

传感器网络中数据融合过程的事务分析与建模^{*})

胡 侃 刘云生

(华中科技大学计算机学院 武汉 430074)

摘 要 传感器网络是计算机科学技术的一个新的研究领域,它综合了传感器技术、嵌入式计算技术、分布式信息处理技术和通信技术,能够协作地实时监测、处理各种环境或监测对象的信息并传送给用户,具有十分广阔的应用前景。本文以面向事务的观点,基于事务 workflow 模型,结合数据的时间特性、事务流的依赖关系及数据与事务流间的相互关联,提出事务的控制区域的概念并给出一种基于应用事务的行为语义控制区域的事务模型,为传感器网络数据融合过程提供一个适当的正式模板;给出该模型的提交、回滚、可见定理及规则;通过性能测试,表明该模型提高了数据的相对外化率;模型主要解决了在分布的多用户环境下系统如何控制应用,保证应用语义逻辑的一致性。

关键词 传感器网络,数据融合,事务模型,事务控制,传感器事务

Transactional Analysis and Modeling of Data Fusion in Sensor Network

HU Kan LIU Yun-Sheng

(Faculty of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074)

Abstract Integrating sensor technology, embedded calculation technology, distributed information processing technology and communication technology, sensor network is a new research field in computer science and technology, which could be in collaboration with real time monitoring, processes all kinds of conditions or monitors the information of objects and then transmits to the users. Based on the model of transactional workflows, combined with the time characteristic of data, reliant relationship of transaction flows and interrelation between data and transaction flow, this paper raised the concept of transaction control zone(TCZ), established a transaction model based on the control zone of behavior semantics and model submission, rollback, visible theorem as well as regulation, and provided a proper formal template for the process of data fusion in sensor network. According to the performance testing result, it has been shown that the model enhanced the relative outer transformation ratio, resolved the problem of how to control the application in distributed multi-user conditions, and ensured the consistency of semantic logic.

Keywords Sensor network, Data fusion, Transaction model, Transaction control, Sensor transaction

传感器网络是信息感知和采集的一场革命。它综合了传感器技术、嵌入式计算技术、分布式信息处理技术和通信技术,能够协作地实时监测、感知和采集网络分布区域内的各种环境或监测对象的信息,并对这些信息处理,获得详尽而准确的信息,传送到需要这些信息的用户。因此,这种网络系统可以被广泛地应用于国防军事、环境监测、交通管理、医疗卫生等领域。它作为一个全新的研究领域,在基础理论和工程技术两个层面向科技工作者提出了大量的挑战性研究课题。目前的研究还处于起步阶段,主要集中在传感器网络技术和通信协议的研究,也开展了一些感知数据查询处理技术的研究,文[2~4]提出了确定传感器网络中节点位置的分布式算法、传感器网络中数据存储方法、变换初始感知为高级数据流的方法等,文[5]提出了几类新的通信协议,文[6~8]提出了传感器网络上可动态调整的感知数据连续查询方法、低能耗无线传感器网络中实现聚集函数的方法、把传感器网络表示为数据库的方法。而对传感器网络环境中数据处理的事务方面的研究尚少。本文试图以面向事务的观点,分析传感器网络应用中数据采集的融合过程。结合传感器网络应用的特点,

提出一种基于应用事务的行为语义控制的事务模型,并给出该模型的提交、回滚、可见规则。

1 事务型的传感器网络应用系统

1.1 传感器网络的概念及传感器网络应用的特点

我们可以如下定义传感器网络:传感器网络是由一组传感器以 Ad Hoc 方式构成的有线或无线网络,其目的是协作地感知、采集和处理网络覆盖的地理区域中感知对象的信息,并发布给观察者^[1]。

从上述定义可以看到,传感器、感知对象、观察者是传感器网络的 3 个基本要素;采集、融合、发布感知信息是传感器网络的基本功能;一组功能有限的传感器协作地完成大的感知任务是其重要特点。另外传感器网络还具有 Ad Hoc 网络的移动性、断接性、电源能力有限、实时性等特点。

