一种用于移动 Agent 数据保护的机制*

谭 湘+, 顾毓清, 包崇明

(中国科学院 软件研究所,北京 100080)

A Method for Mobile Agent Data Protection

TAN Xiang+, GU Yu-Qing, BAO Chong-Ming

(Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: +86-10-82680272, E-mail: java-tan@163.net, http://www.iscas.ac.cn

Received 2003-06-20; Accepted 2004-07-27

Tan X, Gu YQ, Bao CM. A method for mobile Agent data protection. *Journal of Software*, 2005,16(3): 477−484. DOI: 10.1360/jos160477

Abstract: Focusing on the particular security problem of mobile Agent data protection, the IKCE (interrelated keys chains encryption) method is proposed. Security analysis and performance analysis for the IKCE method are surveyed. The key idea of the IKCE method is to establish an interrelated relationship among encryption keys used in encrypting Agent data in order to protect the Agent data. The research states that the IKCE method is available for protecting the Agent data.

Key words: mobile Agent system; security; confidentiality; integrity

摘 要: 针对移动 Agent 数据保护这一特定的安全问题,提出了关联密钥链加密 IKCE(interrelated keys chains encryption)机制,并对该机制进行了安全性分析和性能分析.IKCE 机制的核心思想是在移动 Agent 数据加密的密钥之间建立一种关联关系,从而达到 Agent 数据保护的目的.研究表明,IKCE 机制对于移动 Agent 数据保护是可行的.

关键词: 移动 Agent 系统;安全;机密性;完整性

中图法分类号: TP309 **文献标识码**: A

移动 Agent 是一个能在异构网络中自主地从一台主机迁移到另一台主机,并可以与其他 Agent 或资源交互的程序,实际上,它是 Agent 技术与分布式计算技术的结合体,具有移动性和自主性等特点[1].随着移动 Agent 技术在电子商务、分布式计算、信息搜索等领域的应用,安全性问题显得日益重要.

本文就移动 Agent 安全的特定问题——移动 Agent 数据保护问题进行了探讨.第 1 节分析了移动 Agent 数据保护问题及其相关工作.第 2 节描述了关联密钥链加密 IKCE(interrelated keys chains encryption)机制.第 3 节对 IKCE 机制进行了安全性分析.第 4 节是 IKCE 机制的性能分析与对比.第 5 节在 Aglet 平台上实现了 IKCE 机制.最后做了总结.

* **作者简介**: **谭湘**(1976−),男,湖南攸县人,博士生,主要研究领域为移动 Agent 系统安全技术;**顾毓清**(1940−),男,研究员,博士生导师,主要研究领域为软件工程,分布式计算技术;**包崇明**(1970−),男,博士生,主要研究领域为分布式计算技术.

1 移动 Agent 数据保护问题及相关工作

1.1 移动Agent数据保护问题

移动 Agent 系统是由移动 Agent 和多个为移动 Agent 提供服务的主机组成的.但是我们并不能保证整个系统中每个主机都是完全可靠的,其中有可能存在一些恶意主机.这些恶意主机试图攻击移动 Agent,窃取移动 Agent 的重要信息,更为严重的是,对 Agent 的信息进行篡改,使之产生不正确的结果.因此,当在潜在的恶意环境下执行,移动 Agent 的保护问题是尤为重要的.对移动 Agent 的保护,具体到移动 Agent 所包含的数据来说,主要是保护移动 Agent 数据的机密性和完整性[2].

(1) 移动 Agent 数据的机密性.在移动 Agent 系统中,如果没有采用任何保护措施,移动 Agent 对主机来说是完全透明的,主机可以访问移动 Agent 的任意代码和数据,移动 Agent 对主机来说没有机密性可言.然而在实际的移动 Agent 应用系统中,移动 Agent 所运行的每一个主机上都可能产生新的数据,而 Agent 可能会收集这些数据返回给 Agent 所有者的主机,且不允许他人了解这些信息,这就对移动 Agent 的数据提出机密性的要求.考虑一个具体的应用例子:网上商店 A 和 B,当 Agent 所有者发送一个移动 Agent 到 A 商店的主机上,查询某商品的价格并保存到该 Agent,然后再迁移到 B 商店的主机上,继续查询该商品的价格,从而找到最便宜的价格.为了维护自己的商业利益,每一个主机总是希望自己给 Agent 的商品价格对于系统中其他的主机来说是机密的,Agent 所有者也希望 Agent 已经收集到的价格对其他主机是机密的.此外,还有很多实际的移动 Agent 的应用也有类似的安全要求,如电子商务系统中信用卡账号等敏感信息的机密性要求.因此必须实现对移动 Agent 数据的机密性保护,以防止数据的泄密.

