三方认证协议 BNV 的分析及改进

高三海[1],董荣胜[1],吴光伟[1,2]
GAO San-hai[1],DONG Rong-sheng[1],WU Guang-wei[1,2]

1.桂林电子科技大学 计算机学院,广西 桂林 541004
2.中南林业科技大学 计算机科学学院,长沙 410004
1.School of Computer Science,Guilin University of Electronic Technology,Guilin,Guangxi 541004,China
2.School of Computer Science,Central South University of Forestry and Technology,Changsha 410004,China
E-mail:sanhaigao@163.com

GAO San-hai,DONG Rong-sheng,WU Guang-wei.Analysis and improvement of 3-party authentication protocol:BNV. Computer Engineering and Applications,2008,44(4):155-158.

Abstract: Multi-party authentication protocols are used to assure the security sessions of the Multi participants involved in the protocols.In this paper,based on the operational semantics of security protocols,3-party authentication protocols BNV is analyzed. The result shows that it has agreement and synchronization flaws.For this flaws,the message structure of the protocol is amended and a message term identified the agent is added into the protocol messages.Then,the improved protocol is analyzed,which indicates that it can guarantee agreement and synchronization.Finally,an n-party protocol is presented based on the improved 3-party authentication protocols BNV,which is used to confirm the real identities of n agents.

Key words: 3-party protocol BNV;operational semantics;synchronization;agreement.

摘 要:多个主体之间的安全会话需要有可靠的多方认证协议来保证。基于安全协议的操作语义模型,分析了三方认证协议 BNV 的安全性,结果表明该协议存在一致性和同步性缺陷。为此,修改了协议的消息结构并添加了标识协议主体身份的消息项。对改进后协议的安全性进行分析,结果表明改进后的协议不存在原协议的缺陷,协议参与主体满足一致性与同步性要求。最后,基于改进后协议,提出了一个 n 方认证协议的协议原型。

关键词:三方认证协议 BNV;操作语义;同步性;一致性
文章编号:1002-8331(2008)04-0155-04 **文献标识码**:A **中图分类号**:TP393

1 引言

多方认证协议用于多个主体之间相互确定对方的真实身份，防止假冒攻击，它是保证多个主体安全会话的基础。Buttyán L 等人在文献[1]中用图论的方法证明对 n 个协议参与主体进行相互身份认证的认证协议至少需要传递 $2n-1$ 次消息，在此基础上设计了一个三方认证协议 BNV(Buttyán L,Nagy A,Vajda I)并提出了 n 方 BNV 认证协议的协议原型。Cremers C 等人在文献[2]中用安全协议的操作语义模型[4]分析了 NSL(Needham-Schroeder-Lowe)认证协议的认证机制,提出了一个通过 NSL 协议扩展得到的 n 方认证协议,并指出文献[1]中提出的三方认证协议 BNV 不满足一致性[3]和同步性[4,5]要求,协议存在认证性缺陷,但该文没有对此缺陷做进一步分析与改进。

基于此,本文建立了 BNV 三方认证协议的操作语义模型,验证了该协议的一致性和同步性,给出了攻击路径并分析了该攻击产生的原因。针对该缺陷,修改了协议的消息结构,在协议的消息中添加了标识协议主体身份的消息项。对改进后的协议进行分析,结果表明改进后的三方认证协议不存在原协议的缺陷,协议参与主体满足一致性与同步性要求。最后,基于改进后的 BNV 三方认证协议，提出一个用于确认 n 个协议参与主体真实身份的方认证协议的协议原型。

2 三方认证协议 BNV 的描述

文献[1]给出的三方认证协议 BNV 中,协议参与者共有三方:A、B和 C,N_a、N_b、N_c 分别是这三个主体生成的临时值,$k(a,b)$,$k(a,c)$,$k(b,c)$分别为 A 和 B,A和 C 以及 B 和 C 的共享密钥,协议描述如下:

(1)$A\rightarrow B:N_a$
(2)$B\rightarrow C:N_b,\{B,N_a\}_{k(b,c)}$
(3)$C\rightarrow A:N_c,\{C,N_b,B,N_a\}_{k(a,c)}$
(4)$A\rightarrow B:\{A,N_c,C,N_b\}_{k(a,b)}$
(5)$B\rightarrow C:\{B,A,N_c\}_{k(b,c)}$

