(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 104615542 A
(43)申请公布日 2015.05.13

(21)申请号 201510072607.5

(22)申请日 2015.02.11

(71)申请人 中国科学院软件研究所
地址 100190 北京市海淀区中关村南四街 4 号

(72)发明人 吴晓慧 马恒太 刘小霞 邱春光

(74)专利代理机构 北京科迪生专利代理有限责任公司 11251
代理人 成金玉 孟卜娟

(51)Int.Cl.
G06F 11/36(2006.01)

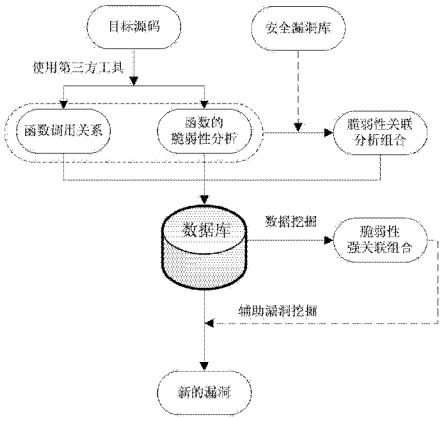
权利要求书2页 说明书6页 附图2页

(54)发明名称

一种基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法

(57)摘要

本发明提供一种基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，涉及安全检测技术领域，包括以下步骤：1、使用工具对目标源码提取函数调用关系，并用静态分析工具对每个函数进行脆弱性分析，形成建立脆弱性信息数据库；2、对公告中与该目标有关的最新漏洞信息，提取其触发点的脆弱性信息，基于函数调用关系形成脆弱性关联分析组合，并自动去重；3、根据漏洞信息数据库中的记录信息，对脆弱性关联分析组合进行数据挖掘与知识发现，提取脆弱性之间的强相关组合；4、根据脆弱性强相关组合，分析脆弱性信息数据库，发现新的可能存在的漏洞。该发明通过数据关联分析找出已公开漏洞隐含的脆弱性关联关系，辅助漏洞挖掘，为批量发现漏洞提供了可行方法。



CN 104615542 A

1. 一种基于函数调用的脆弱性关联分析辅助漏洞挖掘方法，其特征在于实现步骤如下：

步骤 1、使用工具分析目标对象的源码，提取静态函数调用关系；并用静态分析工具对每个函数进行脆弱性分析，形成脆弱性信息数据库；

步骤 2、将以 CVE（即公共漏洞和暴露）为代表的安全漏洞库中与目标对象有关的用户所需的漏洞信息下载，并与脆弱性信息数据库进行操作，形成最原始的漏洞信息数据库。

步骤 3、根据漏洞信息数据库中的记录信息，按类型分类，对不同的漏洞信息分别进行数据挖掘与知识发现，提取漏洞隐含的基于函数调用关系的脆弱性关联分析组合，形成列表并自动去重；

步骤 4、根据脆弱性关联分析组合，分析脆弱性信息数据库，发现新的潜在的漏洞。

2. 根据权利要求 1 所述基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，其特征在于：所述的步骤 1 中对目标对象的源码提取函数调用关系的具体实现为：使用开源工具，给 GCC（即 GNU 编译器套件）打个补丁，让 GCC 在编译每个源文件时复制出其中函数的调用关系，然后用 Perl 脚本收集并整理调用关系，输出为函数调用关系表存到脆弱性信息数据库。

3. 根据权利要求 1 所述基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，其特征在于：所述的步骤 1 中对每个函数进行脆弱性分析的具体实现为：使用第三方静态分析工具对源码进行扫描，得到脆弱性分析结果；由于这些结果往往是定位到每个文件的行数，为方便关联分析，通过分析源码获取每个函数的起始行数和结束行数，进而将脆弱性分析结果定位到函数；经过上述处理，得到每个函数里面包含有哪些脆弱性，将该结果作为函数级的脆弱性信息表存放至脆弱性信息数据库中。

4. 根据权利要求 1 所述基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，其特征在于：所述的步骤 2 提取脆弱性关联分析组合的具体实现如下：