(2) 移动 Agent 数据的完整性.由于移动 Agent 是在主机上运行的,移动 Agent 的代码和数据完全由主机所掌握.主机可以根据代码进行计算,修改移动 Agent 的数据,但是同时恶意主机也可以出于破坏目的来篡改移动 Agent 的数据.举个例子说,B 商店可能修改移动 Agent 中保存的 A 商店的价格.虽然 A 商店的价格信息可能会用 Agent 所有者的公钥加密,别人无法获取秘密,但是 B 可能故意扔掉 A 的信息,或者以明文形式代替它.可见,移动 Agent 本身不能阻止恶意主机篡改或删除它的数据,但是能够采取措施检测到这些篡改或删除.在移动 Agent 系统中,必须提供措施保证任何对于 Agent 的篡改或删除攻击都能被检测到,从而保护移动 Agent 数据的完整性.

1.2 相关工作

移动 Agent 数据的保护并不是一个简单的问题,目前在这方面进行了很多的研究,但是还没有完整的解决方案可用.当前,在该领域的研究可以分为两类:

(1) 基于检测的保护措施

根据对运行环境进行检测来判断其是否安全,以及对移动 Agent 的执行结果进行检测来判断其是否受到了攻击并遭受破坏.如使用可信任第三方实体(TTP)方法[3]、路径哈希链方法[4]、Co-Signing 方法[5]等.

(2) 主动的保护措施

基于检测的方法是被动的,它只能检测到主机对 Agent 的攻击,并不能真正保护移动 Agent 在不信任的运行环境上安全运行.要让移动 Agent 在不信任主机上完全地安全运行,目前大都是基于硬件的方案.如信任运行环境(TPE)就是一种基于硬件的保护方法[6].

2 IKCE 机制

本文针对移动 Agent 数据保护问题提出了一个关联密钥链加密机制——IKCE(interrelated keys chains encryption)机制.该机制基于以下假设:Agent 能够动态决定它所要迁移的目的主机,而不是被强迫地遵守预先定义的路径;公钥体系已经建立,管理每个实体的证书.

2.1 表示法

为了更好地描述,表 1 列举了本文常用符号的意义.

Table 1 The means of symbols

表 1 符号意义

Symbol	Means
KS	Session key used in conventional encryption scheme
KC	Encryption key used in conventional encryption scheme
KR_a	Private key of A, used in public-key encryption scheme
KU_a	Public key of A, used in public-key encryption scheme
EP_K	Public-Key encryption
DP_K	Public-Key decryption
EC_K	Conventional encryption
DC_K	Conventional decryption
SIG	Digital signature
H	Hash function
R	Random number
$\|\|$	Concatenation
$COMP$	Compare
TS	Timestamp

2.2 PGP加密与认证

PGP(pretty good privacy)通过数字签名和加密保证了通信双方的通信数据的机密性和完整性[7].其基本原理如图 1 所示.

