the frequent patterns, our approach is able to help operator analyze and deal with alarms more easily. Our preliminary experiments demonstrate that with this approach operator can reduce more than 50% of the alarms and decrease the false alarm rate efficiently.

Key words: Intrusion Detection System; network security; alarm analysis; data mining; frequent pattern

1. 引言

入侵检测系统（Intrusion Detection System，简称 IDS）是保障网络安全的有效工具，它能够检测网络上的攻击行为，并提示系统管理员进行及时响应，以避免入侵带来的损失[1][2]。但是目前 IDS 技术还不成熟，无论是基于滥用检测的 IDS 还是基于异常检测的 IDS 都普遍存在报告的警报数目过多，且警报中包含大量的误报（false positive）的问题[3][4][5]。这些数目众多又含有大量误报的警报数据使得管理员很难从中找到真正的警报（true positive），也限制了 IDS 的自动响应，IDS 的价值被大大地削弱。

鉴于此，研究者提出了多种方法来试图解决该类问题，一种途径是从底层，从根本上找到一种好的检测模型来提高 IDS 的检测准确度，减少 IDS 产生的误报数。但实际上，研究有效的检测模型难度较大，文献[8]用贝叶斯理论对此进行了分析说明。因此，有些研究者开始从高层对 IDS 产生的警报进行分析，利用警报中的规律来减少 IDS 的警报数目和误报数目，这成为了近来 IDS 领域中一个新的研究热点。

本文通过对 Monster3.0 系统[2] 产生的警报进行观察，发现在警报中存在大量频繁出现的警报，并

[1]基金项目：国家自然科学基金课题资助（90104031）。

[2] Monster3.0 系统是 CERNET 华东北地区网络中心开发的面向大规模网络的分布式入侵检测与预警系统。

具有一定的规律性，即警报的频繁模式。例如某些主机每天总是产生同种类型警报，且数量较多，并且这些警报往往是误报或噪音（noise, 噪音是指 IDS 的诊断是正确的，但警报所指示的攻击对被保护的站点没有威胁或不起作用[11]）。而这些警报往往很难由 IDS 系统本身的改进来消除掉，因为这些频繁模式的产生是多方面的，例有为 windows 操作系统的机器提供升级服务的服务器，当客户端向其请求 windows 升级服务时会使 IDS 触发“WEB-MISC whisker HEAD with large datagram”警报；也有具有 NAT 功能的防火墙，当它代理多个用户主机同时向外发送起服务请求时，发出的多个 SYN 报文就会使 IDS 触发“SYN host sweep”警报；也有当一些网络管理工具来查询一些比较敏感的 MIB 变量时也会触发 IDS 警报；还有当 Macintosh FTP 客户端在 FTP 连接上发送 SYST 命令时，将会触发“FTP SYST command”警报等等[7]。

此外，警报中的这些规律很难靠简单统计方法来发现，除非进行较多的人工指导，例如用简单统计方法可以统计出每天出现次数较多的警报，但出现多的警报是否具有规律性，以及是怎样的规律还需要用户进一步的分析。

据此，本文提出了一种基于模式挖掘的警报分析方法来自动发现警报中的这些规律，以方便用户建立相应的措施来消除这些警报。该方法基本思路是：首先是利用数据挖掘的方法来发现警报中的频繁模式，然后将这些模式反馈给管理员，由管理员对模式进行分析，并根据分析结果来判断挖掘的频繁模式是否为误报或噪音警报的模式，如果是，就可以通过建立过滤器来滤掉这些警报报或对 IDS 的检测规则进行修改或更改网络配置等，使 IDS 以后不再报告这些警报。

本文提出的方法和[6][7][9]方法的目标基本一致，但本文采用频繁模式挖掘的方法来挖掘警报中的频繁模式，不需要用户给定网络的层次结构。此外，本文将挖掘出的模式反馈给用户，由用户作进一步的分析，是一种交互式半自动的警报处理模式，不会像文献[3][5]中提出的方法那样漏掉真正的警报。实验结果表明本文方法能很好地帮助管理员分析和处理警报，经过此方法分析处理警报后可以大大减少警报数目，减低误报率。