在协议的第(1)步,A 向 B 发送临时值 N_a;第(2)步,B 生成临时值 N_b,并向 C 发送该临时值和用密钥 $k(b,c)$加密后的 B 和 N_a。当 C 接受到该消息后,用密钥 $k(b,c)$解密消息项$\{B,N_a\}_{k(b,c)}$并检查该消息中是否含有主体 B 的身份标识。第(3)步,C 生成临时值 N_c,并向 A 发送该临时值,以及用 $k(a,c)$加密后临时值

基金项目:广西研究生教育创新计划资助项目(No.2007105950812M17)。
收稿日期:2007-07-04 **修回日期**:2007-09-03

N_a、N_b 和主体 B、C。当 A 收到 C 的消息后,用密钥 $k(a,c)$解密 $\{C,N_b,B,N_a\}_{k(a,c)}$,检查该消息项中是否含有主体 B 和 C 身份标识,并检查该消息是否返回自己生成的临时值 N_a,从而判断是否对主体 B 和主体 C 的身份进行认证。第(4)步,A 向 B 发送用密钥 $k(a,b)$加密后临时值 N_c、N_b 和主体 A、C,B 收到该消息后,用密钥 $k(a,b)$来解密$\{A,N_c,C,N_b\}_{k(a,b)}$,检查该消息中是否含有主体 A 和主体 B 的身份标识,以及自己在第(2)步生成的临时值 N_b,从而判断是否对主体 A 和 C 的身份进行认证。第(5)步,B 向 C 发送用密钥 $k(b,c)$加密后的主体 A、B 和临时值 N_c,当 C 收到该消息后,用密钥 $k(b,c)$解密并检验该消息中是否含有主体 A 和主体 B 的身份标识,以及自己在第(3)步生成的临时值 N_c,从而判断是否对主体 A 和 B 的身份进行认证。

3　操作语义模型及 Scyther 系统

安全协议的操作语义模型[4]是一种分析安全协议的新模型,它可以直接对含有多个主体的协议进行建模与分析。该模型中,每个协议规格包含多个协议参与者的角色规格,每个角色规格则包含角色的初始知识和由发送角色事件、读角色事件以及声明事件构成的事件序列。当协议被执行时,角色被实例化运行,角色事件也相应的被实例为运行事件,协议主体执行这些运行事件,构成协议的协议运行的迹。此外,入侵者能力被定义成一系列的规则,入侵者同协议合法参与者一起参与协议的运行并利用入侵规则获取协议合法参与者的知识,实现对协议的攻击。

基于操作语义模型开发的 Scyther 系统是一个高效的自动化的协议验证工具。该系统采用 Athena 算法,并结合了定理证明和模型检验的多种技术,可对含有多个主体的协议以及多个协议的系统的安全性进行分析。使用 Scyther 对安全协议进行分析时,首先利用操作语义模型将待验证的协议规格成多个角色规格,然后将这些角色规格转化为相应的 Scyther 语言并输入到 Scyther 系统中,Scyther 将调用 Athena 算法并对协议的安全性进行验证。若 Scyther 不能构造安全声明事件的反例,则输出为真,否则输出为假,并给出了相应的攻击路径。Scyther 原理,如图 1 所示。

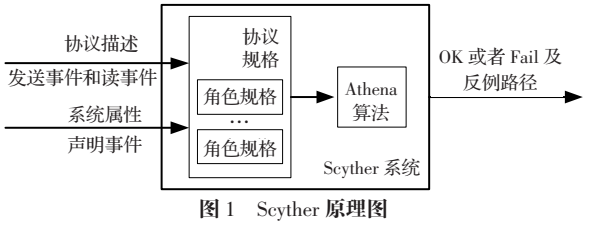

图 1　Scyther 原理图

4　三方认证协议 BNV 的分析

4.1　三方认证协议 BNV 操作语义模型

在 BNV 三方认证协议中,主体将严格按照协议的要求进行通讯,协议的三个参与主体在 Scyther 系统中都有个角色规格,这些角色规格是有角色的初始知识和一系列的角色事件构成的序列组成,当这些角色规格转化为 Scyther 语言输入到 Scyther 系统时,Scyther 就会调用 Athena 算法,按照协议定义的步骤运行协议的角色事件,并检验声明事件是否正确。