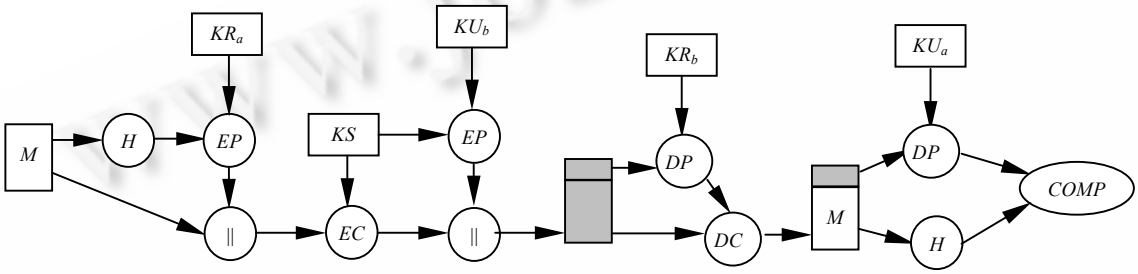

Fig.1 Fundamentals of PGP

图 1 PGP 基本原理图

但 PGP 只是保证了两方通信的机密性和完整性,对于移动 Agent 在多个主机中迁移所产生的机密性和完整性问题就无能为力了.如果能够使每次迁移之间的会话密钥 KS 保持关联关系,就能够保证数据的机密性和完整性.基于这种设想,我们提出了 IKCE 机制.

2.3 IKCE机制描述

IKCE 机制描述如下:

关联关系(如图 2 所示):

$KS_0=H(R1_0)$,

$KC_i=H(KS_{i-1},R1_i)$,

$KS_i=H(KC_i,R2_i)=H(H(KS_{i-1},R1_i),R2_i)$,

$K_i=EP_{KU(i+1)}(KS_i)$.

加密形式:

$C=\{Code,SIG_0(Code)\}$,

$D_i=\{EC_{KCj}(Data_j)|0<j\le i\}$,

$R_i=\{EP_{KU0}(R1_j,R2_j,S_j,SIG_j(D_j,TS_j),TS_j)|0<j\le i\}$.

传输内容:

$S_i\rightarrow S_{i+1}:\{C,K_i,D_j|0\le j\le i,R_j|0\le j\le i\}$

整个机制如图 3 所示.

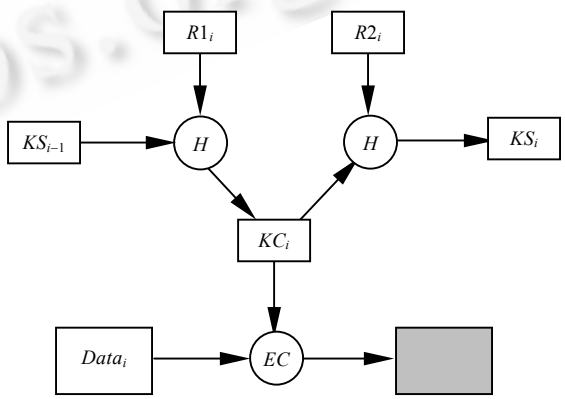

Fig.2 The interrelated relationship among encryption keys

图 2 密钥关联关系

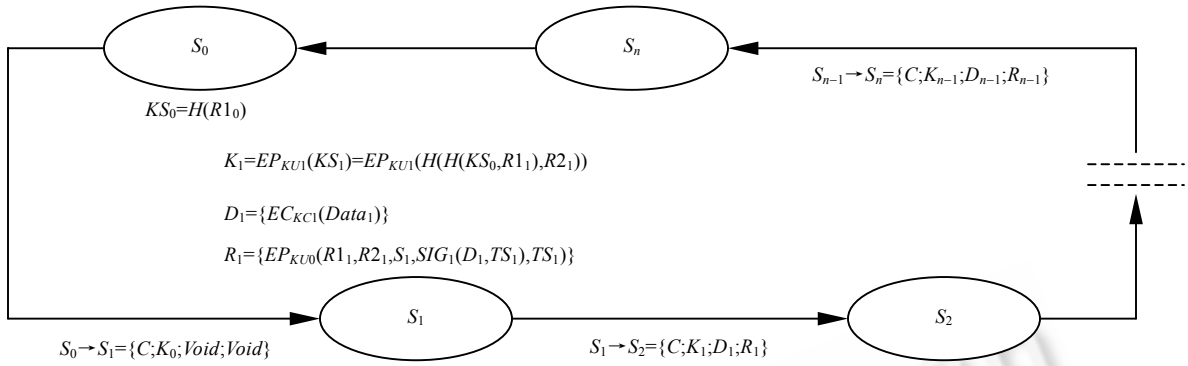

Fig.3 The method of mobile Agent data protection

图 3 移动 Agent 数据保护机制

从上述描述中可以看出,IKCE 机制具有如下特点:

(1) IKCE 机制是一种基于检测的方法,属于软件保护方案.

