第38卷第1期
2010年1月

同济大学学报(自然科学版)
JOURNAL OF TONGJI UNIVERSITY(NATURAL SCIENCE)

Vol.38 No.1
Jan. 2010

文章编号：0253-374X(2010)01-0135-06

DOI:10.3969/j.issn.0253-374x.2010.01.025

用于复合服务的白盒加密算法

史　扬[1]，林　杰[1]，曹立明[2]

(1.同济大学 经济与管理学院，上海 201804；2.同济大学 电子与信息工程学院，上海 201804)

摘要：针对基于移动Agent的服务复合的脆弱性，给出一种白盒加密算法.通过引入有限域上的分块矩阵乘法和带输入输出变换的安全加法器，算法将密钥隐藏在一系列的数据表中，由此实现了基于加密函数的安全数据加密，能够应对白盒攻击环境下密钥泄漏的安全风险.该算法代码体积较小，适合于移动Agent在非固定式服务复合时使用.

关键词：白盒；复合服务；加密算法

中图分类号：TP 309.7；C 931.6　　**文献标识码**：A

A White-Box Encryption Algorithm for Services Composition

SHI Yang[1]，*LIN Jie*[1]，*CAO Liming*[2]

(1. College of Economics and Management，Tongji University，Shanghai 201804，China；2. College of Electronics and Information Engineering，Tongji University，Shanghai 201804，China)

Abstract：A white-box encryption algorithm is proposed for eliminating vulnerabilities of services composition based on mobile agents. By using multiplication of block matrix on finite fields and secure addition with input and output transformation，the key is hidden in a set of tables. Thus the algorithm controls the risk of key compromise in white-box attack context and provides an approach for encrypting data securely by computing with encrypted functions. The size of implementation of this algorithm is small and therefore suitable for non-fixed composition of services.

Key words：white-box；services composition；encryption algorithm

基于移动Agent的复合服务等分布式计算的应用日益广泛，其中的安全问题也日渐突出.笔者提出一种白盒加密算法，来解决此类分布式计算中的安全加密问题.

Web服务复合的实现方法主要有三类[1]：探索式(explorative)、半固定式(semi-fixed)与固定式(fixed).探索式和半固定式由于部分语义可以在执行期确定，成为构建基于服务的柔性信息系统的较好选择.

基于Agent的多域模型[1]是一种用于实现Web服务的复合和执行的构架.域分布在计算机网络上并由域管理者来控制，包括用户域和服务提供者域.当用户请求一个复合服务时，用户Agent代表用户和服务进行交互，并通过远程调用或迁徙后本地调用等方法，对服务进行复合.服务提供者域包括一个工作区和若干个服务门户，每一个门户代表一类或一个相应的服务，而工作区则用来接收迁徙而来的移动Agent.移动Agent通过和原先就部署在本地的代表服务提供者的Agent交互，实现对服务的调用和复合.

参照该模型，给出如图1所示的应用场景.在该场景中，假设某个企业需要采购一批原料，用户通过复合Web服务门户提出原料采购服务请求并给出价格、数量和型号等采购条件，系统将从用户Agent池中生成一个移动Agent实例.首先，通过企业内部远程和供应商管理服务交互获取可能的供应商列表，然后，开始向各个供应商的域迁徙并调用相应的销售服务进行询价操作，直至找到符合要求的原料并签署合同后，再迁徙到企业内的采购部门域，与订单管理服务交互完成其他相关的后续事务.

1　安全风险分析

白盒攻击环境(white-box attack context，WBAC)是指攻击者拥有适应性选择明文攻击的条

收稿日期：2008-08-29

基金项目：国家"八六三"高技术研究发展计划资助项目(2007AA04Z151)；国家自然科学基金重点资助项目(70531020)；新世纪优秀人才支持计划资助项目(NCET-06-0377)

作者简介：史　扬(1977—)，男，高级工程师，工学博士，博士后，主要研究方向为信息安全和柔性信息系统开发方法.
E-mail:cnshiyang@yahoo.com.cn

件,且对加密软件及其运行环境拥有完全的控制权[2-3].例如,对程序运行的二进制追踪、读取内存中的密钥和程序执行的中间结果、对软件进行任意的静态分析,以及通过改变子计算的结果来进行摄动分析,等等.