以下为本文的组织结构：第 2 节简要介绍相关的工作；在第 3 节说明了方法的实现；在第 4 节给出了试验结果和相关分析；最后是对本文的总结。

2. 相关工作

用数据挖掘的方法来分析和处理入侵警报，已经有些人在这方面做了一定的工作。Manganaris[3]等人将连续的警报流划分为很多警报区间（alarm bursts），并把每个警报区间对应一个事务（transaction），警报区间的警报对应事务中的项，然后在这些警报区间上挖掘警报关联规则。只要随后的警报符合这些关联规则，就认为是正常的警报，并被忽略掉。这种方法可以大大减少需要管理员查看的警报数目，但缺点是可能把真正的警报忽略掉。Clifton和Gengo[5]认为，警报中有很多误报是由于在特定的环境里具有与入侵相似特征的正常操作引发的，并且由这些正常操作引发的警报具有一定的序列模式。他们利用序列挖掘的方法挖掘出警报流中的这些序列模式，并以这些序列模式来建立警报过滤器，来过滤符合序列模式的误报。这种方法也可以减少警报中的误报数，但也有可能把真正的警报过滤掉。Klaus Julisch[6][7][9]发现在警报中存在大量的警报是由于少数根源(root cause)产生的，并且这些警报具有相似性，规律性和持续性。他提出利用数据挖掘中层次聚类的方法来聚合具有同样产生根源的警报，使管理员能方便地发现产生这些警报的根源，从而可以对网络进行配置来消除这些根源或建立过滤器来消除这些由根源产生的警报，但该方法的缺点是聚类时要用到泛化层次图，而层次图需要用户根据网络的特定结构来建立，此外，该方法在聚类时每个类的警报数也要用户来定义，定义太大和太小都不合适，定义大了就会将不同根源的警报聚成一类，定义太小就会将同根源的警报聚成不同的类。

3. 频繁模式挖掘算法

在数据挖掘中，频繁模式挖掘是挖掘频繁项集。设 I={i1,i2,…,in}为项的集合，I 中的元素为项（Item），令 X⊆I，则 X 为项集（itemset），包含 k 个项的项集称为 k-项集，设 D 为事务数据库，D={t1,t2,…,tm}，ti⊆I，ti 称为事务，当项集在 D 中的出现次数大于给定的阈值 s，就称该项集为频繁

项集。在数据挖掘中挖掘频繁项集的常见算法有 Apriori 算法，FP-Tree 算法等[10]。

本文的模式挖掘采用数据挖掘中 Apriori 频繁模式挖掘算法，该算法挖掘事务数据库中的频繁项集，事务数据库中的每条事务由事务 ID 和事务中的项（Item）组成，如下表（1）所示，而 IDS 报告的警报记录通常为下表（2）中的格式（本文中只取了警报的五个主要属性，其中，EventType 为警报类型，SIP 为警报源 IP 地址，DIP 为警报目的 IP 地址，SPort 为警报源端口，DPort 为警报目的端口）。

表（1）事务数据格式

事务 ID	事务中的项
T100	I1,I2,I5
T200	I1,I2,I5
T300	I3,I4
T400	I1,I2,I3,I4

表（2）警报数据格式

EventType	SIP	DIP	SPort	DPort
EventType1	IP1	IP2	Port1	Port3
EventType2	IP2	IP3	Port2	Port1
EventType2	IP3	IP2	Port3	Port2
…	…	…	…	

从上可以看出，为了在警报数据上用 Apriori 算法来挖掘频繁项集，需要对警报数据格式进行一定的转换，使之符合事务数据的格式。

本文将每条警报记录看为一条事务记录，将每条警报记录的每个字段的属性名和对应值构成的二元组看为事务的一个项，即让 i_j=（Property_Name，Property_Value），其中，Property_Name 为警报的某个属性名，它可取 EventType、SIP、DIP、Sport 或 DPort 值，变量 Property_Value 为对应的警报属性值。则转换后，表（2）的警报记录所对应的事务记录为下表（3）。