传感器网络应用的上述特点给我们带来了新的挑战。传感器网络系统应维护一个面向事务的应用层面,支持具有上述特点的应用。传统的事务模型已无法描述这种复杂事务,需要新的功能强大的事务模型,以此指导研究新的并发、恢复

^{*}) 本课题得到国家自然科学基金项目(60073045)、博士点基金及国防预研基金(00J15.3.3.JW0529)资助。胡侃 博士生,讲师,主要研究多元实时数据集成、移动数据管理及实时数据库;刘云生 教授,博士生导师,主要研究方向为现代(实时、主动、内存、移动等非传统)数据库理论与技术及其集成实现、数据库与信息系统开发、实时数据工程、软件方法学与工程技术。

机制及系统结构。

1.2 面向事务的传感器网络系统

从感知对象、传感器、观察者这几个不同的角度看到的系统是完全不同的。感知对象把系统看作是一个状态满足 ACID 的抽象对象,感知对象通过传感器输入信息,通过转换器输出。传感器把系统看作是一个功能强大的通信机制。观察者将系统看作是一个大型实时数据库,这个数据库描述了现实世界中传感器网络控制区域的状态。因此,很难对系统下一个唯一的定义。本文采用面向事务的观点,给出如下解释性的事务性传感器网络系统定义。

定义 1 一个传感器网络事务系统提供工具使应用的编程、执行、管理自动化。系统支持多个设备组成的网络,设备可以向应用提交请求,系统应维护着表示现实世界状态的分布式实时数据库,它通过传感器感知现实世界,并可以驱使转换器改变现实世界的状态,应用的响应通常有时间限制,应用执行产生的影响满足 ACID。系统应有较长的生命期并无确定的停机时间。

传感器网络的主要功能是传感器协作地感知、采集数据,输入到传感器网络中进行融合处理,给观察者提供有关传感器网络控制区域中各种有价值的知识。简单地说,传感器网络数据融合系统是一个周期的、自动的或由观察者给出约束条件的函数 f 。该函数将传感器分布域转化为对应地理区域的知识。

2 基于事务流间动态数据依赖的控制区域模型

下面给出一种基于事务流之间动态依赖控制的事务模型(以下简称 MOC 模型)。该模型综合了事务工作流模型^[9]、数据流模型^[10]及扁平、嵌套事务模型,同时考虑数据及事务的时间特性,事务的提交控制由系统根据事务的请求及数据对象的历史做出决策,事务行为的控制由系统保证在只有当事务提交后才能将结果外化给观察者。能较全面地反映复杂的传感器网络应用事务。

2.1 基本概念

我们将传感器网络中由传感器数据采集引起的并由系统进行处理输出给观察者的数据融合应用称为传感器网络事务类型 SNT,它的一次执行称为一个传感器事务实例 SNt。通常系统周期性地接受传感器采集的描述感知区域中感知对象的数据,并触发多个相互独立的 SNT 同时执行。一个 SNT 是由多个满足 ACID 的扁平事务(FT)组成的具有松弛事务特性的事务工作流,其控制逻辑上是复合结构,执行时序是偏序关系。由于传感器网络的协作性,一个应用 SNt 使用的对象在其子事务局部提交后对另外的 SNt 可见,不必隔离到 SNt 提交。在应用逻辑层,系统应保证 SNt 的松弛原子性。MOC 模型通过控制区域的概念,系统使用控制域自动消除 SNt 对系统的影响。在这点上不同于 Saga 和 ConTract 模型^[11,12],它们是根据应用语义用补偿事务来实现的。

定义 2 一个传感器网络数据融合事务类型 SNT 为一个四元组:

$SNT = (A, R, S, C)$

$A = \langle ACTIVE \rangle$

$ACTIVE = \langle FT \rangle \mid \langle ACTIVE \langle OP \rangle ACTIVE \rangle \mid \langle \langle OP \rangle ACTIVE \rangle$

$FT = \text{扁平事务}$

$OP = \text{顺序操作} \mid \text{并行操作} \mid \text{条件操作} \mid \text{优先操作} \mid \text{循环操}$

作

$R = \text{a set of resources required by SNT}$

$S = \text{a structure control relation on FTs}$

$C = \text{constraints on S and R}$

定义 3 扁平事务 FT 的一次执行称为扁平事务实例 Ft。SNT 的一次执行活动 SNt 是一个二元组 $SNt = (F, \ll)$ 。F 是 SNT 中扁平事务实例的集合, $\ll$ 是定义在 F 上的反映扁平事务执行时序的偏序关系。

性质 1 一个 SNT 的执行应满足松弛原子性。

由于传感器网络中应用的协作性,不必要要求 SNt 满足严格的 ACID,放松了隔离性,但系统应保证一个 ABORT 的应用不应该影响到其它的应用以防止应用语义逻辑的不正确。

定义 4 每个 Ft 在 SNt 中是以黑盒形式出现的满足 ACID 的基本单元,从应用(SNT)的层面看,它对于对象 X 只能是读对象 R(X)(Ft 内部只有一个或多个读 X 操作)、写对象 W(X)(Ft 内部是先写 X 操作后读或写 X)或读写对象 U(X)(Ft 内部是先读 X 操作后有写 X 操作)。分别称为 Ft 是相对于对象 X 是读类型的、写类型的、修改类型的。

定义 5 MOC 模型将传感器网络中数据对象分为三种:感知对象,观察者对象,内部对象。感知对象是描述传感器控制区域中监测对象的数据,它由传感器采集并以流数据形式输入系统。观察者对象是描述监测对象经过处理后呈现(外化)给观察者的数据(如终端输出),MOC 将其存放在一个实时 DB 中,这种数据的操作是不可补偿的,因为数据一旦写入 DB 就会被观察者看到,系统无法消除这个写操作的影响。扁平事务实例 Ft 只能是输出(写)这样的数据。内部对象是事务处理过程中使用的系统内的永久对象。扁平事务实例 Ft 可读写内部对象。

性质 2 每个 SNt 看到的多个感知数据应满足时间一致性(即它们是描述感知对象在同一时刻状态的多个数据),外化数据应满足外部一致性(即它们是由满足松弛原子性的事务提交的符合应用语义逻辑的数据),无论出现任何例外内部数据应满足内部一致性(数据库一致性)。

定义 6 事务推迟的写指的是事务在执行过程中的写对象操作并不立即执行,而是推迟到事务 COMMIT 时,即 COMMIT 意味着写。系统要保证 COMMIT 与写的原子性。

SNt 中的任意子事务 Ft 提交后,它所使用的内部对象在 SNt 未提交前可由其它的 SNt 的子事务 Ft 使用,这带来了这样的动态数据依赖问题;Ft 写的一个对象 X 被 Ft 读。当 SNt 因为某原因 ABORT,从系统数据库的角度看 X 是正确的,但从应用层来看,SNt 是以一个不可靠的数据 X 为依据产生一个结果给观察者,这个结果并不一定保证应用语义逻辑的正确。Ft 写 X 对 SNt 应用语义逻辑无影响。在 MOC 模型中,系统并不保证事务工作流中由子事务提交的对象只能由该事务流中其它子事务使用,这种逻辑正确性交由应用保证。

性质 3 从应用语义逻辑来看,应用应满足原子性。即一个 SNt 不应使用(读)另一个已 ABORT 的 SNt 中的子事务提交的对象。

为解决上述问题,我们提出控制域(SOC)的概念。将数据看作是一个随时间变化的值空间,一个 SOC 由一组存在数据依赖的 SNt 集合及与这组 SNt 相关的数据空间组成。一个 SNt 的提交由系统的 SOC 决定是否同意或推迟,其 ABORT 要在相关数据空间中作出标识,消除对系统数据的影响。