(2) 对移动 Agent 的数据采用常规加密,从而保护数据的机密性.

(3) IKCE 机制的核心思想是在移动 Agent 数据加密所用的密钥之间建立关联关系,形成密钥链,任何打破这种链关系的行为都将被检测到,从而保护数据的完整性.

(4) 密钥链中的密钥是通过对随机数和上一个密钥进行两次哈希而形成的.由于随机数的随机性,密钥链中的密钥是一次性的.由于哈希函数的单向性,密钥链是不可逆推的,即任何一个主机都无法逆推出上一个主机所使用的加密密钥.

2.4 算法详细说明

整个算法分为 3 个部分:

(1) 在 Agent 所有者主机 S_0 上的初始化:

(a) Generate Random Number:$R1_0(128)$

主机 S_0 上产生 128 位的随机数 $R1_0$,并保存好该随机数.

(b) Run:$K_0=EP_{KU1}(KS_0)=EP_{KU1}(H(R1_0))$

计算出初始的 KS_0,KS_0 是对 $R1_0$ 进行哈希得到的,采用的哈希算法为 MD5.把 KS_0 用下一个主机的公钥加密后生成 K_0.

(c) Initialize:$D_0=\{void\}$和 $R_0=\{void\}$

初始化 D_0 和 R_0,其初始值均为空.

(d) transfer:K_0,D_0,R_0

将 K_0,D_0,R_0 传送给主机 S_1.

(2) 在中间主机 S_i 上运行

(a) Generate Random Number:$R1_i(128),R2_i(128)$

主机 S_i 产生两个 128 位的随机数 $R1_i$ 和 $R2_i$.采用 Agent 所有者主机 S_0 的公钥 KU_0 加密,这样,只有 S_0 才能解密它们,合法地得到这两个随机数的值.

(b) Run:$KC_i=H(R1_i+KS_{i-1})$

KC_i 是对 $R1_i+KS_{i-1}$ 进行哈希得到的,KS_{i-1} 是上一个主机所传送过来的.采用的哈希算法为 MD5.MD5 算法以任意长度的信息作为输入,生成 128 位的哈希值.

(c) Run:$Data_i$ compute on S_i

Agent 在主机 S_i 上运行产生自己的数据 $Data_i$.

(d) Append:$D_i=D_{i-1}+EC_{KCi}(Data_i)$

主机 S_i 加密自己产生的数据,采用的是常规加密算法 IDEA,密钥为 KC_i.IDEA 算法的密钥长度为 128 位,

正好和 MD5 算法的哈希值长度一致.加密后的数据存储到 Agent 的数据项中.

(e) Append:$R_i=R_{i-1}+EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$

主机 S_i 首先产生时戳,再对已有的数据项 D_i 和时戳签名.然后把该签名和时戳 TS_i 以及随机数 $R1_i$ 和 $R2_i$ 一起采用 Agent 所有者主机 S_0 的公钥 KU_0 加密.采用的签名算法为 DSA,采用的公钥加密算法为 RSA.

(f) Run:$K_i=EP_{KU(i+1)}(KS_i)=EP_{KU(i+1)}(H(R2_i+KC_i))$

KS_i 是对 $R2_i+KC_i$ 进行哈希得到的,采用的哈希算法为 MD5.把 KS_i 用下一个主机的公钥加密后生成 K_i.

(g) Transfer:K_i,D_i,R_i

主机 S_i 将 K_i,D_i,R_i 传送给下一个主机 S_{i+1}.

(3) Agent 返回到 Agent 所有者主机 S_0

(a) Run:$KS_0=H(R1_0)$

这是对 $R1_0$ 进行哈希得到 KS_0.采用的哈希算法为 MD5.

(b) Run:$DP_{KU0}(EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i))$

对 $EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$进行解密,密钥为 KU_0,解密后得到 $R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i$.

(c) Run:$KC_i=H(KS_0+R1_i)$和 $KS_i=H(KC_i+R2_i)$

由上一步得到的 $R1_i$ 和 $R2_i$ 进行计算,得到 KC_i 和 KS_i.采用的哈希算法为 MD5.

