一个基于数据挖掘的入侵检测系统模型*)

杨 莘 刘 恒 吕述望
(中科院研究生院信息安全国家重点实验室 北京100039)

A Data Mining Based Intrusion Detection System Model

YANG Xin LIU Heng LU Shu-Wang
(State Key Laboratory of Information Security, The Graduate School of Chinese Academy of Sciences, Beijing 100039)

Abstract Applying data mining technique to intrusion detection and building a relevant model is the hotpot of study currently. This paper presents a typical data mining based IDS model, including data gathering and selection, data mining algorithm compare, system elements and model structure.

Keywords Data mining, Intrusion detection

1. 概述

入侵检测实质上归结为对安全审计数据的处理。按分析引擎所使用的检测方法可以将入侵检测系统分为误用(基于知识)检测和异常(基于行为)检测。前者运用已知攻击方法,根据已定义好的入侵模式,通过判断这些入侵模式是否出现来进行检测。为了克服误用检测的缺陷,人们提出了针对入侵行为的异常检测模型,指根据使用者的行为或资源使用状况的正常程度来判断是否入侵,而不依赖于具体行为是否出现来检测,目前处于研究阶段。

安全事件审计系统作为保障信息系统安全的基础部件,已越来越多地被应用到操作系统和网络安全管理工具中。然而,操作系统的日益复杂化和网络数据流量的急剧膨胀,导致了安全审计数据以惊人的速度递增。激增的数据背后隐藏着许多重要的信息,人们希望能够对其进行更高抽象层次的分析,以便更好地利用这些数据。目前的审计系统可以高效地实现安全审计数据的输入、查询、统计等功能,但无法发现数据中存在的关联、关系和规则,无法根据现有的数据预测未来的发展趋势,缺乏挖掘数据背后隐藏的知识的手段,导致了"数据爆炸但知识贫乏"的现象。如何在大量的审计数据中提取出具有代表性的系统特征模式,用于对程序或用户行为作出描述,是实现安全事件审计系统的关键。

数据挖掘(Data Mining)本身是一项通用的知识发现技术,其目的是要从海量数据中提取出我们所感兴趣的数据信息(知识)。

目前对挖掘算法的研究已经比较成熟,有许多算法可以使用。然而,真正要从海量数据中提取出我们所感兴趣的数据信息(知识),需要强调的一点就是"特定应用"。算法实现必须建立在特定应用的基础之上,并且需要具有足够的先验知识,实验表明:我们对系统安全的先验知识往往体现在对原始数据中有价值的变量集的选择上。与其它分析方法不同的是:我们将入侵检测看成是一种数据分析过程,着眼于对海量的安全审计数据应用数据挖掘算法,以一种自动和系统的手段建立一套自适应的、具备良好扩展性的入侵检测系统。

2. 审计数据源的选择

采用数据挖掘算法进行数据处理,要求我们所选取的安全审计数据具备以下特性:①攻击事件相对于正常的网络或系统访问是很少见的;②安全审计数据在正常情况下是非常稳定的;③攻击总是使安全审计数据的某些特征变量明显地偏离正常值。

网络数据源和主机数据源各有优势,又各有不足。值得注意的是,两者在各自所擅长的检测领域上存在互补性。因此,最佳的安全审计数据处理方案应该是综合这两方面的审计数据源,最大限度地提高对网络及主机系统的信息收集,为实现准确、高效的入侵检测提供保障。

3. 数据挖掘算法简介

数据挖掘是针对特定应用的数据分析处理过程。如何选择输入数据、变换数据及对应的挖掘算法,取决于具体的数据挖掘目标,即期望从数据中发掘出何种类型的"知识"。按照挖掘目标的不同,数据挖掘可分为以下类型:关联分析(Mining Association Rules);数据总结(Data Generalization);数据分类(Data Classification);聚类分析(Clustering Analysis);序列模式分析(Mining Sequential Patterns)。相应地,数据挖掘的算法也可以以此分类:关联分析算法(如 Apriori, AprioriTid);数据总结算法(如 Data Cube, OLAP);数据分类算法(如 C4.5, RIPPER);聚类分析算法(如 CLARANS, BIRCH);序列分析算法(如 AprioriAll, AprioriSome, DynamicSome)。