在扁平事务层,因为 ACID 特性,系统只需每个内部对象的当前版本。而在 SNT 层,一个 SNT 产生的对象 X 的版本,在其未提交前可以被其它 SNT 使用,因此系统应维护对象的多版本历史。

定义 7 MOC 模型将传感器网络中的内部对象(O)看作是由数据名及其历史构成的二元组 $O=(ID, H)$ 。ID 是对象 O 在系统中的唯一标识。H 是对象 O 的历史, $H=\{(V_i, [t_i, t_{i+1}), A_i), \dots, (V_n, [t_n, *), A_n)\}$, V_i 代表对象 O 在时间段 $[t_i, t_{i+1})$ 的值, A_i 是对象 O 在时间段 $[t_i, t_{i+1})$ 的属性,表示何事务以何方式使用对象 O 等。* 代表当前时间。

在 MOC 模型中,内部对象在应用层是一个基于时间的序列值,每个值在一段时间内是有效的。

性质 4 子事务 Ft 的 COMMIT 意味着对象历史的改变,ABORT 则意味着对象历史无变化。无论 Ft 是读、写或修改类型的操作,当其提交时,对象 O 历史都由 $(V_n, [t_n, *), A_n)$ 改变为 $(V_{n+1}, [t_{n+1}, *), A_{n+1})$ 。对于读操作 $V_{n+1}=V_n$ 。写或修改操作 V_{n+1} 是新写入的当前版本。

性质 5 扁平事务的可串行化使得对象 X 在时间段 $[t_i, t_{i+1})$ 属于且只属于一个 SNT,多个时间段(连续或不连续)可属于同一个 SNT。

定义 8 若一个 SNT 中的任意子事务 Ft 提交后,它所使用的内部对象在 SNT 未提交前由其它的 SNT 的子事务 Ft' 读(即 Ft' 是读类型或修改类型),则称 SNT 动态依赖于 SNT。

2.2 MOC 模型

MOC 模型采用三层系统模型。底层提供具有严格 ACID 特性的扁平事务,它们是组成传感器数据融合应用的基本粒子。中间层是由基本粒子通过静态控制关系复合组成的具有松弛事务特性的事务工作流,是应用的事务描述。最高层是存在动态数据依赖关系的事务流及相关数据历史阶段组成的控制区域(SOC),描述应用的动态行为,如图 1 所示。

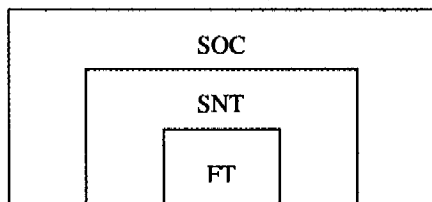


图 1 MOC 模型中系统控制对象的层次模型

定义 9 一个控制域是一个二元组 $SOC=(T, O)$, T 是具有动态依赖关系的 SNT 集合,其中的任一个元素必至少存在 T 中的另一个不同的元素,它们之间存在动态依赖关系。O 是所有属于 T 中 SNT 的对象的时间段区域。

定义 10 对于任一个 SNT,若 S 是由所有动态依赖于 SNT 的事务组成的集合,则 S 的传递闭包 S^+ 称为 SNT 的级连事务集。SNT 的级连域 $SOA=(S^+, O)$, O 是属于 S^+ 中的所有对象数据的时间段集合。

性质 6 由定义 7、8 可知,在任意时刻系统中存在多个(零个)SOC 且它们是不相交的,当前未提交的 SNT 属于且仅属于一个 SOC。随着时间的前进, SOC 自动地发生变化。