表（3）转换后的警报数据格式

事务 ID	事务中的项
1	(EventType,EventType1),(SIP,IP1),(DIP,IP2),(Sport,Port1),(DPort,Port3)
2	(EventType,EventType2),(SIP,IP2),(DIP,IP3),(Sport,Port2),(DPort,Port1)
3	(EventType,EventType2),(SIP,IP3),(DIP,IP2),(Sport,Port3),(DPort,Port2)
…	…

由于转换后的事务中的项是个二元组，而不同于常规的事务中的项，所以，本文对基本的 Apriori 算法要进行适当的改动，使之能够满足挖掘警报中频繁模式的要求。以下为相关的定义和算法的伪代码。

3.1. 相关定义

为了便于算法的理解，在介绍具体算法之前，先给出几个定义。

***定义 1*：警报：**警报 A 为 n 元组（a1,a2,…,an），n 为警报属性个数，ai∈Value_Ai，Value_Ai 为警报第 i 个属性的值域。

***定义 2*：警报属性频繁项：**为二元组（Property_Name, a），其中，Property_Name 为 a 的属性名，a 为在 Property_Name 对应的警报属性列中出现的次数大于给定阈值 s 的属性值。记为 **FA**，以下简称**频繁项**。

***定义 3*：警报属性频繁项集：**为警报属性频繁项的集合，记为 **FA-Set**={(Property_Name,a)|（Property_Name,a）为 FA}，简称**项集**，当项集在警报数据中的出现次数大于给定的阈值，就称该项集为**频繁项集**，包含 k 个频繁项的**频繁项集**，称为**频繁 k-项集**，记为 $\mathbf{AL_k}$。

***定义 4*：警报频繁模式：F-Pattern =** $\bigwedge_{i=1}^{k} \mathbf{FA_i}$ 且 $\mathbf{FA_i} \in \mathbf{AL_k}$，即定义为一频繁 k-项集的频繁项的合取。

3.2. 算法设计

输入：警报数据 AR，指定的阈值 s ；

输出：警报数据的所有频繁项集 Result；
方法步骤：

1. Result=∅;
2. 转换警报数据使之满足事务数据的格式；
3. 扫描警报数据，生成 AL_1;
4. for (k=2; $AL_{k-1} \neq \varnothing$;k++){
5. C_k = AC_Gen(AL_{k-1},s);//产生候选频繁项集 C_k;
6. 为 C_k 中每个项集生成一个计数器;
7. for(i=1;i≤|AR|;i++){//|AR|表示 AR 中记录的条数;
8. 对第 i 个警报记录 A 所支持的每个 C_k 中项集，让项集的计数器加 1;
9. }
10. AL_k=C_k 中满足大于 s 的所有项集;
11. Result = Result ∪ AL_k;
12. }
13. return Result;

产生候选频繁项集时，与原 Apriori 算法存在不同，以下是改动后的产生候选频繁项集的过程。

```
procedure AC_Gen(AL_{k-1},s)
{
  C_k = ∅;
  for each I∈ AL_{k-1} {
    for each J≠I∈ AL_{k-1} {
      if 在 I 和 J 中有 k - 2 个元素相等，且剩余的那个元素满足对应的两项均不相等 then
        if { I ∪ J }的所有子集∈ AL_{k-1} then C_k=C_k ∪ {I ∪ J}
    }
  }
  return C_k;
}
```

在频繁项集产生后，就可生成频繁模式 **F-Pattern**，方法是对每个频繁项集 L_k，取出项集中的每项 i，并取它们用“∧”符连接起来。例：假设最后生成的频繁项集为：

{{(EventType, CHAT MSN user search),(SIP,111.111.111.23), (DPort, 1863)},
{(EventType, ICMP Destination Unreachable) ,(SIP,12.12.12.12),(DSP,123.123.123.123)}}，