(d) Run:$DC_{KCi}(EC_{KCi}(Data_i))$

解密 $EC_{KCi}(Data_i)$得到 $Data_i$,解密密钥为 KC_i.

(e) Verify:$EP_{KUi}(SIG_i(D_i,TS_i)$

验证数字签名 $SIG_i(D_i,TS_i)$和时戳,检查数据的完整性.

3 安全性分析

目前,基于软件的方法大都采用基于检测的方法来保护移动 Agent 的数据,IKCE 机制也是如此.因此安全性定义为:保证攻击者无法获取 Agent 的机密数据,并保证 Agent 所有者能够检测出攻击者对 Agent 数据的篡改.

IKCE 机制的安全性基于以下假设:

(1) 机制所用的对称加密算法、公钥加密算法和数字签名算法是安全的.

(2) 机制所用的随机数生成算法是安全的.

(3) 机制所用的哈希算法是安全的.

(4) 机制所涉及到的各个主体的私钥是安全的.

在以上假设的基础上,IKCE 机制能够防范被动攻击者、主动攻击者、被动骗子、主动骗子的攻击,能够保护移动 Agent 的数据.安全性分析如下:

(1) 被动攻击者

被动攻击者监听信道,试图窃取敏感信息.

假设被动攻击者能够截获在网络上传递的 K_i,D_i,R_i,试图窃取移动 Agent 数据,然而:

$K_i=EP_{KU(i+1)}(KS_i)=EP_{KU(i+1)}(H(R2_i+KC_i))$.根据安全假设,主机的私钥是安全的,所以攻击者无法解密 K_i,无法从 K_i 中获取任何信息.

$D_i=D_{i-1}+EC_{KCi}(Data_i)$.主机 S_i 加密自己产生的数据 $Data_i$,采用的是常规加密算法 IDEA.根据安全假设,攻击者在没有密钥 KC_i 的情况下是无法从 D_i 中获取 $Data_i$ 的.

$R_i=R_{i-1}+EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$.$R_i$ 所包含的信息是采用 Agent 所有者主机 S_0 的公钥 KU_0 加密的.根据安全假设,主机的私钥是安全的,所以攻击者无法解密 $EP_{KU0}(R1,R2,S_i,SIG_i(D_i,TS_i),TS_i)$,也就是说,无法从 R_i 中获取任何信息.

(2) 主动攻击者

主动攻击者可能伪装成 IKCE 机制的参与者,在 IKCE 机制中引入新的信息,删除原有的信息,用另外的信息代替原有的信息等.

首先,主动攻击者无法做到引入新的信息或者删除原有信息而不被发现.假设主动攻击者可以拦截 K_i,D_i,R_i 并进行攻击.因为 K_i,D_i,R_i 都有明确的意义和形式,如果主动攻击者引入新的信息或者删除原有信息,则接受该消息的下一个主机立即能够发现格式的变化,知道被攻击了.

其次,主动攻击者也无法做到用另外的信息代替原有信息而不被发现.假设主动攻击者截获在网络上传递的 K_i,D_i,R_i,并恶意修改 Agent 数据.如攻击者冒充成 S_i 将数据项 $EC_{KCi}(Data_i)$修改成其他数据,但是由于主动攻击者没有 S_i 的私钥,因此无法伪造 S_i 的签名,只能用自己的私钥对 D_i 签名.当回到所有者主机时,检测数字签名就能发现存在恶意篡改.

(3) 被动骗子

被动骗子遵守 IKCE 机制,但企图是获取 IKCE 机制以外的信息,试图窃取前面主机所产生的移动 Agent 数据.

假设被动骗子 S_{i+1} 收到上一个主机 S_i 传递的 K_i,D_i,R_i,试图窃取前面主机所产生的移动 Agent 数据,然而:

$K_i=EP_{KU(i+1)}(KS_i)=EP_{KU(i+1)}(H(R2_i+KC_i))$.这个本来就是给主机 S_{i+1},所以被动骗子 S_{i+1} 能够解密 K_i 得到 KS_i,但是 $KS_i=H(R2_i+KC_i)$,根据安全假设,被动骗子 S_{i+1} 无法获得 KC_i.同理,被动骗子 S_{i+1} 无法获取前面的主机的 $KC_0,KC_1,\ldots,KC_{i-1}$.