性质 7 一个 SNT 对系统内部的影响是该 SNT 的 SOA。针对 MOC 模型,我们给出该模型的提交、回滚、可见定理。

定理 1 一个 SNT 回滚,则该 SNT 的 SOA 中的所有 SNT

应回滚。且系统所有对象 X 的历史中属于这些 SNT 的区间段应被标记为无效。

定理 2 一个 SNT 中的 Ft 看到的(读)对象 X 应是 X 历史中最近的没有被标记为无效的值。

定理 3 一个 SNT 可提交当且仅当在包含该 SNT 的 SOC 中,所有其它的 SNT 的 SOA 中不包含该 SNT 且在今后该 SNT 不会属于任何 SOC。仅当该事务提交后,其中子事务对观察者对象的写操作才可外化给观察者(写入实时 DB)。

显然,上述三定理保证了 SNT 应用逻辑的松弛原子性,它们的证明是直观的。

为保证数据满足性质 2 的三个一致性,给出下面三个规则。

规则 1 时间一致性规则:MOC 模型中, SNT 是被周期执行并由传感器采集的数据驱动的。描述感知对象的流数据是作为输入参数去调用 SNT 的,每个 SNT 具有全局唯一的 ID,同一个 SNT 中每个 Ft 的输出同 ID 一起绑定,作为输入消息发送给同一 SNT 中的其它 Ft。

规则 2 外部一致性规则:MOC 模型要求系统保证 SNT 对观察者对象的写操作是推迟的写(定义 6)。系统使用事务概念将不受保护的实操作在更高的应用级封装变为受保护的,使得 SNT 对外界产生的影响被推迟到其 COMMIT 时,当出现 ABORT 时,系统不影响外界并能通过控制域自动消除对系统内部的影响,从而保证应用的语义逻辑一致性。

规则 3 内部一致性规则:系统内部对象通常存放在系统内部数据库中。内部数据是由满足 ACID 的扁平事务实例使用,从而保证了内部数据库的一致性。

系统保证 SNT 满足三定理且遵守三规则,从而保证了传感器网络系统中数据的外部一致性、时间一致性及内部一致性。

MOC 模型主要是解决在分布的多用户环境下系统如何控制应用(SNT),保证应用语义逻辑的一致性。其主要思想是系统保持每个对象的执行历史并监测每个应用事务之间的相互依赖关系。尽管扁平事务提交已保证了内部数据库的一致性,但系统从应用逻辑层建立更高的控制区域(SOC)控制应用的原子性。一个应用对外界的影响只有在应用提交后才能产生,当出现意外时,系统不仅撤消扁平事务使其保持原子性,同时系统能通过对象历史从更高的应用逻辑层撤消 SNT 的行为影响(SNT 的 SOA)。

2.3 MOC 的提交、回滚、可见协议

按照 MOC 模型,一个 SNT 的提交由传统的分布式环境下的二阶段提交协议发展为三阶段提交协议,它包括扁平事务提交阶段、SNT 提交阶段及 SOC 提交阶段。下面给出该模型的提交、回滚、可见规则。

提交协议:

- 1)扁平事务实例 Ft: IF Ft 提交 THEN
 { 将输出数据与 ID 绑定
 连同提交信息发送给 SOC
 管理器};
 IF 收到同意提交 THEN
 {提交子事务};
- 2)SNT: IF SNT 提交 THEN
 {发提交请求给 SOC 管理器};

```

3)SOC:  IF 收到 SNt 提交请求 THEN
    {按定理 3 判断是否可提交;
    IF 不可提交
    THEN
        {挂起该 SNt}
    ELSE
        {求出该 SNt 的 SOA;
        将处于挂起状态且属于该 SOA
        中的所有 SNt 提交};
    IF 收到一个 Ft 提交请求 THEN
        IF 存在一个在运行且包含 Ft 的 SNt
        THEN
            {修改数据历史并同意提交;
            转调度器调度其它 Ft}
        ELSE
            {回滚该 Ft};

