移动Agent中一次代理签名体制的安全性分析*)

傅晓彤 张 宁 肖国镇

(西安电子科技大学 综合业务网国家重点实验室信息保密研究所 西安710071)

摘 要 移动Agent(Mobile agent)在电子商务中具有广泛的应用。H. Kim等人提出的为确保在移动Agent中进行秘密计算的安全性的一次代理签名体制,使得移动Agent协议具备了许多更好的特点,但是它不满足抵抗伪造攻击的性质。我们通过对该体制进行密码学分析,给出了一个成功的伪造攻击,在这种攻击下,一个不诚实的客户可以成功地假冒服务商对伪造的商品报价生成一个有效的一次代理签名。

关键词 安全性分析,一次代理签名,代理签名,移动Agent

Security Analysis of the One-time Proxy Signature Scheme Used in Mobil Agents

FU Xiao-Tong ZHANG Ning XIAO Guo-Zhen

(Information Security and Privacy Institute, National Key Lab of ISN Xidian University, Xi'an 710071)

Abstract It is found that the one-time proxy signature scheme proposed by Kim et al used in mobile agent to ensure the secret computation with secrests is not, in fact, secure against the forgery attack, as claimed. We cryptanalyzed the scheme, then a successful forgery be introduced. It is showed that a dishonest customer can successfully forge a valid one-time proxy signature by impersonating the server.

Keywords Cryptanalysis, Proxy signature, One-time proxy signature, Mobile agent

1 引言

移动Agent(Mobile agent)是一个自动控制的软件实体,它可以在不同的操作环境中转换使用。这种代理能够提供低带宽连接和异步通信模式,并且能够较好地支持在不同应用环境中的运行操作。这一特征使得移动Agent在电子商务中具有很高的应用价值。例如一个移动Agent能够代表客户在网络中有序地查找符合客户要求的商品或服务信息,一旦发现符合客户要求的商家报价,就可以代表客户和商家进行谈判并签名定单。移动Agent需要在运算平台上通过与其它Agent交互从而最终完成用户所指派的任务,协作性是移动Agent的基本要求,因此我们必须为其系统提供可靠并且有效的通信机制,保障移动Agent对周围环境的感知以及基本的通信需求。H. Kim, J. Baek, B. Lee, 和 K. Kim 在文[1]中首次提出了一次代理签名体制的概念,取代了在移动Agent中进行秘密计算时惯常采用的不可分签名技术。这一改进使得移动Agent协议具备了许多更好的特点,如较高的运行效率、消息长度的缩短、对被篡改移动代码的检测能力的增强,以及有效地控制了服务商的签名权限等等。但是,通过分析发现文[1]中给出的一次代理签名方案在不诚实客户的伪造攻击下是不安全的。我们给出了一个成功的伪造攻击,在这种攻击下,一个不诚实的客户可以成功地假冒服务商对伪造的商品报价生成一个有效的一次代理签名。

一般的代理签名体制就是一个用户,称为原始签名人,将其数字签名权利委托给另外一个用户,称为代理签名人,这样,代理签名人就可以代表原始签名人对消息进行代理签名。代理签名的基本方法是原始签名人对委托信息(一般是代理签名人的身份信息或其他有效的委托信息)生成一个签名,并交其秘密地交给代理签名人,代理签名人直接使用该签名作为代理私钥,或者用该签名生成一个代理私钥进行代理签名。这样,代理签名人就可以应用代理签名,验证者首先应用公开信息计算出代理签名公钥,然后按照进行代理签名时所使用的签名方案所对应的验证方法验证签名的有效性。代理签名体制具有广泛的应用性,如在电子货币系统中[2],电子商务的移动代理中[3],移动通信[4],格计算,全球分布式网络以及分布式计算等系统中都有对代理签名体制的应用。

本文第2节具体分析了应用于移动Agent中的一次代理签名体制;第3节给出一个成功的伪造攻击;最后是小结。

2 一次代理签名

我们首先回顾Kim等人在文[1]中给出的一次代理签名方案。一次性是使用故障-停止式签名的一次性性质来限制代理签名人的签名权限,即商家的签名权限。该方案实际上是代理签名体制和故障-停止式签名体制的一个结合。设系统参数由代理中介或可信第三方生成,公钥为 p,q,α,β,私钥为 a。$h(\cdot)$是一个无碰撞hash函数。(x_C,y_C,y'_C)和(x_S,y_S)分别表示客户 C 和服务商 S 的公私密钥对。req_C 表示客户的委任状,其中包括商品特征描述、最高出价以及委任日期等信息,bid_S 是商家的报价信息。