A. 定义与漏洞信息相关的搜索字段规则，包括字段名称、类型、前标识符、后标识符等关键信息；

B. 对以 CVE 为代表的安全漏洞库进行自动搜集处理即网络爬虫挖掘方法，将与目标对象有关的用户所需的漏洞信息下载，输出为漏洞表形成最原始的漏洞信息数据库；

C. 通过数据库接口中间件，对漏洞信息数据库中的漏洞表和脆弱性信息数据库中的函数调用关系表及函数级的脆弱性信息表进行各种查询和匹配，形成以漏洞触发点为参照点的脆弱性关联分析组合。

5. 根据权利要求 1 所述基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，其特征在于：所述的步骤 3 提取脆弱性之间的强相关组合的具体实现如下：

(1) 根据漏洞信息数据库中的记录信息，依据漏洞类型，对漏洞脆弱性关联分析组合进行分类，形成列表并自动去重；

(2) 运用关联规则挖掘算法，对同一漏洞类型的脆弱性关联分析组合进行关联规则挖掘，形成频繁项集，进而运用挖掘算法产生该漏洞类型对应的强相关组合；

(3) 将得到的脆弱性强相关组合存入到漏洞信息数据库中。

6. 根据权利要求 1 所述基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，其特征在于：所述的步骤 4 根据脆弱性强相关组合，辅助漏洞挖掘的具体实现为：根据漏洞信息数

据库中脆弱性强相关组合，分析脆弱性信息数据库，除已公开漏洞触发点外，如果该目标对象的源码中存在漏洞信息数据库中的脆弱性强相关组合，则认为该脆弱性组合可能是潜在的漏洞，将相关数据作为潜在漏洞表存入漏洞信息数据库；该数据库为批量漏洞挖掘提供了可信度较高的数据。

一种基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法

技术领域

[0001] 本发明涉及一种基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法，属于漏洞挖掘领域。首先利用静态分析工具得到大量的脆弱性分析数据，以函数调用关系为基础提取已公开漏洞隐含的脆弱性相关关系，并进行数据挖掘获得强相关分析组合，以此辅助漏洞挖掘，从而对可能的漏洞进行预警和防范。

背景技术

[0002] 据国家计算机病毒应急处理中心调查分析，“未修补网络（系统）安全漏洞”是网络安全事件发生的最主要原因。这些漏洞不仅是蠕虫、病毒等恶意代码的重要传播途径，也是网络攻防的焦点。保障网络系统安全的一个重要前提就是，快速挖掘并修复系统中的安全漏洞，消除系统的安全隐患。系统漏洞的数量在不断增加，其中隐含了一定的必然规律和规则，通过数据挖掘等手段对漏洞信息库中已公开漏洞隐含的有效信息进行处理，提取内在关联关系，可以有效辅助漏洞挖掘。

[0003] 程序静态分析是指在不运行代码的方式下，通过词法分析、语法分析、控制流分析等技术对程序代码进行扫描，找到匹配某种规则模式的代码从而发现代码中存在的问题。目前成熟的代码静态分析工具每秒可扫描上万行代码，相对于动态分析，具有检测速度快、效率高的特点。随着静态分析工具的扫描规则库的完善，静态分析发现的代码隐含错误，为进一步提取代码缺陷信息提供了一定可信度的数据。

[0004] 软件中隐含的缺陷数目与可靠性直接相关。一个漏洞的产生，可能是由一连串脆弱性组合累积导致的，而这个脆弱性组合的形成就是基于函数调用关系。函数调用关系体现了函数调用的先后次序，又包含了函数调用过程的重要信息，包括函数调用中传递的参数、隐含传递的参数以及返回值等。基于函数调用的脆弱性关联分析辅助漏洞挖掘，能够很好地与数据挖掘技术相结合，提高漏洞挖掘的自动化水平，有较强的现实意义。