则可生成两条频繁模式：

1. (EventType, CHAT MSN user search) ∧ (SIP,111.111.111.23) ∧ (DPort, 1863)；
2. (EventType, ICMP Destination Unreachable) ∧ (SIP,12.12.12.12) ∧ (DSP,123.123.123.123)。

这两条模式表示的含义分别是事件类型为 CHAT MSN user search 且源地址为 111.111.111.23 且目标端口为 1863 警报和警报类型为 ICMP Destination Unreachable 且源地址为 12.12.12.12 且目的地址为 123.123.123.123 的警报在警报数据中频繁出现。

4. 试验结果及数据分析

本文收集了Monster 3.0系统对CERNET江苏省网主干一周（2005/04/17~2005/04/23）的入侵检测的警报数据作为模式挖掘的数据集，该数据集包含208012条警报记录，为了挖掘的目的，本文将警报记录看为一个5元组，即：（警报类型，源IP地址，源端口，目的IP地址，目的端口）。

挖掘的方法是先对每天的警报数据进行频繁模式挖掘，挖掘其中的频繁模式，然后，再从每天的频繁模式中找出在 7 天中都出现的频繁模式，最后将结果反馈给管理员，并由管理员作进一步的分析。以下是挖掘的结果：表（4）是在给定支持度为 200 的情况下，在 2005/4/17 日的警报记录中挖掘出的出现频率位于前 5 位的频繁模式；表（5）为在 7 天中，每天都出现的频繁模式，其中表的最后一列表示该模式的警报在警报数据中出现的次数；图 1 列出了 7 天中，每天总的警报数和满足模式的警报数的对比情况，其中 X 轴为时间（单位：日/月），Y 轴为警报个数（单位：条）。

表（4）：2005/4/17 日的警报记录中的频繁模式

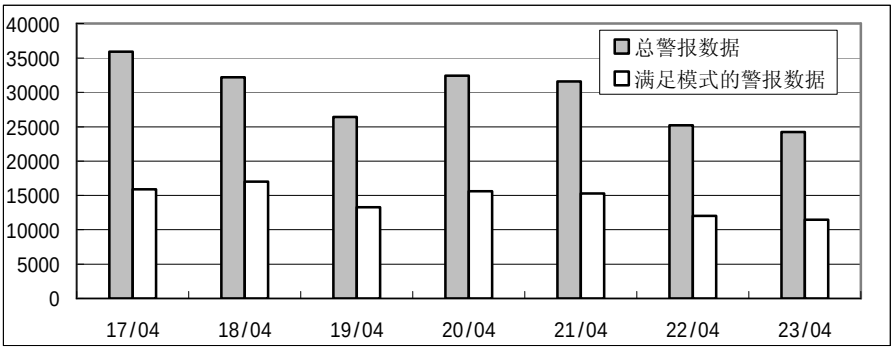
序号	频繁模式	出现频率
1	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 211.xx.xx.248) ∧ (SPort,80)	5698
2	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 211.xx.xx.99) ∧ (SPort,80)	2315
3	(EventType, ICMP Destination Unreachable) ∧ (SIP, 61.xxx.xxx.103) ∧ (SPort, 778)	1899
4	(EventType, P2P GNUTella GET) ∧ (SIP, 202.xxx.xx.36) ∧ (DPort,6346)	1093
5	(EventType, WEB-MISC whisker HEAD with large datagram) ∧ (SIP, 202.xxx.xx.36) ∧ (DPort,80)	1015