初始化过程(可信第三方 T 执行):

1)随机选择 $p,q,\alpha\in_R Z_q^*$

2)选择 $a\in_R Z_q^*$ 作为秘密密钥

3)计算 $\beta=\alpha^a \bmod p$

4)公开 p,q,α 和 β

代理签名密钥生成过程:

1)客户 C 在本地环境中执行以下步骤生成委托密钥对

*)基金项目:国家自然科学基金重大项目(90104005)。

(rc_1, rc_2, rc_3, rc_4),(Sc_1, Sc_2, Sc_3, Sc_4)。

a)随机选择 kc_1, kc_2, kc_3 和 $kc_4 \in_R Z_q^*$

b)计算 rc_1, rc_2, rc_3 和 rc_4,其中

$rc_i = \alpha^{kc_i} \bmod p, i \in \{1,3\}, rc_j = \beta^{kc_j} \bmod p, j \in \{2,4\}$

c)计算 Sc_1, Sc_2, Sc_3 和 Sc_4,其中

$Sc_i = x_C h(C, req\_C, rc_i) + kc_i \bmod p, i \in \{1,2,3,4\}$

客户将信息$(C, rc_1, rc_2, rc_3, rc_4, Sc_1, Sc_2, Sc_3, Sc_4, req_C)$加载于移动 Agent,令其在网络中运行,当移动 Agent 搜索到一个商家,其报价信息或商品质量等符合客户要求时,就在这个服务商所处的环境中执行定单操作。

2)商家 S 计算 $msg = h(S, C, req\_C, bid\_S)$,并根据客户的代理授权信息生成代理签名密钥。接着执行以下步骤,使用代理密钥对消息 msg 进行签名

a)首先,如下验证客户的授权密钥:

$\alpha^{Sc_i} = y_C^{h(C, req\_C, rc_i)} rc_i, i \in \{1,3\}$

$\beta^{Sc_j} = y_C'^{h(C, req\_C, rc_j)} rc_j, j \in \{2,4\}$

如果上面两个等式都成立,那么商家进行以下步骤:

b)计算代理签名密钥 $x_{P_1}, x_{P_2}, x_{P_2}$ 和 x_{P_4}

$x_{P_i} = Sc_i + x_{S_i} h(C, req\_C, rc_i), i \in \{1,2,3,4\}$

c)计算代理签名公钥 β_1 和 β_2

$\beta_1 = \alpha^{x_{P_1}} \beta^{x_{P_2}} \bmod p$,

$\beta_2 = \alpha^{x_{P_3}} \beta^{x_{P_4}} \bmod p$

d)用代理签名密钥 $x_{P_1}, x_{P_2}, x_{P_2}, x_{P_4}$,对消息 msg 进行签名,生成代理签名 $\sigma_{1,m}$ 和 $\sigma_{2,m}$

$\sigma_{1,m} = x_{P_1} + msg \cdot x_{P_3} \bmod q$,

$\sigma_{2,m} = x_{P_2} + msg \cdot x_{P_4} \bmod q$

e)商家 S 通过移动 Agent 将$(\sigma_{1,m}, \sigma_{2,m})$和$(S, C, bid_S)$,$(\beta_1, \beta_2)$以及消息 msg 送给客户 C。

3)用户 C,代理中介或可信第三方进行如下操作验证接收到的数据$(\sigma_{1,m}, \sigma_{2,m})$和$(S, C, bid_S)$,$msg$,$(\beta_1, \beta_2)$:

a)计算 $m = h(S, C, req\_C, bid\_S)$,检验 $m = msg$ 是否成立;

b)验证公钥的有效性

$\beta_1 = (y_C y_{S_1})^{h(C, req\_C, rc_1)} (y_C' y_{S_2})^{h(C, req\_C, rc_2)} rc_1 rc_2 \bmod p$

$\beta_2 = (y_C y_{S_3})^{h(C, req\_C, rc_3)} (y_C' y_{S_4})^{h(C, req\_C, rc_4)} rc_3 rc_4 \bmod p$

c)计算 v_1 和 v_2

$v_1 = \beta_1 \beta_2^{msg} \bmod p$,

$v_2 = \alpha^{\sigma_{1,m}} \beta^{\sigma_{2,m}} \bmod p$

如果 $v_1 = v_2$,客户判定接收到的$(\sigma_{1,m}, \sigma_{2,m})$是一个有效的代理签名。接受报价信息$((S, C, bid_S), msg$,为有效的报价。