发明内容

[0005] 本发明技术解决问题：随着当前系统软件的爆炸式增长和安全问题的凸显，为了快速有效地进行漏洞挖掘，本发明充分利用了当前成熟的第三方静态分析工具得到的脆弱性分析数据，并提供了有效的方法将其转化为可能的漏洞。

[0006] 本发明的技术解决方案：通过对目标对象的源码进行静态分析，获得其脆弱性分布点及函数调用关系并建立脆弱性信息数据库；对漏洞信息库中已公开漏洞的有效信息进行处理，提取其隐含的基于函数调用的脆弱性关联关系；利用数据挖掘提取同一类型的脆弱性强相关组合，并将其用于辅助快速有效地漏洞挖掘。

[0007] 本发明具体实现步骤如下：

[0008] 步骤 1、使用工具分析目标对象的源码，提取静态函数调用关系；并用静态分析工具对每个函数进行脆弱性分析，形成脆弱性信息数据库；

[0009] 步骤 2、将以 CVE（公共漏洞和暴露）为代表的安全漏洞库中与目标对象有关的

用户所需的漏洞信息下载，并与脆弱性信息数据库进行操作，形成最原始的漏洞信息数据库；

[0010] 步骤 3、根据漏洞信息数据库中的记录信息，按类型分类，对不同的漏洞信息分别进行数据挖掘与知识发现，提取漏洞隐含的基于函数调用关系的脆弱性关联分析组合，形成列表并自动去重；

[0011] 步骤 4、根据脆弱性关联分析组合，分析脆弱性信息数据库，发现新的潜在的漏洞。

[0012] 所述步骤 1 中对目标对象的源码提取函数调用关系的具体实现为：使用开源工具，给 GCC(GNU 编译器套件）打个补丁，让 GCC 在编译每个源文件时复制出其中函数的调用关系，然后用 Perl 脚本收集并整理调用关系，输出为函数调用关系表存到脆弱性信息数据库。

[0013] 所述步骤 1 中对每个函数进行脆弱性分析的具体实现为：使用第三方静态分析工具对源码进行扫描，得到脆弱性分析结果。由于这些结果往往是定位到每个文件的行数，为方便关联分析，通过分析源码获取每个函数的起始行数和结束行数，进而将脆弱性分析结果定位到函数。经过上述处理，可以得到每个函数里面包含有哪些脆弱性，将该结果作为函数级的脆弱性信息表存放至脆弱性信息数据库中。

[0014] 所述步骤 2 所述的提取脆弱性关联分析组合的具体实现如下：

[0015] A. 定义与漏洞信息相关的搜索字段规则，包括字段名称、类型、前标识符、后标识符等关键信息；

[0016] B. 对以 CVE 为代表的安全漏洞库进行自动搜集处理即网络爬虫挖掘方法，将与目标对象有关的用户所需的漏洞信息下载，输出为漏洞表形成最原始的漏洞信息数据库；

[0017] C. 通过数据库接口中间件，对漏洞信息数据库中的漏洞表和脆弱性信息数据库中的函数调用关系表及函数级的脆弱性信息表进行各种查询和匹配，形成以漏洞触发点为参照点的脆弱性关联分析组合。

[0018] 所述步骤 3 所述的提取脆弱性之间的强相关组合的具体实现如下：

[0019] (1) 根据漏洞信息数据库中的记录信息，依据漏洞类型，对漏洞脆弱性关联分析组合进行分类，形成列表并自动去重；

[0020] (2) 运用关联规则挖掘算法，对同一漏洞类型的脆弱性关联分析组合进行关联规则挖掘，形成频繁项集，进而运用挖掘算法产生该漏洞类型对应的强相关组合；

[0021] (3) 将得到的脆弱性强相关组合存入到漏洞信息数据库中。

[0022] 所述步骤 4 所述的根据脆弱性强相关组合，辅助漏洞挖掘的具体实现为：根据漏洞信息数据库中脆弱性强相关组合，分析脆弱性信息数据库，除已公开漏洞触发点外，如果该目标对象的源码中存在漏洞信息数据库中的脆弱性强相关组合，则认为该脆弱性组合可能是潜在的漏洞，将相关数据作为潜在漏洞表存入漏洞信息数据库。该数据库为批量漏洞挖掘提供了可信度较高的数据。