```

考虑到传感器事务是周期性的不间断执行,一个 Ft 的 ABORT 意味着整个 SNt 的 ABORT。这样可避免该 Ft 的 RESTART 推迟了其后面周期已开始的 SNt 的提交。而传感器网络并不要求每个周期的事务必须提交。

回滚协议:

```

1)Ft:  IF Ft 回滚 THEN
    {回滚该 Ft 并将 ID 及回滚请求绑
    定发给 SNt}
2)SNt:  IF 收到一个 Ft 回滚请求 THEN
    IF 存在一个在运行且包含该 Ft
    的 SNt THEN
        {置其为 ABORT 状态,
        其提交数据历史时间段无效,
        发送回滚请求给 SOC}
3)SOC:  IF 收到一个 SNt 回滚请求 THEN
    {求出该 SNt 的 SOA;
    标记 SOA 中数据历史;
    置 SOA 中的所有 SNt 回滚}

```

当 SNt 收到一个 Ft 回滚通知后,并不回滚其它正在运行的属于该 SNt 的 Ft,它们随后的提交将由上述规则保证自动回滚。

可见协议:

```

1)每个 Ft 正在使用的对象当其未提交或回滚时对其它
任何 Ft 是隔离的,提交后对它们可见。
2)一个 Ft 能读的对象值是该对象历史中最近的没有被
标记为无效的值。
3)每个 SNt 提交后,其包含的所有对观察者对象操作结
果才对外界可见。

```

3 MOC 体系结构

系统体系结构如图 2 所示。

包括扁平事务 (FT) 管理器、传感器事务 (SNT) 管理器、控制区域 (SOC) 管理器。FT 管理器负责扁平事务的提交、回滚、读写,并与局部日志管理器、锁管理器一起保证扁平事务的 ACID。SNT 管理器执行负责传感器事务的提交、回滚。SOC 管理器负责决策 SNT 的提交、回滚,这样使得系统能够灵活地根据应用的特征制定多种提交、回滚策略,从而实现多

样的、动态的提交、回滚控制。每个管理器包括一个监控器和一个永久性的请求/响应队列管理器,永久性队列中存放该对象接受的请求及响应,队列管理器负责保证队列中请求能够实时地、原子地得到响应。监控器负责保证收到的请求放入队列中。这种缓存请求的方案用来减少系统损耗。

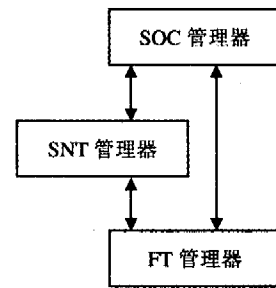


图 2 系统体系结构图

4 性能评价

我们在由国家自然科学基金资助、自行研制的分布式主动实时数据库原型系统 ARTs-II 上模拟传感器网络计算环境完成了 MOC 的提交协议的性能测试。实验采用采集数据的外化率 (单位时间观察者看到数据) 与采集数据的采集率 (单位时间输入系统的采集数据) 之百分比 MR (相对外化率) 为测试指标。

4.1 相关模型对比

传统的扁平事务“要么全做,要么全不做”的特点既是优点也是缺点。它用最简单的扁平逻辑描述事务及操作对象的结构、用最简单的方法处理所有的失败语义。但在描述传感器网络数据融合应用中存在着描述复杂应用逻辑能力弱、采集数据的外化率低、实时性差等问题。嵌套事务对扁平事务进行了改进,它形成了一个层次的控制框架。这种框架对描述具有复杂结构的应用及对回滚范围进行精细控制提供了有力的机制,它们在解决分布环境下复杂长事务问题是有效的,由于其根事务对其它嵌套事务是隔离的,对传感器网络中多应用事务处理中数据外化率仍较低且协作性较差。这两种模型对传感器网络数据融合应用来说太严格。