对于应用到入侵检测系统中的数据挖掘算法,目前主要集中在关联、序列和分类这三种类型上。

3.1 关联分析算法

考虑一些涉及许多项目(Items)的事务:事务1中出现了项目A,事务2中出现了项目B,事务3中则同时出现了项目A和B。那么,项目A和B在事务中的出现之间是否有规律可循呢?在数据库的知识发现中,关联规则就是描述这种在一个事务中项目之间同时出现的规律的知识模式。更确切地说,关联规则通过量化的数字描述项目A的出现对项目B的出现有多大的影响。

设 $R=\{I_1, I_2, \cdots, I_m\}$ 是一组数据项集,W 是一组事务集。W 中的每个事务 T 是一组数据项,且满足 $T \subseteq R$。假设有一个数据项集(Itemset)X,一个事务 T,如果 $X \subseteq T$,则称事务 T 支持数据项集 X。

我们所要发掘的关联规则是指如下形式的一种数据隐含

*)国家863资助项目(2001AA140211)。杨 莘 硕士生,研究方向为计算机网络安全。刘 恒 博士生。吕述望 研究员,博士生导师。

规则：

$X \Rightarrow Y$，其中 X，Y 是两组数据项，$X \subset T$，$Y \subset T$，$X \cap Y = \phi$

一般用四个参数来描述一个关联规则的属性：置信度(confidence)、支持度(support)、期望置信度(Expected confidence)和作用度(Lift)。置信度是对关联规则准确度的衡量，支持度则是对关联规则重要性的衡量。支持度说明了这条规则在所有事务中有多大的代表性，显然支持度越大，关联规则越重要。期望置信度描述了在没有数据项集 X 的作用下，数据项集 Y 本身的支持度；作用度描述了数据项集 X 对数据项集 Y 的影响力。作用度越大，说明数据项集 Y 受数据项集 X 的影响越大。

发掘关联规则通常可分为以下两个步骤进行：

第一步，从事务集 W 中找出所有支持度大于最小支持度的数据项集，称之为大数据项集(large itemsets)，其它不满足支持度要求的数据项集则称为小数据项集(small itemsets)。

第二步，使用大数据项集产生期望的关联规则。产生关联规则的基本原则是其置信度必须大于预先指定的门限值。

3.2 序列分析算法

简单来说，关联分析是发掘数据记录中不同数据项之间的关联性，而序列分析则是发现不同数据记录之间的相关性。序列分析的目标是在事务数据库中发掘出序列模式(large sequences)，即满足用户指定的最小支持度要求的大序列，并且该序列模式必须是最高序列(maximal sequence)。