[0023] 本发明与现有技术相比的优点如下：

[0024] (1) 为快速有效地进行漏洞挖掘，本发明充分利用了当前成熟的静态分析工具容易得到的脆弱性分析数据。这些数据可信度尚可，但是缺乏转化为漏洞的有效途径。本发明提供了一种有效的途径将其转化为可能的漏洞，提高漏洞挖掘的自动化水平，有较强的实践意义。

[0025] (2) 本发明以函数调用关系为基础，提取出已公开漏洞隐含的脆弱性关联分析组合，并利用 Apriori 关联规则算法挖掘出不同类型的漏洞对应的脆弱性强关联分析组合，形成匹配模型，为批量漏洞挖掘提供了可能性。

附图说明

[0026] 图 1 为本发明的系统架构图；

[0027] 图 2 为本发明中漏洞隐含的关联关系提取过程；

[0028] 图 3 为本发明的 Apriori 关联规则挖掘算法。

具体实施方式

[0029] 下面结合附图和具体实施方式对本发明作进一步详细的描述。

[0030] 如图 1 所示，本发明一种基于函数调用的脆弱性关联分析辅助漏洞挖掘的方法具体包括以下步骤：

[0031] 步骤 1、使用工具分析目标对象的源码，提取静态函数调用关系；并用静态分析工具对每个函数进行脆弱性分析，形成脆弱性信息数据库；

[0032] 步骤 2、将以 CVE 为代表的安全漏洞库中与目标对象有关的用户所需的漏洞信息下载，并与脆弱性信息数据库进行操作，形成最原始的漏洞信息数据库。

[0033] 步骤 3、根据漏洞信息数据库中的记录信息，按类型分类，对不同的漏洞信息分别进行数据挖掘与知识发现，提取漏洞隐含的基于函数调用关系的脆弱性关联分析组合，形成列表并自动去重；

[0034] 步骤 4、根据脆弱性关联分析组合，分析脆弱性信息数据库，发现新的潜在的漏洞。

[0035] 所述对目标源码提取函数调用关系具体实现为：使用开源工具（如 Codeviz 等），给 GCC 打个补丁，让 GCC 在编译每个源文件时复制出其中函数的调用关系，然后用 Perl 脚本收集并整理调用关系，输出为函数调用关系表存到脆弱性信息数据库。

[0036] 对每个函数进行脆弱性分析的具体实现为：使用第三方静态分析工具对源码进行扫描，得到脆弱性分析结果，这些结果往往是定位到每个文件的行数。根据 CWE 的定义，部分脆弱性标识号和描述如表 1 所示。为方便关联分析，通过分析源码获取每个函数的起始行数和结束行数，进而将脆弱性分析结果定位到函数。经过上述处理，可以得到每个函数里面包含有哪些脆弱性，将该结果作为函数级的脆弱性信息表存放至脆弱性信息数据库中。

[0037] 表 1 常见 CWE 脆弱性描述列表

[0038]

序号	脆弱性名称
CWE-89	SQL 命令（“SQL 注入”）中使用的特殊元素的消毒不当
CWE-78	OS 命令（“OS 命令注入”）中使用的特殊元素消毒不当
CWE-120	缓冲复制不检查输入大小（典型的缓冲区溢出“）
CWE-79	未能保存网页结构（“跨站点脚本”）

CWE-306	重要功能缺乏安全认证
CWE-805	不正确的长度值缓冲区访问
CWE-789	使用硬编码的全权证书
CWE-352	跨站点请求伪造 (CSRF)
CWE-434	无限制上传文件危险类型
CWE-807	依赖非信任的输入在安全的决议