表（5）：2005/4/17 ~ 2005/4/23 一周每天都出现的频繁模式

序号	频繁模式	出现频率
1	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 211.xx.xx.248) ∧ (SPort, 80)	27805
2	(EventType, SCAN Proxy Port 8080 attempt) ∧ (DIP,202.xxx.x.10) ∧ (DPort,8080)	9211
3	(EventType, P2P GNUTella GET) ∧ (SIP,202.xxx.xx.36) ∧ (DPort, 6346)	8239
4	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 202.xxx.xx.40) ∧ (SPort,80)	6907
5	(EventType, P2P GNUTella GET) ∧ (SIP, 202.xxx.xx.36)	6582
6	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 211.xx.xx.68) ∧ (SPort,80)	6437
7	(EventType, WEB-MISC whisker HEAD with large datagram) ∧ (SIP,202.xxx.xx.36) ∧ (DPort,80)	6197
8	(EventType, CHAT MSN user search) ∧ (SIP,202.xxx.xx.36) ∧ (DPort,1863)	5425
9	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SIP, 202.xxx.xx.47) ∧ (SPort,80)	4298
10	(EventType, P2P GNUTella GET) ∧ (DIP, 202.xxx.x.10) ∧ (DPort,8080)	4198
11	(EventType, FTP large SYST command) ∧ (DIP, 202.xxx.xx.31) ∧ (DPort,21)	3897
12	(EventType, WEB-IIS scripts access) ∧ (DIP,202.xxx.x.199) ∧ (DPort,80)	3440
13	(EventType, ATTACK-RESPONSES 403 Forbidden) ∧ (SPort,80) ∧ (DIP, 202.xxx.xx.36)	3227

14	(EventType, P2P GNUTella GET) ∧ (SIP,202.xxx.xx.36) ∧ (DPort,8080)	2264
15	(EventType, P2P GNUTella GET) ∧ (SIP, 202.xxx.xx.23) ∧ (DPort,8000)	2085
16	(EventType, P2P GNUTella GET) ∧ (SIP,202.xxx.xx.36) ∧ (DPort,8000)	1946
17	(EventType, SCAN Proxy Port 8080 attempt) ∧ (DIP,202.xxx.xx.35) ∧ (DPort,8080)	1915
18	(EventType, FTP large SYST command) ∧ (SIP, 202.xxx.xx.34) ∧ (DPort,21)	1761

图（1）：2005/4/17 ~ 2005/4/23 每天总的警报数与满足模式的警报数对比图

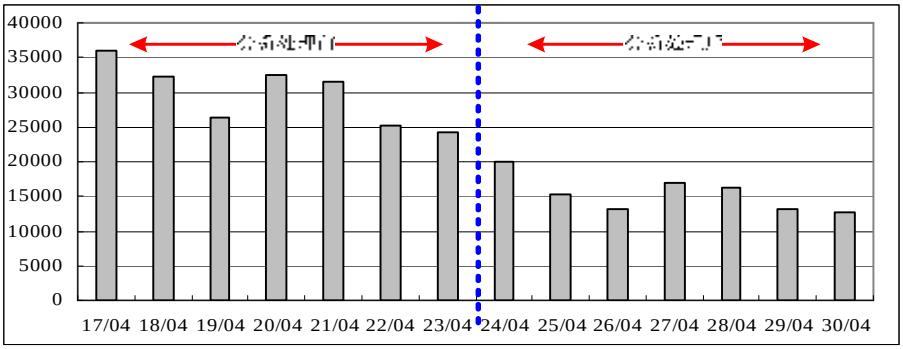


从上表（5）和图（1）可知，无论从 7 天总的满足频繁模式的警报（105834 条），还是每天满足模式的警报数来看，满足模式的警报都占警报数据的一半以上。虽然满足模式的警报数量较大，但这些模式的警报是否为误报或噪音，还需要管理员的核实分析，接下来的工作就是要对这些模式进行分析，找出产生这些频繁模式的原因，核实该类警报是否误报或噪音，并做出相应的处理方法。