3 安全性分析

一个不诚实的客户 C 可以对假消息 msg' 生成一个伪造的签名,即试图对伪造的报价信息 bid'_S 生成验证有效的代理签名。为生成伪造代理签名,C 首先伪造代理签名密钥 $\tilde{x}_{P_1}$,$\tilde{x}_{P_2}, \tilde{x}_{P_2}$ 和 $\tilde{x}_{P_4}$。

3.1 伪造代理签名密钥

不诚实的客户 C 选择 rc_1, rc_2, rc_3 和 rc_4 如下:

$$\begin{cases} rc_1 = y_{S1}^{-h(C, req\_C, rc_1)} \\ rc_2 = y_{S2}^{-h(C, req\_C, rc_2)} \\ rc_3 = y_{S3}^{-h(C, req\_C, rc_3)} \\ rc_4 = y_{S4}^{-h(C, req\_C, rc_4)} \end{cases}$$

计算$(\tilde{x}_{P_1}, \tilde{x}_{P_2}, \tilde{x}_{P_3}, \tilde{x}_{P_4}, \tilde{\beta}_1, \tilde{\beta}_2)$:

$$\begin{cases} \tilde{x}_{P_1} = x_C h(C, req\_C, rc_1) \\ \tilde{x}_{P_2} = x_C h(C, req\_C, rc_2) \\ \tilde{x}_{P_3} = x_C h(C, req\_C, rc_3) \\ \tilde{x}_{P_4} = x_C h(C, req\_C, rc_4) \end{cases}$$

$$\begin{cases} \tilde{\beta}_1 = \alpha^{x_C h(C, req\_C, rc_1)} \beta^{x_C h(C, req\_C, rc_2)} \\ \tilde{\beta}_2 = \alpha^{x_C h(C, req\_C, rc_3)} \beta^{x_C h(C, req\_C, rc_4)} \end{cases}$$

$(\tilde{x}_{P_1}, \tilde{x}_{P_2}, \tilde{x}_{P_3}, \tilde{x}_{P_4}, \tilde{\beta}_1, \tilde{\beta}_2)$即为有效的代理密钥对,这是因为:

$$\begin{aligned} \tilde{\beta}_1 &= (y_C y_{S_1})^{h(C, req\_C, rc_1)} (y_C' y_{S_2})^{h(C, req\_C, rc_2)} rc_1 rc_2 \bmod p \\ &= y_C^{h(C, req\_C, rc_1)} y_C'^{h(C, req\_C, rc_2)} \\ &= \alpha^{x_C h(C, req\_C, rc_1)} \beta^{x_C h(C, req\_C, rc_2)} \end{aligned} \quad (1)$$

$$\begin{aligned} \tilde{\beta}_2 &= (y_C, y_{S_3})^{h(C, req\_C, rc_3)} (y_C' y_{S_4})^{h(C, req\_C, rc_4)} rc_3 rc_4 \bmod p \\ &= y_C^{h(C, req\_C, rc_3)} y_C'^{h(C, req\_C, rc_4)} \\ &= \alpha^{x_C h(C, req\_C, rc_3)} \beta^{x_C h(C, req\_C, rc_4)} \end{aligned} \quad (2)$$

这样,一个不诚实的客户 C 就能应用伪造的代理签名密钥$(\tilde{x}_{P_1}, \tilde{x}_{P_2}, \tilde{x}_{P_2}, \tilde{x}_{P_4})$假冒服务商对伪造的报价信息进行代理签名。

3.2 伪造代理签名

1. 签名过程:

C 使用代理签名密钥 $\tilde{x}_{P_1}, \tilde{x}_{P_2}, \tilde{x}_{P_2}$ 和 $\tilde{x}_{P_4}$,对消息 msg' 生成代理签名,这里 $msg' = hash(S, C, req\_C, bid'\_S)$,$bid'\_S$ 是一个对商家 S 的伪造报价消息。$\sigma'_{1,m}$ 和 $\sigma'_{2,m}$ 的计算如下:

$\sigma'_{1,m} = \tilde{x}_{P_1} + msg' \cdot \tilde{x}_{P_3} \bmod q$

$\sigma'_{2,m} = \tilde{x}_{P_2} + msg' \cdot \tilde{x}_{P_4} \bmod q$

$(\sigma'_{1,m}, \sigma'_{2,m})$即为 C 假冒 S 生成的代表签名。

2. 验证过程:

代理中介或可信第三方通过以下方式验证签名$(\sigma'_{1,m}, \sigma'_{2,m})$:

a)计算 $m = hash(S, C, req\_c, bid'\_S)$,检验 $m = msg'$ 是否成立

b)验证公钥的有效性

$\tilde{\beta}_1 = (y_C y_{S_1})^{h(C, req\_C, rc_1)} (y_C' y_{S_2})^{h(C, req\_C, rc_2)} rc_1 rc_2 \bmod p$