$D_i=D_{i-1}+EC_{KCi}(Data_i)$.主机 S_i 加密自己产生的数据 $Data_i$,采用的是常规加密算法 IDEA.根据安全假设,攻击者在没有密钥 KC_i 的情况下无法从 D_i 中获取 $Data_i$.同理,被动骗子 S_{i+1} 也无法获取前面的主机 $S_0,S_1,\ldots,S_{i-1}$ 所生成的数据.

$R_i=R_{i-1}+EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$.$R_i$ 所包含的信息是采用 Agent 所有者主机 S_0 的公钥 KU_0 加密的.根据安全假设,主机的私钥是安全的,所以攻击者无法解密 $EP_{KU0}(R1,R2,S_i,SIG_i(D_i,TS_i),TS_i)$,也就是说,无法从 R_i 中获取任何信息.

(4) 主动骗子

主动骗子假装遵守 IKCE 机制,但企图通过如篡改或删除前面主机的数据、重放等手段来破坏 IKCE 机制.

主动骗子无法做到篡改前面主机的数据而不被发现.假设主动骗子 S_i 出于某种目的将收到的 K_{i-1},D_{i-1},R_{i-1} 中的某个数据项 $EC_{KCj}(Data_j)$修改成其他数据,生成新的 D_i,再生成 $EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$.但是由于保护信息中的相应项 $EP_{KU0}(R1_j,R2_j,S_j,SIG_j(D_j,TS_j),TS_j)$是用发送者主机公钥加密的,主动骗子 S_i 无法修改.当回到所有者主机时,校验时检测到数字签名 $SIG_j(D_j,TS_j)$就能发现存在恶意篡改.

主动骗子无法做到截取掉某些数据项和保护信息而不被发现,如主动骗子 S_i 出于某种目的把它前面主机所产生的数据和保护信息都截掉,生成新的 D_i,再生成 $EP_{KU0}(R1_i,R2_i,S_i,SIG_i(D_i,TS_i),TS_i)$.但由于主动骗子 S_i 无法获取前面主机的各个 $KS_j(0<j<i)$,因此主动骗子 S_i 只能用他收到的 KS_{i-1} 和他自己生成的随机数 $R1_i$ 和 $R2_i$ 来生成 KC_i 和 KS_i,密钥链关系被破坏.当 Agent 回到所有者主机时,进行校验,用 KS_0 和主动骗子 S_i 生成的随机数 $R1_i$ 和 $R2_i$ 生成解密密钥,这与当初主动骗子 S_i 生成的 KC_i 和 KS_i 不同,解密 D_i 的结果不可解,从而检测出篡改.部分截取的情况可类似分析.

主动骗子也可能会保留目前结果,以后重新发出,但是 IKCE 机制采用时戳防止了重放攻击.同时,由于常规加密密钥 KC_i 是通过对随机数哈希生成的,而随机数也是一次一用的,也防止了重放攻击.

4 性能分析与对比

4.1 性能分析

IKCE 机制对移动 Agent 数据进行保护是以增加执行和传输时间为代价的.执行时间是由于在每个主机上的随机数生成、哈希、加密、签名等操作所引起的,而传输时间是由于移动 Agent 带有保护信息而引起移动 Agent 容量增加而引起的.

(1) 执行时间分析

执行时间取决于两个:在中间站点的执行时间 T_{TOT_INT} 和 Agent 所有者确认 Agent 完整性的时间 T_{TOT_SNT}.

在访问 N 个主机时,在中间主机上总的执行时间 T_{TOT_INT} 为

$$T_{TOT\_INT}=N(2T_{HASH}+2T_{RAND}+T_{CRYPT\_IDEA}+T_{CRYPT\_RSA}+T_{SIGN})$$
$$\approx N(T_{CRYPT\_IDEA}+T_{SIGN}).$$

其中:T_{HASH} 是哈希操作的时间;T_{RAND} 是生成随机数操作的时间;T_{CRYPT_IDEA} 是对主机所收集的数据进行 IDEA 加密的时间;T_{CRYPT_RSA} 是对保护信息和密钥链 K 进行 RSA 加密的时间;T_{SIGN} 是签名的时间.