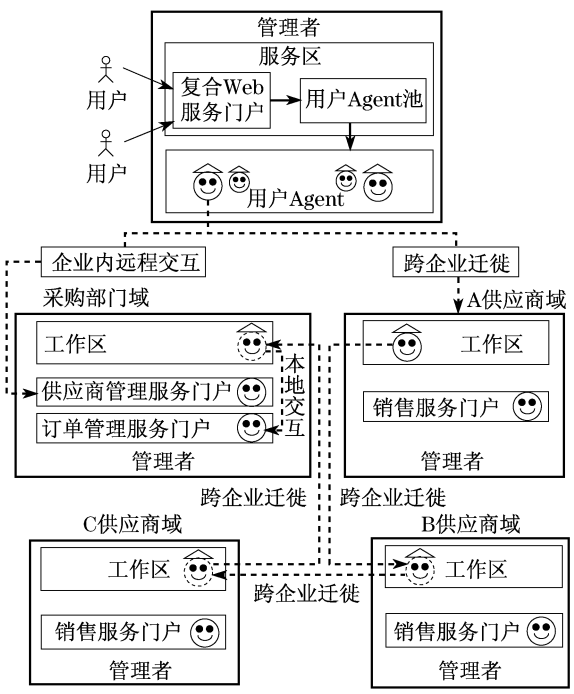

图1 基于多域模型的服务复合应用场景

Fig.1 Application scenario of services composition based on multi-domain model

移动Agent由于移动性和自治性,是实现探索式和半固定式的服务复合的有力工具.但是,安全问题却成为目前制约其应用的最大障碍.比如,在很多的实际应用中,由于Agent运行的物理环境受到他人的控制,处于白盒攻击环境之中,用于加密的密钥不能够由Agent来携带[4].以图1所示的复合服务为例,这是一个典型的存在非信任关系和利益冲突的白盒攻击环境.在询价操作中,每个供应商的报价都应对其他的供应商保密.这就要求进行服务复合的移动Agent能够在白盒攻击环境下安全地加密计算且不泄漏密钥.

针对此类脆弱性,可以将安全需求抽象为对基于加密函数计算[5]的需求:服务A有计算函数 f 的算法,服务B有数据 x 且愿意帮助服务A计算出 $f(x)$,但A不希望B知道 f,而且B在计算时也无须与A交互.

从软件实现的角度,则应针对此类脆弱性引入混淆变换[6-7].其实质是提供了一种转换机制,使转换后的程序(或其反编译结果)难以被理解,但具有相同的可观测行为.

基于对安全需求的分析,现提出一种白盒加密算法以控制复合服务的安全风险.该算法实现了对加密函数的加密计算,也可视为对加密函数代码混淆变换的产物,具有较小的体积和较快的计算速度.算法代码随Agent移动时,占用带宽较小,且可以在瘦客户端上高效运行.

2 一种白盒加密算法

2.1 Khazad对称加密算法

Khazad是一个有64位对称块的加密算法[8],密钥长度为128位,是新欧洲密码标准的候选方案之一.

Khazad是一个迭代对合的块加密算法,由一系列依赖于密钥的轮变换操作组成.这些操作的对象都是64 bit(比特)的数据块(可以视为 $GF(2^8)^8$ 中的元素),称为加密的"状态".其轮变换涉及如下三个层:

(1) 非线性层

函数 $\gamma:GF(2^8)^8\to GF(2^8)^8$ 可以看作对自变量中所有字节并行应用非线性置换盒 $S:GF(2^8)\to GF(2^8)$,$x\mapsto S[x]$ 所得,即 $\gamma(a)=b\Leftrightarrow b_i=S[a_i]$,$0\leqslant i\leqslant 7$. S 的设计原则之一是要求 S 为对合变换,即 $\forall x\in GF(2^8)$,$S[S[x]]=x$.显然,γ 也是对合变换.Khazad算法的 S 盒定义参见文献[8].