[0039] 所述的提取脆弱性关联分析组合的具体实现如下：

[0040] A. 定义与漏洞信息相关的搜索字段规则，包括字段名称、类型、前标识符、后标识符等关键信息，如表 2 所示；

[0041] B. 对以 CVE 为代表的安全漏洞库进行自动搜集处理即网络爬虫挖掘方法，将与目标对象有关的用户所需的漏洞信息下载，输出为漏洞表，形成最原始的漏洞信息数据库。

[0042] C. 通过数据库接口中间件，对漏洞信息数据库中的漏洞表和脆弱性信息数据库中的函数调用关系表及函数级的脆弱性信息表进行各种查询和匹配，形成以漏洞触发点为参照点的脆弱性关联分析组合。

[0043] 表 2 漏洞信息字段

[0044]

字段名	描述
“CVE 编号”	该漏洞在国际上的统一编号，是 CVE 漏洞库中的唯一编号。
“名称”	该漏洞的中文名称
“漏洞类型”	该漏洞所属于的类型名
“风险级别”	该漏洞的危险系数。
“漏洞描述”	参考各种漏洞公告，对该漏洞的详细描述。
“测试方法”	为了让用户能进一步了解该漏洞，提供了供参考的漏洞测试方法。
“影响系统”	该漏洞所影响的操作系统版本。
“影响软件”	该漏洞所影响的各种软件名称及版本。

[0045] 以影响 Linux kernel 多个版本的漏洞 CVE-2010-3081 为例，漏洞描述为“部分 "compat_alloc_user_space()" 函数的调用者（如 net/compat.c 中的 "compat_mc_getsockopt()" 函数）没有对返回内存区域进行任何校验，本地攻击者可以利用漏洞触发内核内存破坏。”形成脆弱性关联分析组合的过程为：由漏洞描述可知，漏洞的触发可能是由一个语句块触发，也可能是由一个函数调用另外一个函数操作不当引起的。根据漏洞信

息，提取出有缺陷的语句块或函数操作不当的调用关系，根据脆弱性信息数据库，对应到CWE的脆弱性描述条目X1；同时定位到其所在的函数，设为函数A，调用关系为函数B调用函数A，为了说明简洁，此处只选择两级调用关系。函数B包含的脆弱性为Y1，则该脆弱性分析组合为二元组<A(X1),B(Y1)>。此处，组合的顺序表示函数调用关系，该组合为有序组合。

[0046] 对漏洞的脆弱性关联分析组合进行分析，过程如图2所示。首先通过将数据规格化，变为可挖掘的数据源；之后进行数据挖掘，找到数据其中的规律；最后对挖掘结果用可以理解的方式进行解释表达，成为知识。其中，数据挖掘使用统计分析与建模能发现很多任何有用的模式和关系。

[0047] 所述的提取脆弱性之间的关联分析组合的具体实现如下：

[0048] a)根据漏洞信息数据库中的记录信息，依据漏洞类型，对漏洞脆弱性关联分析组合进行分类，形成列表并自动去重；

[0049] b)运用Apriori关联规则挖掘算法，对同一漏洞类型的脆弱性关联分析组合进行关联规则挖掘，形成频繁项集，进而运用挖掘算法产生该漏洞类型对应的强相关组合；

[0050] c)将得到的脆弱性强相关组合存入到漏洞信息数据库中。

[0051] 如图3所示，提供了Apriori关联规则挖掘算法的处理流程，具体包括以下步骤：

[0052] 1)执行初始化操作，主要包括遍历脆弱性关联分析组合数据库D，使用迭代方法对数据集中的所有项集进行扫描，设置最小支持度阀值min_sup。

[0053] 2)筛选出数据集中的所有频繁项集，即将支持度低于阈值的项集全部淘汰掉，而将支持度高于这个阈值的项集认为是频繁项集。这些项集称为频繁集I＝{I1，I2，……Im}，设$A,B\subset I$，则$A\Rightarrow B$是所期望的关联规则。