4.1.1　主体 A 的角色规格

$BNV(A)=(\{A,B,C,N_0,k(a,b),k(a,c)\})$

$send\_0(A,B,N_0)$

$read\_2(C,A,W,\{B,V,A,N_0\}_{k(a,c)})$

$send\_3(A,B,\{A,W,B,V\}_{k(a,b)})$

$claim\_0(A,Niagree)$

$claim\_1(A,Nisynch)$

其中 $A,B,C,N_0,k(a,b),k(a,c)$为角色 A 所掌握的初始知识,V,W 表示变量。

4.1.2　主体 B 的角色规格

$BNV(B)=(\{A,B,C,N_1,k(a,b),k(a,c)\})$

$read\_0(A,B,X)$

$send\_1(B,C,N_1,\{B,X\}_{k(b,c)})$

$send\_3(A,B,\{A,W,C,N_1\}_{k(a,b)})$

$send\_4(B,C,\{B,C,\{A,B,W\}_{k(b,c)}$

$claim\_2(B,Niagree)$

$claim\_3(B,Nisynch))$

其中 $A,B,C,N_1,k(a,b),k(b,c)$为角色 B 所掌握的初始知识,X,W 表示变量。

4.1.3　主体 C 的角色规格

$BNV(C)=(\{A,B,C,N_2,k(a,c),k(b,c)\})$

$read\_1(B,C,V,\{B,X\}_{k(b,c)})$

$send\_2(C,A,W,\{C,V,B,X\}_{k(a,c)})$

$read\_4(B,C,\{B,A,N_2\}_{k(b,c)})$

$claim\_5(C,Niagree)$

$claim\_6(C,Nisynch)$

其中 $A,B,C,N_2,k(a,c),k(b,c)$为角色 C 所掌握的初始知识,X,V 表示变量。

4.2　三方认证协议 BNV 的验证结果分析

将上述 BNV 三方认证协议的 A、B、C 三个主体的角色规格转化为 Scyther 语言,输入到 Scyther 中,该协议的三个角色规格中所定义的一致性(*Niagree*)声明事件和同步性(*Nisynch*)声明事件都不满足,其验证结果,如表 1 所示。

表 1　BNV 三方认证协议验证结果

性质	角色 A	角色 B	角色 C
Niagree	Fail	Fail	Fail
Nisynch	Fail	Fail	Fail

4.2.1　角色 A 的声明事件的攻击路径

角色 A 规格中所定义的一致性(*Niagree*)声明事件和同步性(*Nisynch*)声明事件的攻击路径,如图 2 所示。

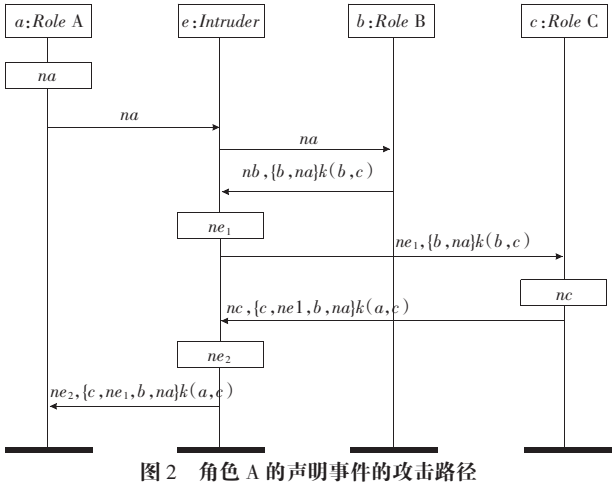

图 2　角色 A 的声明事件的攻击路径

在该攻击中,入侵者首先截获主体 a 的临时值 na,然后将该临时值转发给主体 b,b 收到该临时值后,生成临时值 nb 并向

c 发送消息 $nb,\{b,na\}_{k(b,c)}$。此消息被入侵者截获后,入侵者会生成临时值 ne_1,并冒充 b 向 c 发送消息 $ne_1,\{b,na\}_{k(b,c)}$。c 收到该消息后生成临时值 nc,并向 a 发送消息 $nc,\{b,c,ne_1,na\}_{k(a,c)}$。此时,入侵者再次截获此消息,获取消息项$\{c,ne_1,b,na\}_{k(a,c)}$,生成临时值 ne_2 并向主体 a 发送消息 $ne_2,\{b,c,ne_1,na\}_{k(a,c)}$。$a$ 收到该消息,解密得到自己的临时值 na 以及主体 b,c 的身份标识。此时,a 就会认为相信它与 b,c 成功的进行了会话。但这个会话过程是在入侵者的控制下进行的,因此角色 A 的一致性(*Niagree*)声明事件和同步性(*Nisynch*)声明事件不成立。