(2) 线性扩散层

线性扩散层的实质是线性变换,其对应的变换 θ 定义如下:

$$\theta:GF(2^8)^8\to GF(2^8)^8$$
$$\theta(a)=b\Leftrightarrow b=a\boldsymbol{H} \tag{1}$$

其中,矩阵 $\boldsymbol{H}$ 的定义参见文献[8].

容易验证,$\boldsymbol{H}$ 是对称阵,也是酉阵,所以,θ 是对合变换.

(3) 密钥作用层

密钥作用层对应的函数为 $\sigma[k]:GF(2^8)^8\to GF(2^8)^8$,$\sigma[k]$ 利用密钥 $k\in GF(2^8)^8$,对输入执行如下操作:

$$\sigma[k](a)=b\Leftrightarrow b_i=a_i\oplus k_i,\quad 0\leqslant i\leqslant 8 \tag{2}$$

将上述三层叠加起来,可得轮函数

$$\rho[k]:GF(2^8)^8\to GF(2^8)^8,\quad k\in GF(2^8)^8$$
$$\rho[k]\equiv\sigma[k]\circ\theta\circ\gamma \tag{3}$$

令 R 为算法的轮数(标准值 $R=8$),对密钥 K 即有

$$K_{\mathrm{K}}[K]=\alpha_R[K_0,\cdots,K_R]=$$
$$\sigma[K_R]\circ\gamma\left(\overset{r=R-1}{\underset{1}{\circ}}\rho[K_r]\right)\circ\sigma[K_0] \quad (4)$$

其中,轮密钥的生成算法参见文献[8].

2.2　一种新的白盒加密算法

此处给出一种新的白盒加密算法.该算法通过引入有限域上的分块矩阵乘法和带输入输出变换的安全加法器,将密钥隐藏在一系列的数据表中,实现了基于加密函数的安全数据加密,能够应对白盒攻击环境下密钥泄漏的安全风险.

记 $\boldsymbol{H}=(h_{i,j})\in\mathrm{GF}(2^8)^{8\times 8}$,则

$$\boldsymbol{H}=\begin{Bmatrix}h_1\\ \vdots\\ h_i\\ \vdots\\ h_8\end{Bmatrix},\{h_1,\cdots,h_8\}\subseteq\mathrm{GF}(2^8)^8,h_i=(h_{i1},\cdots,h_{i8}) \quad (5)$$

对 $b=a\boldsymbol{H}$,设 $a=(a_1,\cdots,a_8)$,$b=(b_1,\cdots,b_8)$,显然有

$$(b_1,\cdots,b_8)=(a_1,\cdots,a_8)\begin{pmatrix}h_1\\ \vdots\\ h_i\\ \vdots\\ h_8\end{pmatrix}=\sum_{i=1}^{8}a_i h_i=$$
$$\sum_{i=1}^{8}a_i(h_{i1},\cdots,h_{i8})=\sum_{i=1}^{8}(a_i h_{i1},\cdots,a_i h_{i8}) \quad (6)$$

其中

$$b_j=\sum_{i=1}^{8}a_i h_{ij}=[((a_i h_{i1})\oplus(a_i h_{i2}))\oplus((a_i h_{i3})\oplus(a_i h_{i4}))\oplus(((a_i h_{i5})\oplus(a_i h_{i6}))\oplus((a_i h_{i7})\oplus(a_i h_{i8}))] \quad (7)$$

由以上的分析,令 θ_i 为线性扩散层的一部分.定义如下:

$$\theta_i:\mathrm{GF}(2^8)\to\mathrm{GF}(2^8)^8$$
$$\theta_i(x)=x(h_{i1},\cdots,h_{i8})=(xh_{i1},\cdots,xh_{i8}) \quad (8)$$

显然,将 $\theta_1,\theta_2,\cdots,\theta_8$ 组合起来,就可以得到线性变换 θ.