[0054] 3)指定最小支持度（表示成minsup）和最小置信度（表示成minconf）。首先，将数据集中所有的1阶项集全部找出来，并且根据一个预先设定的支持度阈值来找出这些1阶项集中的频繁项集，并且将频繁项集记为l_1；然后，根据上一步所计算出来的1阶频繁项集计算出2阶候选集c_2，同样通过与最小支持度的比较，得到2阶频繁项集，记为l_2；不断的重复，直到根据l_k-1所生成的候选c_k中的所有项集支持度都小于阈值为止，即不再有更长的频繁项集出现为止。其中：支持度$\mathrm{support}(A\Rightarrow B)=P(A\cup B)$，即A和B这两个项集在数据库D中同时出现的概率；置信度$\mathrm{confidence}(A\Rightarrow B)=P(B|A)=\frac{\mathrm{support}(A\cup B)}{\mathrm{support}(A)}$，即在出现项集A的数据库D中，项集B也同时出现的概率。

[0055] 4)对于每个频繁数据项集A，找到A的所有非空子集，即$B\subseteq A$，B≠Φ；如果confidence(B→(A-B))≥minconf，则生成关联规则B→(A-B))；

[0056] 所述的根据脆弱性强相关组合，辅助漏洞挖掘的具体实现为：根据漏洞信息数据库，分析脆弱性信息数据库。除已公开漏洞触发点外，如果该目标源码存在漏洞信息数据库中的脆弱性强相关组合，则认为该脆弱性组合可能会隐含新的漏洞，并将数据存入至数据库，为批量漏洞挖掘提供了可信度较高的数据。