数据流模型及事务流模型放宽了隔离性,使得系统的实时性提高,在测控系统中得到很好的应用。但数据流没有考虑事务上下文问题,现有事务流要求存在应用语义上的补偿事务,这在传感器网络数据融合的实际复杂应用中通常不能满足且补偿事务增加了系统开销。

MOC 模型结合上两者特点。它允许已 ABORT 的事务流的子事务提交的数据存在于系统中,因为这种数据是保证数据库满足一致性的,但在应用层又对其历史做了无效标记,防止其它事务流使用该数据的无效历史值,保证应用的语义逻辑原子性。这种方案的实时性强于扁平事务及嵌套事务模型低于数据流模型及事务流模型,但在保证应用逻辑正确性方面强于后者。

4.2 实验结果分析

受实验环境限制,我们采用由一个传感器采集数据,通过改变采样周期,分别用扁平事务、嵌套事务、MOC 事务进行测试。表 1 是主要模拟参数。

图 3 是采样周期对 MR 的影响。可以看出,当采样周期越小, MOC 提交方式的 MR 值明显优于其它两种方式,这是松弛原子性的特点。

表1 模拟参数

parameter	Value	parameter	value
test time	10s	size of database	500
number of nodes	5	number of sub-transaction	1—5
average message send time	316us	number of operations in each sub-transaction	5—15
data send period	10—100ms		

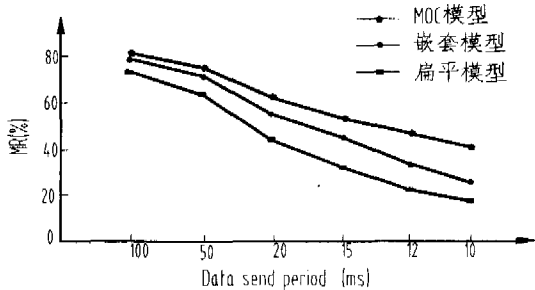


图3 采样周期对MR的影响

结束语 传统的事务模型不能够较好的支持传感器网络中数据融合事务的执行。本文结合扁平事务、嵌套事务模型、数据流及事务工作流模型提出了一种基于事务及相关数据历史的控制域的较为适应该环境的 MOC 模型,并给出了相关定理和提交规则、回滚规则、可见规则。通过模拟实验进行测试,证明其可以提高监测数据的系统融合事务的实时性并能保证应用的松弛原子性。本系统在我们的原有分布式数据库 ARTs-II 的基础上正在研制过程中。

(上接第 80 页)

列不同的变量中,如图 3(b)所示。

使用该转换函数,我们产生了一个简易的“黑匣子”机制,将原有的代码转换为黑匣子保护代码(如图 3(a)所示的原有代码和图 3(c)所示的黑匣子保护代码)。

攻击者试图获取电子货币的值。它知道代理的黑匣子保护代码,但不知道转换函数,所以并不知道原来的代码。由于黑匣子保护方案存在着一个严重的问题:即使原有的值通过转换函数已经变化了,但它仍然是明文,仍然可以作为程序调用的参数,因此攻击者可以等待代理执行调用购买子程序的指令,然后读出正确的调用参数,该参数即电子货币的值。攻击程序的代码如图 4 所示。

```

攻击子程序
10  push "Call 40"      //将指令所对应的数值放入堆栈;
13  equals?            //比较栈顶两个元素的值
14  if true goto 17
   else goto 20
17  ...
18  ...
20  push "Call 40"
23  return

```

图4 攻击程序

攻击 RASPS 将 Call 40 的指令放入到自己的堆栈中,然后将下一条指令“equals”放入到堆栈中,并在攻击程序执行前进行解释执行,该指令将代理所要执行的指令和 Call 40 指