令第 r 轮的第 i 个轮输出变换为 $\tau_{r,i}\circ\tau_{r,i}$ 在白盒加密算法的实现中被定义为 2 个 4 bit 置换的组合,这 2 个置换依次记作 $\tau_{1r,i}$ 和 $\tau_{2r,i}$.

令第 r 轮的第 i 个轮中间变换为 $\pi_{r,i}\circ\pi_{r,i}$ 也在白盒加密算法的实现中被定义为 16 个 4 bit 置换的组合,这 16 个置换依次记作 $\pi_{1r,i},\cdots,\pi_{16r,i}$.

定义白盒子轮函数如下:

$$\rho_{\mathrm{W}}[r,i,K]:\mathrm{GF}(2^8)\to\mathrm{GF}(2^8)^8$$
$$\rho_{\mathrm{W}}[r,i,K]=\pi_{r,i}\circ\theta_i\circ\gamma\circ\sigma_i[K_{r-1}]\circ\tau_{r-1,i}^{-1},\quad r=1,2,\cdots,7 \quad (9)$$
$$\rho_{\mathrm{W}}[8,i,K]=\pi_{8,i}\circ\theta_i\circ\sigma_i[K_8]\circ\gamma\circ\sigma_i[K_7]\circ\tau_{7,i}{}^{-1} \quad (10)$$

在实现中,每个函数 $\rho_{\mathrm{W}}[r,i,K]$ 被定义为 1 张 8 bit 输入 64 bit 输出的表,记此类表为 A 型表.

为将白盒子轮函数的输出叠加起来得到白盒轮函数,令

$$\psi_{\mathrm{W}}[r,i]:\mathrm{GF}(2^8)^8\to\mathrm{GF}(2^8)$$
$$\psi_{\mathrm{W}}[r,i](a_1,\cdots,a_8)=\sum_{i=1}^{8}a_i \quad (11)$$

如果直接计算 $\mathrm{GF}(2^8)^8$ 上的加法,将导致运算中间结果的泄漏,破坏安全性.所以,ψ_{W} 应当通过多次应用图 2 所示的结构来实现加法.其中,相邻的输入置换和输出置换对应互为逆变换.图 2 所示的运算组件构成一个函数,在实现中定义为 1 张 8 bit 输入、4 bit 输出的表.

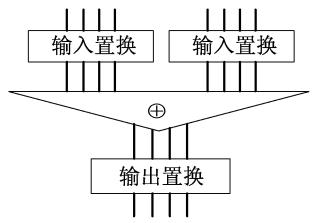

图 2　安全的加法器

Fig.2　Secure adder

进一步,可以按照图 3 中所示的方式定义函数 $\varphi_{\mathrm{W}}[r,j]:\mathrm{GF}(2^8)^8\to\mathrm{GF}(2^8)$,$r=0,1,\cdots,8$,$j=1,\cdots,8$.实际上,$\varphi_{\mathrm{W}}[r,j]$ 由两个如图 3 中所示的结构组成.在实现中,每个 $\varphi_{\mathrm{W}}[r,j]$ 按照图 3 所示的方式被定义为 15 张 8 bit 输入、4 bit 输出的表,记此类表为 B 型表.

设 τ_r 为 $\tau_{r,1},\cdots,\tau_{r,8}$ 的组合,π_r 为 $\pi_{r,1},\cdots,\pi_{r,8}$ 的组合,可得白盒 Khazad 加密函数 $K_{K,\mathrm{W}}[K]=\tau_8\circ\theta\circ K_K[K]\circ\tau_0^{-1}$;可得白盒 Khazad 加密函数的逆运算为白盒 Khazad 解密函数 $K_{K,\mathrm{W}}^{-1}[K]=\tau_0\circ K_K[K]^{-1}\circ\theta\circ\tau_8^{-1}$.显然,$I=K_{K,\mathrm{W}}^{-1}[K]\circ K_{K,\mathrm{W}}[K]$.