[0057] 本发明未详细阐述部分属于本领域技术人员的公知技术。

[0058] 以上所述，仅为本发明中的具体实施方式，但本发明的保护范围并不局限于此，任

何熟悉该技术的人在本发明所揭露的技术范围内，可理解想到的变换或替换，都应涵盖在本发明的包含范围之内，因此，本发明的保护范围应该以权利要求书的保护范围为准。

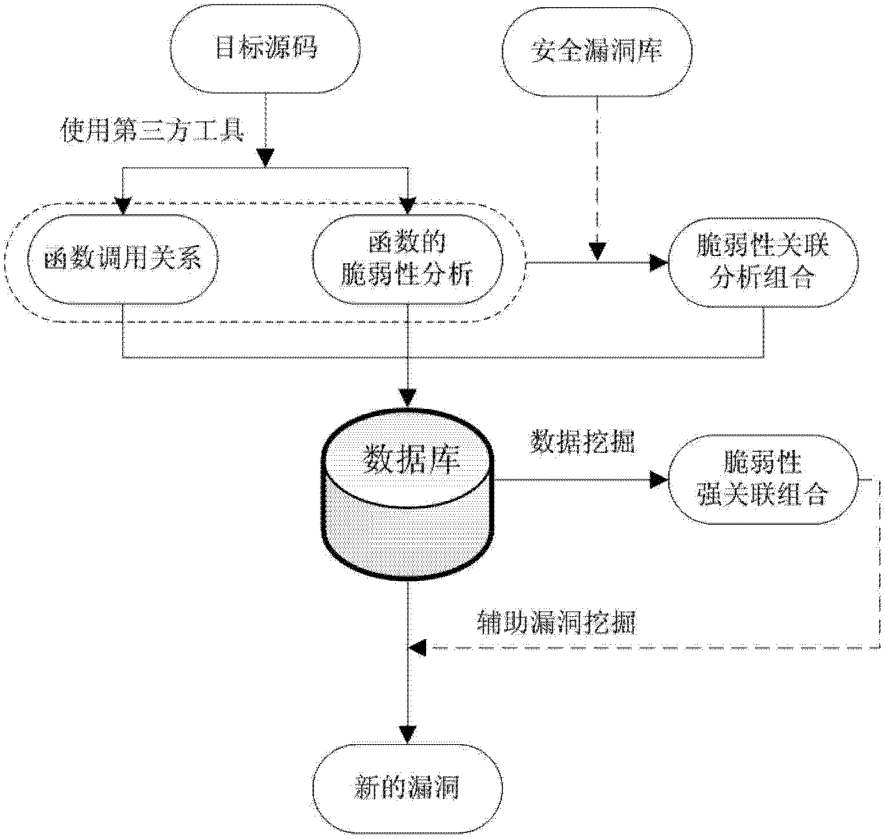

目标源码
安全漏洞库
使用第三方工具
函数调用关系
函数的
脆弱性分析
脆弱性关联
分析组合
数据库
数据挖掘
脆弱性
强关联组合
辅助漏洞挖掘
新的漏洞

图 1

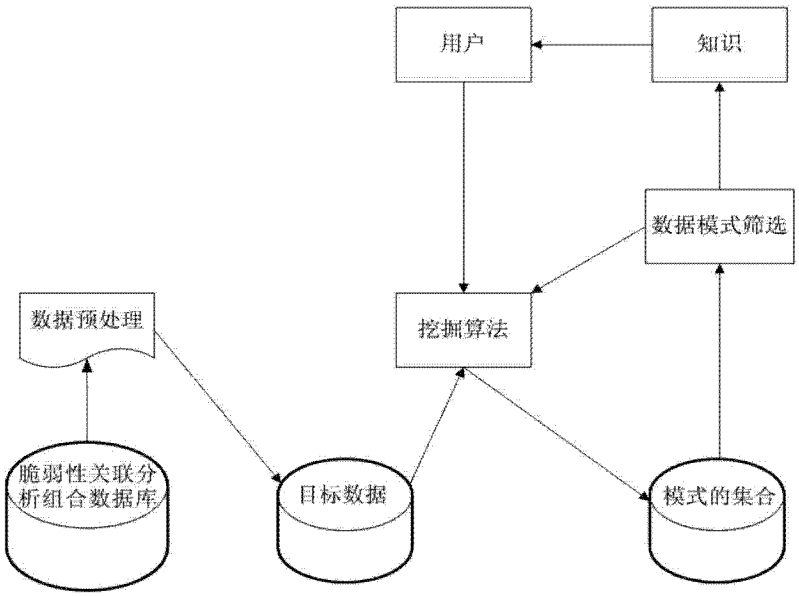

用户
知识
数据模式筛选
数据预处理
挖掘算法
脆弱性关联分
析组合数据库
目标数据
模式的集合

图 2

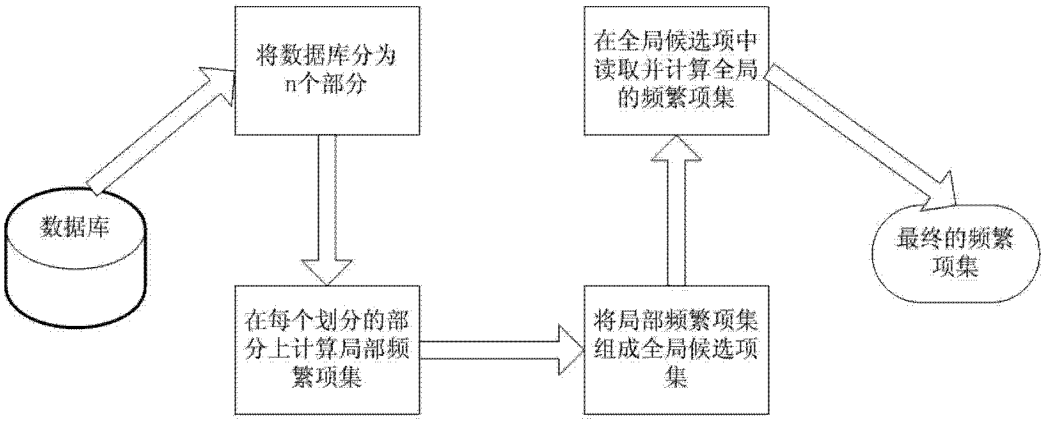

将数据库分为n个部分
在全局候选项中读取并计算全局的频繁项集
数据库
最终的频繁项集
在每个划分的部分上计算局部频繁项集
将局部频繁项集组成全局候选项集

图 3