$\tilde{\beta}_2 = (y_C y_{S_3})^{h(C, req\_C, rc_3)} (y_C' y_{S_4})^{h(C, req\_C, rc_4)} rc_3 rc_4 \bmod p$

显然,它们是满足方程(1)和(2)的,验证有效。

c)计算 v_1 和 v_2

$$\begin{aligned} v_1 &= \tilde{\beta}_1 \tilde{\beta}_2^{msg'} \bmod p \\ &= y_C^{h(C, req\_C, rc_1)} y_C'^{h(C, req\_C, rc_2)} \\ &\quad (y_C^{h(C, req\_C, rc_3)} y_C'^{h(C, req\_C, rc_4)})^{msg'} \bmod p \end{aligned}$$

$$\begin{aligned} v_2 &= \alpha^{\sigma_{1,m}} \beta^{\sigma_{2,m}} \bmod p \\ &= (\alpha^{\tilde{x}_{P_1}} \alpha^{msg' \tilde{x}_{P_3}}) (\beta^{\tilde{x}_{P_2}} \beta^{msg' \tilde{x}_{P_4}}) \bmod p \\ &= \alpha^{x_C h(C, req\_C, rc_1)} \beta^{x_C h(C, req\_C, rc_2)} (\alpha^{x_C h(C, req\_C, rc_3)} \\ &\quad \beta^{x_C h(C, req\_C, rc_4)})^{msg'} \\ &= y_C^{h(C, req\_C, rc_1)} y_C'^{h(C, req\_C, rc_2)} (y_C^{h(C, req\_C, rc_3)} \\ &\quad y_C'^{h(C, req\_C, rc_4)})^{msg'} \bmod p \end{aligned}$$

我们得到 $v_1 = v_2$。$(\sigma'_{1,m}, \sigma'_{2,m})$是一个有效的代理签名,其生成者被认为是合法代理签名人即商家 S。

因此,代理中介或可信第三方将判断$(\sigma'_{1,m}, \sigma'_{2,m})$是商家对消息 msg' 的有效代理签名。即,对于伪造的报价信息 bid'_S,不诚实客户 C 成功地伪造了一个有效的一次代理签名。而且,他可以要求商家对伪造的显然对他有利的报价信息 $bid'\_$

S 负责。

小结 本文主要分析讨论了 H. Kim 等人提出的移动 Agent 中的一次代理签名体制所存在的安全性问题。该体制应用故障-停止签名保证代理签名的一次性要求。但是它不满足抗伪造攻击这一性质。而在移动 Agent 中，为保证代理签名以及秘密计算的安全性需求，要求系统能够抵抗伪造攻击。通过我们的分析，可以看到 H. Kim 等人提出的一次代理签名方案中不诚实的原始签名人即不诚实的客户可以成功地伪造有效的代理签名。因此该方案需要进一步改进以保证移动 Agent 的安全运行。

参 考 文 献

1 Kim H, Baek J, Lee B, Kim K. Secret Computation with Secrets for Mobile Agent using One-time Proxy Signature. In: Cryptography and Information Security 2001, 2001

2 Okamoto T, Tada M, Okamoto E. Extended proxy signatures for smart cards. In: Information Security Workshop (ISW'99), LNCS 1729, Springer-Verlag, 1999. 247～258

3 Lee B, Kim H, Kim K. Strong proxy signature and its applications. In: Proc. of SCIS, 2001. 603～608

4 Park H-U, Lee I-Y. A digital nominative proxy signature scheme for mobile commuinications. In: Information and Communications Security (ICICS'01), LNCS 2229, Springer-Verlag, 2001. 451～455

5 Sun H-M, Hsieh B-T. On the Security of Some Proxy Signature Schemes. Available at: http://eprint.iacr.org/2003/068

6 Yi Lijiang, Bai Guoqiang, Xiao Guozhen. Proxy multi-signature scheme. Eletronics Letters, 2000, 36: 527～528

（上接第39页）

第一种是 Sprocket 语言，它是一种很类似于 C 的高级语言，但是 Sprocket 取消了像 C 中指针这样的一些威胁安全的结构，增加了一些对 MIB 访问接口和 Smart 包等用于网络和网络管理新的特征。下面的代码定义了一个 Packet 和 address 类型的变量，并指定了一个包的目的地址。

```
packet p;
address a;
a = p.destination();
```