参考文献

- 1 Tilak S, Abu-Ghazaleh N B, Heinzelman W. A taxonomy of wireless micro-sensor network models. *Mobile Computing and Communications Review*, 2002, 1(2): 1~8
- 2 Savarese C, Rabaey J. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In: Park Y, ed. *Proceedings of the USENIX Technical Annual Conference*, Monterey: USENIX, 2001. 317~328
- 3 Ratnasamy S, Karp B, GHT: A geographic hash table for data-centric storage. In: Raghavendrv CS, ed. *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, New York: ACM Press, 2002. 94~103
- 4 Girod L, Bychkovskiy V, Elson J, et al. Locating tiny sensors in time and space: A case study. In: Manoli Y, Kim KS, eds. *Proceedings of the International Conference on Computer Design*, Piscataway: IEEE Press, 2002. 195~204
- 5 Rentala P, Musunuri R, Gandham S, et al. Survey on sensor networks. [Technical Report]. UTDCS-33-02, University of Texas at Dallas, 2002
- 6 Madden SR, Szewczyk R, Franklin MJ, et al. Supporting aggregate queries over ad-hoc wireless sensor networks. In: Kindberg T, ed. *Proceedings of the Workshop on Mobile Computing and Systems Applications*, Los Alamitos: IEEE Computer Press, 2002. 49~58
- 7 Madden SR, Shah MA, Hellerstein JM, et al. Continuously adaptive continuous queries over streams. In: Franklin MJ, Moon B, Ailamaki A, eds. *Proceeding of the SIGMOD Conference*, New York: ACM Press, 2002. 49~60
- 8 Bonnet P, Gehrke JE, Seshadri P. Towards sensor database systems. In: Tan K-L, Franklin MJ, Lui JCS, eds. *Proceedings of the 2nd International Conference on Mobile Data Management*, Hong Kong: Springer-Verlag, 2001. 3~14
- 9 Jajodia S, Kerschberg L, eds. *Advanced Transaction Models and Architectures*, Kluwer, 1997
- 10 Whiting PG, Pascoer SV. A history of data-flow languages. *IEEE Annals of the History of Computing*, 1994, 16(4): 38~59
- 11 Garcia M H, Salem K. Sagas. In: *Proceedings of SIGMOD International Conference on Management of Data*, San Francisco, 1987. 249~259
- 12 Reuter A, Schwenkres F. ConTracts-a low level mechanism for building general purpose workflow management system. *Data Engineering*, 1995, 18(1): 4~10

令进行比较,如果结果为 true,则这时攻击 RASPS 就获得了电子货币的值,因为此时电子货币的值刚好在堆栈的顶部。因为“equals”操作去掉了原有堆栈顶部的两个元素,而“if”操作又去掉了新放在堆栈顶部的布尔值,这样作为 buy 操作的 call 40 指令所需要的参数刚好在堆栈的顶部。

小结 恶意主机问题是移动代理系统获得广泛使用必须解决的问题。我们分析了恶意主机对代理的攻击,并比较了目前已有的各种代理保护方案,进而基于抽象机器模型,提出的恶意主机对代理的攻击模型,从理论上分析了攻击的可能性以及攻击的方法;另一个方面也有助于提出新的代理保护方案,并对各种保护方案进行评价。

参考文献

- 1 王常杰,张方国,王育民. Internet 移动代理技术中的安全性研究. *西安电子科技大学学报(自然科学版)*, 2001, 28(2)
- 2 朱向华,万燕,孙永强. 移动代理系统的安全机制. *计算机科学*, 2001, 27(1)
- 3 何炎祥,陈萃萌. *Agent 和多 Agent 系统的设计和应用*. 武汉:武汉大学出版社
- 4 (美) Bruce Schneier 著. *应用密码学——协议、算法与 C 源程序*. 吴世忠,祝世雄,张文政,等译. 北京:机械工业出版社, 2000
- 5 Pfleeger C P. *Security in Computing*. Second Edition. Upper Saddle River, NJ: Prentice Hall, 1997
- 6 Chess D. Security Issues in Mobile Code Systems. In: Vigna G, Ed. *Mobile Agent Security*, LNCS 1419, Springer, 1998. 1~14