挖掘序列模式通常分为以下五个步骤进行，其中序列阶段是序列分析的关键所在。

1)排序阶段：以事务的主体为主键，事务时间为次键，对原始数据库进行排序，将其转换为主体序列(customer sequences)的数据库；

2)大数据项阶段：找出所有的大数据项集 L(此过程也相当于找出了所有长度为1的大序列)，并把大数据项集映像为一组相邻的整数，每个大数据项对应一个整数；

3)转换阶段：将数据库中主体序列的每一次事务用该事务包含的大数据项集 litemsets(映像的整数)代替；

4)序列阶段：利用大数据项集发掘序列模式(large sequences)；

5)序列最高化阶段：找出所有序列模式(large sequences)的最高序列集。

3.3 分类算法

数据分类的目的是提取数据库中数据项的特征属性，生成分类模型，该模型可以把数据库中的数据项映像到给定类别中的一个。数据分类的处理步骤如下：

·获得训练数据集(training set)，该数据集中的数据记录具有和目标数据库中数据记录相同的数据项。

·训练数据集中每一条数据记录都有已知的类型标识与之相关联。

·分析训练数据集，提取数据记录的特征属性，为每一种类型生成精确的描述模型。

·使用得到的类型描述模型对目标数据库中的数据记录进行分类或生成优化的分类模型(分类规则)。

3.4 算法的扩展

上面所述的都是通用算法，这些算法没有考虑任何领域知识，所以很自然的，算法运行会产生大量无意义的结果，下面以分类算法 Ripper 为例讨论这个问题。例如：下面是一些经过预处理的 tcp 会话记录，其中包含了若干次端口扫描，一次正常 telnet 会话和一次正常 www 会话。

```
983156875.144179 0.00029707 priv-436 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.144363 0.000303984 priv-225 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.144580 0.000285983 other-32772 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.144768 0.00029397 priv-247 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.144961 0.000295043 priv-948 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.145155 0.00029707 priv-349 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.145360 0.000288963 priv-823 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.145551 0.000295997 priv-654 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.145743 0.000298023 priv-885 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.146124 0.000264049 priv-319 ? ? 192.168.0.51 192.168.0.53 REJ L
983156875.148734 0.000277042 other-cmip-agent ? ? 192.168.0.51 192.168.0.53 REJ L
983175301.103723 1.5439 other-www 5 224 192.168.0.51 192.168.0.1 SF L
983175305.905590 26.4298 other-telnet 170 2298 192.168.0.51 192.168.0.64 SF L
```

由 Ripper 算法产生的规则是：

```
normal 1 0 IF timestamp～983175301.103723.
normal 1 0 IF timestamp～983175305.905590.
abnormal 1520 0 IF.
```

也就是说，Ripper 算法根据时间戳来判断出 normal 或 abnormal，这显然是没有意义的，时间戳跟安全事件没有任何直接的关系。因此我们有必要扩展基本算法，使算法的结果合理化。在基于规则的挖掘算法中，控制挖掘过程的两个门限值是支持度(support)和置信度(confidence)，现在我们希望根据先验知识来控制挖掘过程，所以有必要引入第三个控制变量。

在原始数据的每一条记录中有许多变量，这些变量具有不同的重要程度。例如：在 IP 报头和 TCP 报头的层次上，一次 TCP 连接可以用下面的四元组来标识：(timestamp，src_host，src_port，dst_host，dst_port)，其中 dst_port 直接与这次连接对应的服务相关，但是对入侵检测系统来说，timestamp 相对来说并不重要，src_host 和 dst_host 跟攻击方法本身没有关系，所以最重要的特征变量就是 dst_port，另外还要加上会话结束时 tcp 连接所处的状态，在下面的预处理中，我们将会话结束时 tcp 连接所处的状态分为14种，例如大量的会话结束时处在状态0，即三次握手只进行到第一步，如下所示，这时就有很大的把握判断这是一次伪造 IP 地址对 192.168.0.1这台主机进行的 SYN flood 攻击。

```
982041531.713226 ? other-www ? ? 192.168.0.1 159.210.38.200 S0
982041531.713423 ? other-www ? ? 192.168.0.1 51.173.60.13 S0
982041531.713620 ? other-www ? ? 192.168.0.1 217.168.50.75 S0
982041531.713856 ? other-www ? ? 192.168.0.1 152.17.125.254 S0
982041531.714055 ? other-www ? ? 192.168.0.1 29.245.65.41 S0
982041531.714253 ? other-www ? ? 192.168.0.1 174.191.246.33 S0
982041531.714451 ? other-www ? ? 192.168.0.1 215.191.58.253 S0
982041531.715361 ? other-www ? ? 192.168.0.1 187.49.53.247 S0
982041531.715560 ? other-www ? ? 192.168.0.1 115.208.229.165 S0
982041531.715756 ? other-www ? ? 192.168.0.1 235.62.45.105 S0
982041531.715953 ? other-www ? ? 192.168.0.1 64.133.50.128 S0
982041531.716150 ? other-www ? ? 192.168.0.1 22.47.109.13 S0
```

在基本算法中，通过支持度(support)和置信度(confidence)来控制数据挖掘过程，这个过程可以表示为：

```
if p is a pattern and f(support(p),confidence(p))> defined_value then
generate_rules(p);
end if
```