第二种是 Spanner 语言，它是一个基于 CISC 的汇编语言，Spanner 的程序通过编译成相应的代码，再通过编译成一个压缩的独立于机器的二进制代码，并将它注入到 Smart 包的程序段中。Spanner 所提供的原语可以实现 Smart 包的发送与传输，由于 Spanner 是为 Smart 包传输路径上路由器执行的程序，因此利用路由器可以鉴别 IP 选项，程序可以控制 Smart 包中的代码是交付节点执行还是沿下一个节点传送。

4 Smart 包的鉴定与授权

智能包的安全结构由两部分组成：鉴定和授权。使用 ANEP 的鉴定机制可以鉴定数据包的来源和数据的完整性。鉴定可选项规定了对数据报文签名了的实体，它包含签名的类型、认证的类型、标识的长度、认证和在有效负载域中的数据都被签名。包含在 Smart 包中的授权是安全结构的第二部分，有两个主要的授权因素：

(1)控制 MIB 访问，它包含 MIB 的视图与读/写权限。

(2)在虚拟主机中进行时间环境限制了每次调用被发送数据包的最大数量，能够被分配的最大内存数，每次调用能被执行的最大指令数。

用户标识的获取来自于鉴定组件，用来获得在数据库中的程序的资源限制，这些限制被设定在虚拟机当中，虚拟机实施这些限制在主机资源和对特权指令的访问控制上。

一个被鉴定的智能包携带了 X.509公钥认证来标识智能包的发送者。数据签名用来验证数据完整性，使用数字签名防止了恶意篡改数据包的这些部分。

一个重要的问题是如何选择智能包中的数据域进行鉴定与保护，鉴定信息不仅要验证智能包的来源，而且要验证数据包的数据完整性。

在节点收到一个 Smart 包时，通过以下进程实现鉴定和授权：

(1)如果一个鉴定可选项并没有出现在 Smart 包中，该报文不能被鉴定。因此，它被授予在授权数据库中对任何实体最小的访问权限。

(2)如果鉴定可选项存在，公钥认证生效，这个生效处理过程需要认证发行机构的公钥。

(3)以下不进行 ANEP 报文的签名检验：ANEP 数据包的长度是否为零及 Smart 包的载荷域。

(4)如鉴定失败，则数据包被丢弃。如果鉴定无法完成(如认证有效期的超时)，数据包被转发。如果验证成功，数据包进入鉴定阶段。

(5)包含在认证中的标识被用于寻找授权数据库的访问权限。数据库中的记录标识着任何一个程序应该具有的约束或特权。

要实现上述的服务，就需要增加 Smart 包中证书域所占用的空间大小。例如，X.509公钥认证包含有长度约为400 bytes。另外一个问题就是验证证书所需要的时间与资源，很多的减少处理证书时间的方法都是通过增加数据包中的附加验证信息来实现的，因此这些方法都有一定的局限性。

结束语 主动网络管理体现了主动网络的思想，将一部分网络管理功能动态地分布在主动节点上，充分利用了主动节点的计算能力，使节点能够自动发现、解决问题，从而极大地优化了网络管理。

本文探讨了一种基于 Smart 包的主动网络管理模式，并对其结构、性能与管理机制进行了分析。利用 Smart 包来实现网络管理时，网络的基础结构要足够地简化以适应 Smart 包的传送，网络要足够强壮以适应远程控制，还要足够灵活以允许远程授权和委托。该管理方案满足了主动网络中节点的灵活性和主动应用的扩展性要求，是一种较为理想的网络管理方案，相信它会在将来的网络管理中发挥巨大的作用。

参 考 文 献

1 Fry M, Ghosh A. Application Layer Active Networking. Computer Networks, 1999, 31(7): 655～667

2 Marshall I W, et al. Application Layer Programmable Internet work Environment. British Telecom. Technol. J., 1999, 17(2): 82～94

3 Fatta G D, Gaglio S, Re G L, Ortolani M. Adaptive Routing in Active Networks. IEEE Openarch 2000, Tel Aviv Israel, March 2000

4 Schwartz B, et al. Smart Packets: Applying Active Networks to Network Management. ACM Transactions on Computer Systems, 2000, 18(1): 67～88

5 Wetherall D, Legedza U, Guttag J. Introducing New Internet Services: Why and How. IEEE Networks Magazine, May/June 1998

6 Marshall I W, et al. Active management of multiservice networks. In: Proc. IEEE NOMS, 2000. 981～983

7 Fatta G D, Re G L. Active Networks: an Evolution of the Internet. In: Proc. of AICA2001 - 39th Annual Conf. Cernobbio, Italy, Sept. 2001