哈希操作只是对主机所产生的固定大小的 128 位的随机数进行的.RSA 加密操作是对小容量的信息进行的.随机数生成采用的是普通的随机数生成算法.因此,T_{HASH},T_{RAND},T_{CRYPT_RSA} 与 T_{CRYPT_IDEA} 相比可以忽略不计.T_{CRYPT_IDEA} 依赖于应用数据,即中间主机所收集的数据的不同容量.当数据容量增长,T_{CRYPT_IDEA} 也会增长,根据不同的应用会有所变化.

当 Agent 返回,Agent 所有者要确认移动 Agent 数据的完整性.执行时间 T_{TOT_SNT} 为

$$T_{TOT\_SNT}=N(2T_{HASH}+T_{DECRYPT\_IDEA}+T_{DECRYPT\_RSA}+T_{VER})$$
$$\approx N(T_{DECRYPT\_IDEA}+T_{VER}).$$

其中:T_{HASH} 是哈希操作的时间;$T_{DECRYPT\_IDEA}$ 是对主机所收集的数据进行 IDEA 解密的时间;$T_{DECRYPT\_RSA}$ 是对保护信息进行 RSA 解密的时间;T_{VER} 是确认签名的时间.

(2) 传输时间分析

假设传输时间随着 Agent 容量的大小线性地增长[2].Agent 的在两个站点之间的传输时间 T_{TX} 由以下组成:$T_{TX}=aD_{CID}+bD_{AD}+cD_{PD}$.其中:$D_{CID}$ 是移动 Agent 的代码和初始数据等不变部分的容量大小;D_{AD} 是移动 Agent 在主机上所收集数据的容量大小;D_{PD} 是移动 Agent 在主机上保护信息的容量大;a,b,c 是常数.

当访问 N 个主机时,移动 Agent 在每个主机上运行后,所增加的容量是:在主机上新收集的数据和保护信息.总的传输时间 T_{TOTTX} 可以表述为 $T_{TOTTX}=NaD_{CID}+b\sum_0^N\Delta D_{AD}+c\sum_0^N\Delta D_{PD}$.

4.2 与相关工作的对比

IKCE 机制与相关工作对比如下:

(1) 与 TPE 方法[6]相比,IKCE 机制是一种基于检测的方法,属于软件保护方案.该方法不需要专用的硬件,开放性和扩展性比较好.

(2) 与 TTP 方法[3]相比,IKCE 机制的计算不借助于第三方实体,既没有由于移动 Agent 迁移到第三方实体而带来的时间消耗,又没有第三方实体的瓶颈问题.

(3) 与 Co-Signing 方法[5]相比,IKCE 机制的节点交互少,减少了由于节点交互带来的性能损失.

(4) 与 MH 协议[2]相比,IKCE 机制的思想也是在数据之间建立一种关联关系以达到保护移动 Agent 数据的目的.不同的是,MH 协议采用路径哈希链,而 IKCE 机制是对加密密钥建立关联关系.MH 协议采用公钥加密来实现数据保护;IKCE 机制采用的是常规加密.一般来说,常规加密比公钥加密速度快,这在加密大数据时尤为明显.因此,IKCE 机制更加适用于移动 Agent 在中间主机收集的数据量比较大的应用场合,如数据安全性要求高且数据量比较大的电子商务应用、信息服务、网络管理等.

5 实 验

我们在 IBM 公司的 Aglet 平台中进行实验[8],测试环境为 4 台 P4 1.7G,256M 内存,Window2000 操作系统的计算机.由于美国出口控制法对于加密算法的控制,使得 Java 运行环境的 JCE(Java cryptography extension)的部分加密算法只能在美国和加拿大使用.因此,我们采用了全球免费可用的 JCE 替代产品 Cryptix JCE 来实现 IKCE 机制.