我们可以添加一个函数 validspattern(p)表示模式 p 包含了一组预先定义的变量，这些变量我们认为是重要的，任何一条产生的规则都必需包含这组变量，这样上面的过程就变为：

```
if p is a pattern and f1(validpattern(p), f(support(p),
  confidence(p))> defined_value then
generate_rules(p);
end if
```

根据变量集中各个变量的重要程度可以把它们分为两大类：主因子变量和参考变量：

·主因子变量：必须包括在生成规则集中的变量，这些变量在识别攻击事件中起决定性的作用，是数据挖掘的主要对象。理论和实验都证明：通过合适的选取主因子变量，可以显著地降低算法的时间和空间复杂度。

·参考变量：在原始数据包含的变量集合中，有一些变量可以作为其它变量的参考变量(reference attributes)，通常参考变量描述事件主体，而其它变量描述主体的行为，在某些情况下，只有通过对同一主体的不同行为的分析，得到的结果才是有意义的。在这些情况下，基本算法修改为：首先根据主因子变量生成规则集合，对每一条规则，如果该规则覆盖的实例的参考变量一致，则保留该规则，否则删除这条规则。

上面的算法还存在一个很大的缺陷，我们是通过控制支持度和置信度来控制算法执行过程的，如果把支持度定得较高，就会漏掉一些出现频率很低的网络应用，这样我们得到的模式集就不完整，如果把支持度定得较低，就会对一些高频率的应用产生大量的模式。解决这个问题有两个办法。

第一种方法将算法修正为：当算法产生一条新的规则的时候，其中必须至少包含一个新的主因子变量；

第二种方法是将支持度修改为相对支持度，所谓相对支持度是针对某一变量而言的，如果定义 S_i 是变量 A_i 的相对支持度，并且变量值对 $A_i=V_{ij}$ 在整个记录集中一共出现 N_{ij} 次，那么如果包含 $A_i=V_{ij}$ 的规则覆盖实例的个数超过 $S_i * N_{ij}$，那么我们认为这条规则满足支持度约束。通过这两种办法都可以既避免对出现频率高的变量产生大量的无意义的规则，又可以包含一些出现频率很低但是比较重要的变量。

4. 系统原理与模块结构

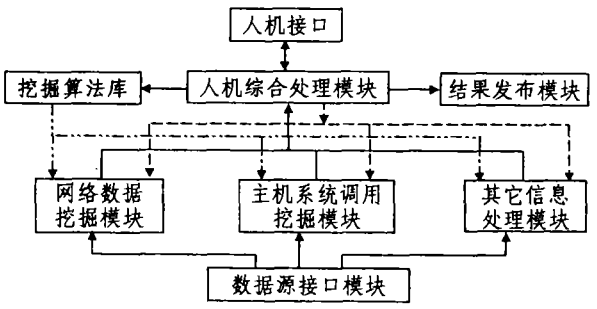

图1 数据挖掘系统模块结构图

这里要介绍的安全审计数据分析挖掘系统将针对典型的网络攻击和异常行为来进行系统设计，本系统综合采用误用检测和异常检测两种入侵检测引擎，引入对海量的安全审计数据进行处理的数据挖掘算法，从网络数据和主机系统调用数据两方面分别进行相应的检测模型生成及测试工作。图1是系统的模块结构图。

4.1 网络数据挖掘模块

对网络攻击的检测方法大概可以分为两大类，第一类是可以通过对网络层和传输层报头的检测就能识别，比如拒绝服务攻击(DoS)和端口扫描，它们共同的特点是在一段比较短的时间内发送出大量的数据包，从而在网络中造成很有规律的流量；第二类是只有通过对数据包的内容进行检测才能识别的攻击方法，比如缓冲区溢出攻击和口令猜测，这一类检测还往往涉及到高层协议解析，也就是说必须具备相应的领域知识。目前，我们所建立的基于网络数据的检测模型分为两种类型：基于连接(会话)记录的误用检测和基于用户行为的异常检测。