经过分析发现，对于表（5）中 1 号模式的警报所对应的 IDS 检测规则是：当 Web 服务器响应客户端时，在响应串中含有“HTTP/1.1 403”内容。通常，Web 服务器发现客户端在没授权的情况下访问服务器上的资源时，服务器就会向客户端发出该类响应。但实际上，211.xx.xx.248 为一台惠普网络打印机，因此是配置不当使它不停产生该类响应，从而引发 IDS 产生大量的“ATTACK-RESPONSES 403 Forbidden”警报。对于该类模式警报，管理员可以通过对打印机进行合理配置或者直接在 IDS 上将该模式的警报过滤掉来解决。对于表（5）中 2 号模式，它对应的警报产生规则是，当机器对某个主机的 8080 端口扫描就产生警报，但经过分析发现主机 202.xxx.x.10 上 8080 端口不存在，可见也可以滤掉该类警报。

同样可以对其它的模式进行分析，找出产生频繁警报的原因，然后建立过滤规则或对 IDS 的规则进行改进或对网络配置进行重新设定，在这里就不再一一列举。图（2）列出了 14 天的警报数据，其中前 7 天是分析处理前的警报数据，后 7 天是通过本文方法分析处理后的警报数据，其中 X 轴为时间（单位：日/月），Y 轴为警报个数（单位：条），从图可以看出通过本文方法可以大大减少警报数目，从而降低误报率。

图（2）：分析处理前后警报数据总数的比较



从实验结果来看，本文提供的模式挖掘算法能很方便地为管理员提供警报中的频繁模式，能有效地减少需要用户分析的警报数目。这是利用简单的统计方法难以办到的，因为简单的统计方法很难自动发现警报中的规律，除非有较多的人工指导参与。

5. 结论

目前，国内外的入侵检测系统都普遍存在报告的警报数目过多，且警报中包含大量的误报的问题，这给安全管理员的分析与处理带来很大困难，从而迫切需要自动或半自动分析警报数据的技术来辅助管理员来分析与处理警报。本文利用数据挖掘技术来对警报进行分析处理，挖掘出警报中的频繁模式，并由管理员对挖掘的模式进行进一步的分析与处理。试验结果表明该方法能有效帮助管理员对警报的分析处理，可大大减少警报数目，降低误报率。此外，从本文的方法实现上看，本文的方法不需要像[7]中的方法那样定义网络层次图，实现方便；另外，由于本文采取了半自动交互式的警报分析处理方法，这在误报消除的准确性要上高于[3][5]文献的方法。

参考文献

1. 龚俭，陆晟，王倩 计算机网络安全导论[M]，东南大学出版社, 2000
2. D E Denning. An intrusion detection model. IEEE Trans on Software Engineering, 1987, 13 (2):222-232
3. S. Manganaris, M. Christensen, D. Zerkle, and K. Hermiz. A Data Mining Analysis of RTID Alarms, Computer Networks, 34(4), October 2000.
4. E. Bloedorn, B. Hill, A. Christiansen, C. Skorupka, L. Talboot, and J. Tivel. Data Mining for Improving Intrusion Detection, http://www.mitre.org/support/papers/tech_papers99_00/, 2000.
5. C. Clifton, G. Gengo. Developing Custom Intrusion Detection Filters Using Data Mining. In Military Communications Int'l Symposium (MILCOM2000), October 2000.
6. K. Julisch. Mining Alarm Clusters to Improve Alarm Handling Efficiency. In 17th Annual Computer Security Applications Conference (ACSAC), pages 12–21, December 2001.
7. K. Julisch. Clustering Intrusion Detection Alarms to Support Root Cause Analysis, ACM Transactions on Information and System Security 6(4), November 2003.
8. Axelsson, S. The Base-Rate Fallacy and the Difficulty of Intrusion Detection. ACM Transactions on Information and System Security (TISSEC) 3(3), 186-205 2000.
9. K. Julisch, Marc Dacier. Mining intrusion detection alarms for actionable knowledge The 8th ACM Int'l Conf on Knowledge Discovery and Data Mining , Edmonton, 2002
10. Jiawei Han, Micheline Kamber 著，范明,孟小峰等译，数据挖掘概念与技术[M],机械工业出版社，2001.8.
11. Marcus J. Ranum, False Positives: A User's Guide to Making Sense of IDS Alarms. ICSA Labs IDSC February 2003