白盒 AES(advanced encryption standard,高级加密标准)加密器[3]可以看作对 AES 加密器混淆变换的结果[6-7],符合混淆变换的定义.这里给出的白盒加密算法虽然不能严格符合混淆变换的定义,但并不影响应用;给出的白盒 Khazad 算法的输出与原始的 Khazad 加密算法的输出是不同的,但是通过原始的 Khazad 解密算法,容易得到本加密算法对应的

解密算法.

如果移动Agent复合的某个服务需要使用其他服务加密计算的结果,则可以选用以下两种方法之一:

(1) 事先对密钥协商使用密钥交换技术,密钥交换的具体算法可根据需要选用[9].

(2) 调用公共的可信任的安全服务中有关解密的方法.公共安全服务的解密方法所使用的密钥由生成白盒加密器的服务提供,而公共安全服务是否提供解密结果,则可由加密者提供的安全策略来决定.

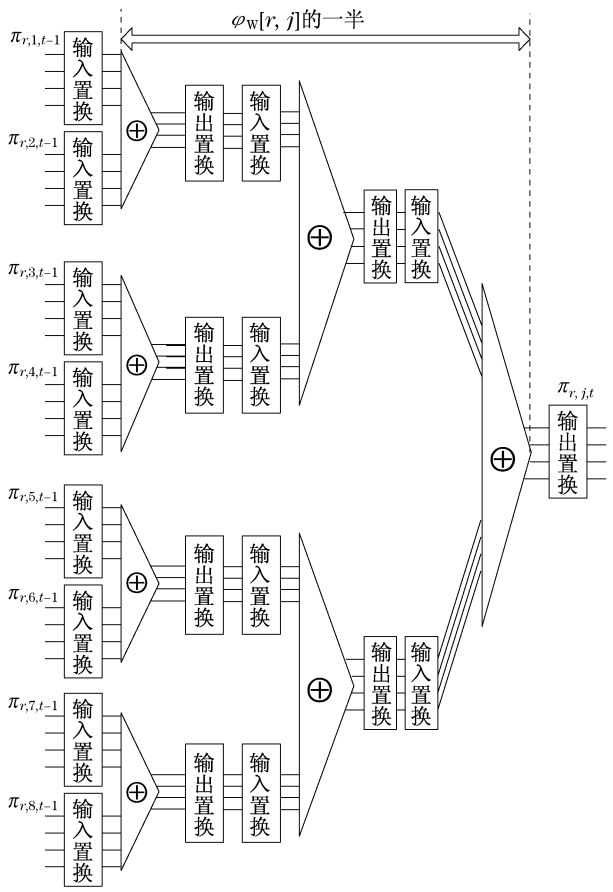

图3 $\varphi_w[r,j]$结构示意

Fig.3 Structure of $\varphi_w[r,j]$

3 算法分析和比较

3.1 算法的安全性

Chow,Eisen和Johnson等人给出了白盒多样性和白盒模糊性的概念,对白盒密码术的安全性进行初步分析[2].密钥空间为密码学算法的安全性评估提供了一个上界.类似地,如果编码"加密"了实现的步骤,统计可能的编码步骤也可以用于度量白盒密码的安全性.这称之为白盒多样性.某一类型的表的多样性即指其对应的可能的不同构造的数量.这一度量体现了实现的可变性.另一个更为重要的度量则是白盒模糊性.某一类型的表的白盒模糊性即指其中确定的某个表的可能的不同构造的数量.这一度量体现了某个表的具体解释的多样性.遗憾的是,白盒模糊性自提出以来,尚未发现有效的计算方法.

对于A型表,轮输出变换有$(2^4!)\times(2^4!)$种构造方法,轮中间变换有$(2^4!)^{16}$种构造方法,轮密钥有2^{64}种可能,其白盒多样性为$(2^4!)^2\times(2^4!)^{16}\times 2^{64}=(16!)^{18}\times 2^{64}$.

对于B型表,2个输入置换各有$(2^4!)$种构造方法,1个输出置换有$(2^4!)$种构造方法,其白盒多样性为$(2^4!)^2\times(2^4!)=(16!)^3>2^{130}$.