1. *基于连接(会话)记录的误用检测* 首先将网络中传输的数据包还原成基于传输层的连接记录，从中提取出可以用于对传输层连接记录进行分类的特征属性。对于在传输层无法判断的连接记录，则进行高层的协议解析，分解为相应的FTP、Telnet、HTTP 会话，针对每一种高层协议，提取出可以用于判断的特征属性。针对各种已知的攻击手段，利用数据挖掘分类算法，通过对包含特定攻击手段的训练数据的机器学习，挖掘出对应的分类规则，用于对实际网络中的连接(会话)记录进行分类。

2. *基于用户行为的异常检测* 异常检测的关键问题在于正常使用模式的建立以及如何利用该模式对当前用户行为进行比较和判断。行为模式是指程序执行或用户操作过程中体现出的某种规律性。大量的实践经验表明，无论是程序的执行还是用户的行为，在系统特性上都呈现出紧密的相关性，这些带有强一致性的行为特征正是我们希望挖掘出的正常使用模式的组成部分。

为了对用户行为进行异常检测，应用数据挖掘中的关联分析和序列挖掘，提取出正常情况下用户所执行命令中存在的相关性，建立每个用户的历史行为模式，为实际检测过程中用户行为的判别提供比较的依据。图2是系统的模块示意图。

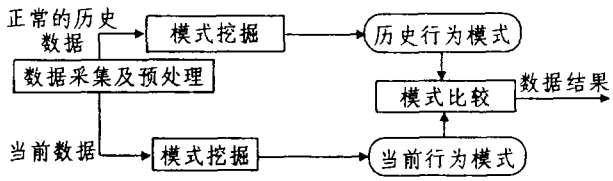

图2 用户行为异常检测模块示意图

网络系统中，主机系统通常具备高速的计算能力、大容量的存储设备、高性能的系统吞吐率、庞大的系统数据库以及其它软硬件资源。为了节省开支和充分利用主机的资源，通常由主机向外提供 Telnet 服务，为用户访问系统资源提供便捷的方式。用户向 Telnet 主机提交的命令和主机返回的结果通过获取网络中传输的数据包并进行相应的协议解析，就可以完全复原。我们把用户提交的每一条 shell 命令并结合与之相关的其它属性作为一条审计记录。每一条审计记录包含以下数据项：

审计记录格式

用户名	时间戳	主机 IP	用户 IP	命令	命令参数
Username	Timestamp	HostIP	UserIP	Command	Param

这样做的目的就是要从复原的用户命令审计记录中分析出每个用户的行为特征，并基于这种特征为每个用户建立正常的行为模式。目前所提取的用户行为特征包括关联和序列两方面内容。所谓关联，是指同一条审计记录中不同字段之间存在的关联关系。例如通过对用户历史数据的分析，可以发现如下的关联规则：

1) $Username = A \sim\sim Timestamp = am \sim\sim HostIP =$ 192.168.0.1$\sim\sim UserIP=$192.168.0.10(0.98,0.60)

2) $Username=A\sim\sim Command=vi\sim\sim Param=.c$(0.45,0.05)

规则1)表示用户 A 通常在每天的上午登录，登录的主机是192.168.0.1，登录时的 IP 地址是192.168.0.10。规则2)表示用户 A 经常执行 vi 命令，执行该命令时所使用的参数通常是.c 为后缀名的文件。

关联规则的关键属性包括支持度(Support)和置信度(Confidence)。对于关联规则 R，我们采用的支持度和置信度的计算公式为：

(1)支持度：$Support(R)=N_R/N$

(2)置信度：$Confidence(R)=N_R/N_{Username=A}$

其中 N 代表审计记录集所包含的记录总数，N_R 表示满足规则 R 的记录数，$N_{Username=A}$则代表用户名字段值为 A 的记录数。上述关联规则1)的置信度是98%，支持度是60%，关联规则2)的置信度是45%，支持度是5%。

序列模式的挖掘是希望发掘出不同审计记录之间的相关性。例如通过对用户 A 所执行的命令序列进行分析，我们发现如下的序列模式：

$Command=vi, Param=.c\rightarrow Command=gcc, Param=-g-o\rightarrow Command=gdb$(0.4)