实验分为两组进行:一组代表在信任环境中,未采用保护机制,Agent 在主机之间迁移,只保存一个 Agent 认为最好的数据;另一组代表在不信任的环境中,Agent 在主机之间迁移,采用 IKCE 机制来保存每个中间主机所提供的数据,在 Agent 所有者主机进行完整性检查.为实验方便,每个中间主机提供相同大小的数据量.实验结果如图 4、图 5 所示.

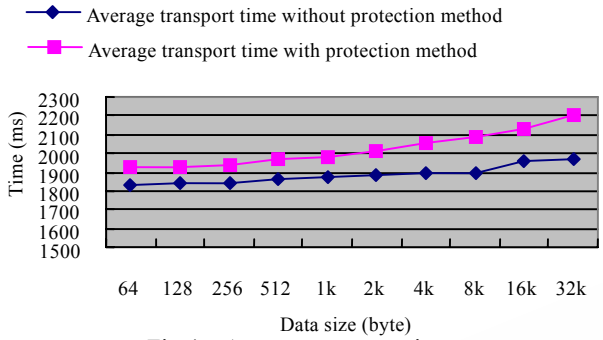

Fig.4 Average transport time

图 4 平均传输时间

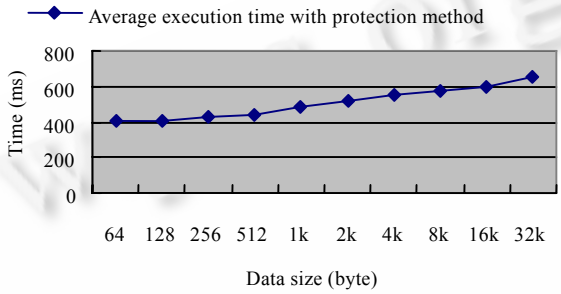

Fig.5 The adding average execution time

图 5 增加的平均执行时间

图 4 对比了未采用 IKCE 机制和采用 IKCE 机制的平均传输时间.从图中可见,采用 IKCE 机制后,传输时间有所增加,但不是很多,可以满足实际应用的需要.图 5 记录了在采用 IKCE 机制的情况下所增加的平均执行时间.在数据量较小(1k)的时候,本实验结果为 480ms,而文献[2]中的实验结果为 320ms,然而文献[2]的实验结果只是测试了完整性,没有测试机密性.在数据量较大(32k)的情况下,本实验所增加的平均执行时间比小数据情况下只增加了 170ms,而文献[2]没有对较大数据量进行实验.

6 总 结

本文针对特定的安全问题——移动 Agent 数据保护问题进行了研究,提出了 IKCE 机制.研究表明,IKCE 机制虽然牺牲了一定的速度和效率,但对于移动 Agent 数据保护是可行的.然而,在移动 Agent 系统安全性这一领域还存在着许多安全问题亟待研究,只有解决了这些安全问题,才能促进移动 Agent 系统的广泛应用.

References:

[1] Jansen W. Countermeasures for mobile Agent security. Computer Communications, 2000,23(10):1667−1677.

[2] Corradi A, Montanar R. Mobile Agent protection in the Internet environment. In: Gibson RM, ed. Proc. of the 23th Annual Int'l Computer Software and Applications Conf. Los Alamitos: IEEE Computer Society Press, 1999. 80−85.

[3] Corradi A, Cremonini M. Mobile Agents integrity for electronic commerce applications. Information Systems, 1999,24(6): 519−533.

[4] Karjoth G, Asokan N, Glc C. Protecting the computation results of free-roaming Agents. In: Rothermel K, Hohl F, eds. Mobile Agents: 2nd Int'l Workshop. London: Springer-Verlag, 1998. 195−207.

[5] Cheng JSL, Victor KW. Defenses against the truncation of computation results of free-roaming Agents. In: Deng RH, Qing SH, Bao F, Zhou JY, eds. Information and Communications Security. London: Springer-Verlag, 2002. 1−12.

[6] Wilhelm G, Staamann M. A pessimistic approach to trust in mobile Agent platforms. IEEE Internet Computing, 2000,4(5):40−48.

[7] Stalling W. Network Security Essentials: Applications and Standards. New York: Prentice Hall, 1999. 119−123.

[8] Karjoth G, Lange DB, Oshima M. A security model for aglets. IEEE Internet Computing, 1997,1(4):68−77.