根据上述计算,算法的白盒多样性小于白盒AES算法,但其数值仍足以提供较好的安全性.

Billet,Gilbert和Ech-Chatbi给出了一种从白盒AES[2]的实现中提取密钥的方法[10],但SyncroSoft已经改进了设计并能够抵抗这种攻击[11].Jacob,Boneh和Felten提出了一种通过注入故障来对白盒DES(data encryption standard,数据加密标准)攻击的方法[12],但他们在同一篇论文中给出了相应的对策,且该方法不适用于对本算法攻击.Wyseur,Michiels和Gorissen等通过对一个混淆后的轮进行差分分析[13],从白盒DES[4]的实现中提取密钥.但是,因为白盒DES引入了两个额外的变换,这一方法并不能破解白盒DES加密的结果,同时,由于对T变换做了一些限制,不具有广泛应用的价值,也不能实现对本算法的攻击.Goubin,Masereel和Quisquater最近提出了一种新的攻击方法[14],能够攻击现有的各种白盒DES实现方法,包括Link和Neumann提供的改进后的实现方法[15].这一方法使用C语言编程实现后,进行了攻击测试,取得了较好的攻击效果.由于DES和AES在设计上有相当大的差异,白盒DES的实现方法与白盒AES的实现方法也有明显的不同,该方法不适合于攻击白盒AES,也不适合于攻击本算法.

将白盒Khazad加密器用于保护移动Agent的有限时间黑盒,看来是可行的,至少目前没有发现可以在较短时间内对其有效攻击的方法,且其中两种表的白盒多样性数值相当大.

3.2 代码体积和时间复杂度及比较

白盒AES算法[3]的输出与原始的AES算法的输出是相同的.这是本算法和白盒AES的不同之处.

本做法有利于减小加密器的体积.给出的白盒加密算法的实现体积分析如下:每一轮需要 8 个 A 型表,总共需要 8×8 个 A 型表.每个 A 型表需要 $2^8 \times 64$ bit $= 2^8 \times 8$ byte(字节) $= 2^{11}$ byte,总共为 $8 \times 8 \times 2^{11}$ byte $= 2^{17}$ byte = 128 kb.每一轮需要 8 个 $\varphi_w[r, j]$函数,每个 $\varphi_w[r, j]$函数需要 15 个 B 型表,总共需要 8×8×15 个 B 型表.每个 B 型表需要 $2^8 \times 4$ bit $= 2^7$ byte,总共为 120 kb.总体积为 248 kb,即 253 952 byte,仅为白盒 AES 所需的 770 048 byte 的 33%,更适用于移动 Agent.

根据上面的分析,还可以求出每次加密运算需要的查表操作次数.由于白盒加密算法唯一的计算就是查表,比较表 1 和表 2 中给出的查表次数数据,使用白盒 AES(1 次)加密 128 bit 数据,较之使用本算法(2 次,每次 64 bit)加密 128 bit 所要的查表计算次数多 50%以上,由此即可估计出本算法的运算速度较白盒 AES 方法快[3].在查表时,耗费的时间与最终查出的具体内容无关,所以,只需针对每一种表格的所有可能输入进行实验即可.下面两个表格给出的实测耗时,就是对所有可能输入消耗的时间计算的算术平均值.从实测数据也可以看出,本算法的使用对实际系统影响较小.由于 Web 服务和移动 Agent 平台大多使用 Java 语言开发,故使用 Java 编写测试程序,运行环境为 JDK 1.5,测试所用计算机的 CPU 为奔腾 1.73 G.

表 1　白盒 AES 每次执行所需查表次数

Tab.1　Number of lookups during each execution of white-box AES

表格类型	表格体积	查表次数	实测每次耗时/ns
T-box	8×32	288	6.3
MixColumns add	8×4	1 728	5.5
IDM& T-box/ODM	8×128	128	7.0
IDM & ODM	8×4	960	5.5
总计		3 104	

表 2　本算法每次执行所需查表次数

Tab.2　Number of lookups during each execution of this algorithm

表格类型	表格体积	查表次数	实测每次耗时/ns
A	8×64	64	6.5
B	8×4	896	5.5
总计		960	

根据实验结果可知,在如前所述的实验环境中使用白盒 AES 算法,每加密 128 bit 数据,约需 17.5 ns,而本算法只需约 10.7 ns.