表示用户 A 经常执行的命令序列是：首先用 vi 编辑 c 程序，然后用 gcc 编译，再使用 gdb 进行程序的调试。

序列模式的关键属性是模式的支持度。对于序列模式 P，假设其长度为 l，则支持度计算公式为：$Support(P)=N_p/N_l$，其中 N_p 代表序列模式 P 在审计记录集中出现的次数，N_l 表示审计记录集所包含的所有长度为 l 的序列数目。上述序列模式的支持度是40%。

依照以上挖掘出的关联规则和序列模式，我们可以判断出用户 A 应该是一个 c 程序员，其工作时间是每天的上午，并且通常从 IP 为192.168.0.10的客户机登录到 IP 为192.168.0.1的主机上进行编程操作。如果在实际的检测过程中，发现某一天该用户突然在晚间登录，或者从一个陌生的 IP 地址登录到系统主机，或者在登录过程中执行了大量与编程无关的操作，访问主机的敏感目录和文件，则可以推断出该用户出现了某种异常。这种异常可能是该用户正在试图超越其正常的操作权限，也可能是有人冒用该用户的账号进行恶意的操作。

4.2 主机数据挖掘模块

针对主机系统调用的数据挖掘主要从以下方面入手：

1)*系统调用的关联分析* 从关联分析的角度看，与系统调用相关的各个变量具有很强的相关性。如果将一次系统调用表示为(进程号 系统调用号 用户 ID 访问对象 访问权限)，不难发现，这五个变量都具有很强的关联关系，特别适合利用数据挖掘算法进行分析。

2)*系统调用的序列分析* 任何系统进程归根到底都是一段程序，如果一段程序完全没有 if-then-else 这类选择语句和 while 等循环语句，那么这段程序所产生的系统调用一定是完全固定的，如果该进程受到攻击，必然会打乱正常的系统调用，从而可以通过对系统调用序列的分析检测出来。当我们把选择语句和循环语句考虑进来，我们发现，虽然整个程序产生的系统调用具有一定的随机性，但是如果将系统调用序列划分为一定窗口大小的子序列，这些子序列仍然具有相当地稳定性。对由 sendmail 产生的近700万次系统调用进行分析，以窗口为6进行划分，发现当分析到10万次系统调用以后，长度为6的子序列就稳定在1000左右，此后的增加是非常缓慢的。因此，对系统调用进行序列分析，可以得到较好的结果。

4.3 其它模块

1)*挖掘算法库* 存放各种数据挖掘算法(包括关联分析、序列模式分析、数据分类)以及其它用于数据预处理、模式匹配的算法，并提供实时的更新功能。

2)*其它信息处理模块* 通过对其他安全信息的分析处理，为检测模型的生成提供参考。

3)*人机综合处理模块* 控制各个挖掘模块的运行及调整，对挖掘模块的分析结果作综合处理，并负责结果发布和挖掘算法库的更新。

4)*人机接口模块* 提供系统的人机接口，保证数据挖掘的全过程，包括数据预处理、映像、关联分析、序列分析、分类模型生成等步骤的可控性。

5)*结果发布模块* 发布数据挖掘的结果，包括关联规则、序列模式、分类模型等。

6)*数据源接口模块* 提供对网络数据源和主机系统调用数据源的访问接口。

参 考 文 献

1 Bace R G. Intrusion Detection. Macmillan Technical Publishing, U.S.A., 1999

2 Lee W. A Data Mining Framework for Constructing Features and Models for Intrusion Detection Systems: [PhD thesis]. Columbia University, 1999

3 Lee W, Stolfo S J, Mok K W. A data mining framework for building intrusion detection models. In: Proc. of the 1999 IEEE Symposium on Security and Privacy, May 1999

4 Chen M, Han J, Yu P. Data mining: An overview from database perspective. IEEE Transactions on Knowledge and Data Eng., 1996, 8(6): 866～883

5 连一峰，戴英侠，王航. 基于模式挖掘的用户行为异常检测