本算法的解密算法与 Khazad 的解密算法非常相似,使用时不会对系统的性能带来显著影响.

3.3　其他相关工作比较

目前,关于混淆变换的研究成果(如文献[6-7]等),均未给出对于对称加密方法的安全混淆;而基于加密函数计算的研究(如文献[5-6]等),均未给出加密"对称加密算法"的方法.

4　结语

为了控制移动 Agent 在白盒攻击环境下密钥泄漏的安全风险,笔者提出了适用于基于移动 Agent 的复合 Web 服务的安全加密方法.该方法以 Khazad 对称加密算法为基础,对其进行混淆变换,给出了一种白盒加密算法,实现了基于加密函数的安全数据加密.后续拟开展的研究主要包括两个方面,一是从理论上探索白盒模糊性的计算方法,二是对本算法在实际环境中进行各种攻击测试.

参考文献:

[1] Maamar Z, Sheng Q Z, Benatallah B. Interleaving web services composition and execution using software agents and delegation [C] // Proceedings of the AAMAS2003 Workshop on Web Services and Agent-based Engineering (WSABE2003). New York: ACM Press, 2003: 56-63.

[2] Chow S, Eisen P, Johnson H, et al. White-box cryptography and an AES implementation [C] // Proceedings of the Ninth Workshop on Selected Areas in Cryptography. Berlin: Springer Verlag, 2002: 250-270.

[3] Chow S, Johnson H, van Oorschot P, et al. A white-box DES implementation for DRM applications[C]// Proceedings of ACM CCS-9 Workshop DRM 2002 - 2nd ACM Workshop on Digital Rights Management. Berlin: Springer Verlag, 2002: 1-15.

[4] Hohl F. Time limited blackbox security: protecting mobile agents from malicious hosts[C] // Lecture Notes in Computer Science. Berlin: Springer Verlag, 1998(1419): 92-113.

[5] Tomas Sander, Christian F Tschudin. On software protection via function hiding[C] // Lecture Notes in Computer Science. Berlin: Springer Verlag, 1998(1525): 111-123.

[6] Tomas Sander, Christian F Tschudin. Protecting mobile agents lecture notes in computer science: against malicious hosts[C]// Mobile Agents and Security. Berlin: Springer Verlag, 1998 (1419): 44-60.

[7] Collberg C, Thomborson C, Low D. Manufacturing cheap, resilient, and stealthy opaque constructs[C] // Proceedings of the 25th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages. New York: ACM Press, 1998: 184-196.

[8] Barreto P, Rijmen V. The Khazad Legacy-level block cipher

[EB/OL]. [2001 - 10 - 10]. https: // www. cosic. esat. kuleuven. ac. be/nessie/workshop/submissions/khazad. zip.
[9] TSENG Yuhmin. An efficient two-party identity-based key exchange protocol[J]. Informatica, 2007, 18(1): 125.
[10] Olivier Billet, Henri Gilbert, Charaf Ech-Chatbi. Cryptanalysis of a white box AES implementation eds[C] // Preceedings of Selected Areas in Cryptography 2004. Berlin: Springer-Verlag, 2004: 227 - 240.
[11] Syncrosoft. Syncrosoft white-box cryptography[EB/OL]. [2008 - 05 - 10] http: // www. syncrosoft. com/ Syncrosoft_White-Box_Cryptography - 78 - 52. html.
[12] Jacob M, Boneh D, Felten E. Attacking an obfuscated cipher by injecting faults [C] // Proceedings of ACM Digital Rights Management Workshop. Berlin: Springer-Verlag, 2002: 16 - 31.
[13] Wyseur B, Michiels W, Gorissen P, et al. Cryptanalysis of white-box DES implementations with arbitrary external encodings[C] // Preceedings of Selected Area in Cryptography, Lecture Notes in Computer Science. Berlin: Springer-Verlag, 2007, 4876: 264 - 277.
[14] Goubin L, Masereel J, Quisquater M. Cryptanalysis of white box DES implementations [C] // Proceedings of the 14th Annual Workshop on Selected Areas in Cryptography, Lecture Notes in Computer Science. Berlin: Springer Verlag, 2007, 4876: 1611 - 3349.
[15] Link H E, Neumann W D. Clarifying obfuscation: improving the security of white-box DES[C] // Preceedings of International Conference on Information Technology: Coding and Computing. Los Alamitos: IEEE Computer Society, 2005, 1 (4 - 6): 679 - 684.

(上接第 64 页)

[3] Waller S T, Schofer J L, Ziliaskopoulos A K. Evaluation with traffic assignment under demand uncertainty[J]. Transportation Research Record, 2001(1771): 69.
[4] Pradhan A, Kockelman K M. Uncertainty propagation in an integrated land use-transportation modeling framework: output variation via UrbanSim[J]. Transportation Research Record, 2002(1805): 128.
[5] Krishnamurthy S, Kockelman K M. Propagation of uncertainty in transportation-land use models: an investigation of DRAM-EMPAL and UTPP predictions in Austin, Texas[C/CD] // the 82nd Annual Meeting of the Transportation Research Board. Washington D C: Transpratation Research Board, 2003.
[6] Oppenheim N. Urban travel demand modeling[M]. New York: John Wiley & Sons Inc, 1995.
[7] McKay M D, Beckman R J, Conover W J. A comparison of three methods for selecting values of input variables on the analysis of output from a computer code[J]. Technometrics, 1979, 21 (2): 239.
[8] Ortuzar J D, Willumsen L D. Modeling transport[M]. 3rd ed. New York: John Wiley & Sons Inc, 2001.
[9] Garret M, Wachs M. Transportation planning on trial[M]. Thousand Oaks: Sage, 1996.
[10] Boyce D, Daskin M S. Urban transportation[C] // Design and Operation of Civil and Environmental Engineering Systems. New York: John Wiley & Sons Inc, 1997: 277 - 341.
[11] Evans S. Derivation and analysis of some models for combining trip distribution and assignment[J]. Transportation Research, 1976, 10(1): 37.
[12] 蒲琪, 杨超, 涂颖菲. 基于二次插值法的交通需求组合模型算法[J]. 同济大学学报: 自然科学版, 2009, 37(12): 1615.
PU Qi, YANG Chao, TU Yinfei. A quadratic interpolation method based algorithm for a combined travel demand model [J]. Journal of Tongji University: Natural Science, 2009, 37 (12): 1615.
[13] Fiacco A V. Introduction to sensitivity and stability analysis in nonlinear programming [M]. New York: Academic Press, 1983.
[14] Yang C, Chen A. Sensitivity analysis of the combined travel demand model with applications [J]. European Journal of Operational Research, 2008(9): 44.
[15] Rai S N, Krewski D, Bartlett S. A general framework for the analysis of uncertainty and variability in risk assessment[J]. Human and Ecological Risk Assessment, 1996, 2(4): 972.
[16] Brattin W J, Barry T M, Chiu N. Monte Carlo modeling with uncertain probability density functions [J]. Human and Ecological Risk Assessment, 1996, 2(4): 820.
[17] Bell M G H, Cassir C, Iida Y, et al. A sensitivity based approach to network reliability assessment[C] // Ceder A. Proc of the 14th Int Symp On Transportation and Traffic Theory. Oxford: Pergamon, 1999: 283 - 300.
[18] Robbins J. Mathematical modeling— the errors of our ways [J]. Traffic Engineering and Control, 1978, 19: 32.
[19] Bonsall P W, Champerwone A F, Mason A C, et al. Transport modeling: sensitivity analysis and policy testing[J]. Progress in Planning, 1977, 7(3): 153.
[20] Leurent F. Sensitivity and error analysis of the dual criteria traffic assignment model[J]. Transportation Research, Part B, 1998, 32(3): 189.