4.2.2　角色 B 的声明事件的攻击路径

角色 B 规格中所定义的一致性声明事件和同步性声明事件的攻击路径,如图 3 所示。

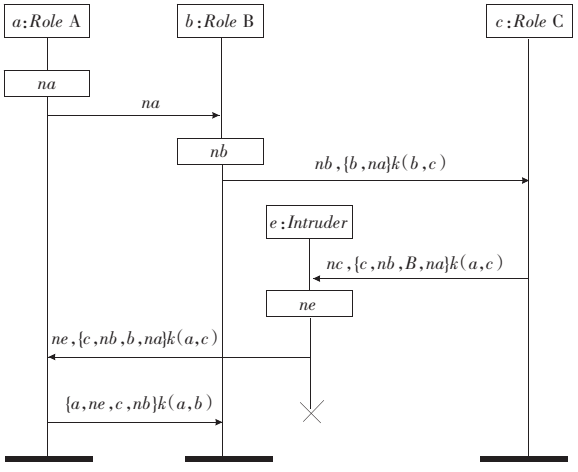

图 3　角色 B 的声明事件的攻击路径

在此攻击路径中,入侵者 e 截获 c 发送给 a 的消息后,获取消息项$\{n,nb,b,na\}_{k(a,c)}$,并将该消息项与自己构造的临时值 na 转发给 a,a 收到该消息后,对该消息进行解密,得到自己构造的临时值 na,主体 b,c 的身份标志,此时,a 相信 b,c 成功的进行了会话。因此,a 会用密钥 $k(a,b)$加密消息主体构造的临时值 na,入侵者 e 构造的临时值 ne 以及主体 a,c 主体 b 接受到该消息后,用密钥 $k(a,b)$解密得到主体构造的临时值 nb 和主体 a,b 的身份标志,此时,b 相信 a,c 成功的进行了会话。但这个会话过程是在入侵者的控制下进行的,因此角色 B 的一致性声明事件和同步性声明事件不成立。

4.2.3　角色 C 的声明事件的攻击路径

角色 C 规格中所定义的一致性声明事件和同步性声明事件的攻击路径,如图 4 所示。

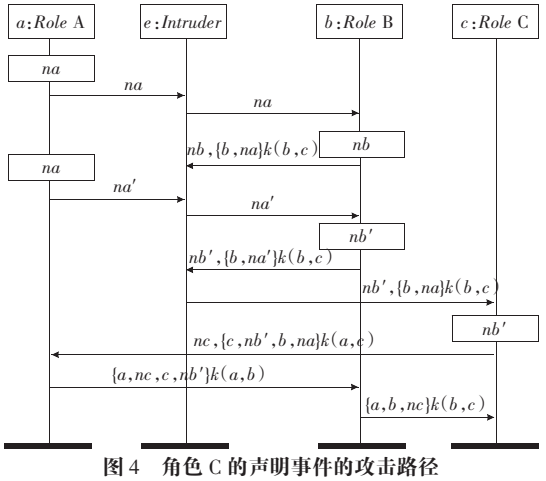

图 4　角色 C 的声明事件的攻击路径

在此攻击路径中,入侵者 e 首先截获第一轮协议会话中主体 a 生成的临时值 na,并冒充 a 将该临时值转发给主体 b,b 收到 na 后,生成临时值 nb 并向 c 发送该临时值及消息$\{b,na\}_{k(b,c)}$,入侵者截获该消息并获取消息$\{b,na\}_{k(b,c)}$。然后,入侵者 e 截获第二轮协议会话中主体 a 生成的临时值 na'并冒充 a 向 b 发送该临时值,b 收到 na'后,生成临时值 nb',并向 c 发送该临时值及消息$\{b,na\}_{k(b,c)}$,入侵者截获该消息并获取临时值 nb'。此时入侵者 e 已经掌握第一轮协议$\{b,na\}_{k(b,c)}$和第二轮协议中 b 生成的临时值 nb',因此入侵者 e 会冒充 b 向 c 发送消息 $nb',\{b,na\}_{k(b,c)}$,主体 c 收到该消息后将生成临时值 nc,并向 a 发送消息 $nc,\{c,nb',b,na\}_{k(a,c)}$,$a$ 收到该消息后用密钥 $k(a,c)$解密后,发现临时值 na 以及 b、c 的身份标识,因此 a 误认为它和 b、c 成功的进行了会话。从而 a 会继续向 b 发送消息$\{a,nc,nb'\}_{k(a,c)}$,b 收到该消息后用密钥 $k(a,b)$解密该消息得到自己构造的临时值 nb'和主体 a,c 的身份标识,因此,b 误认为它同 a、c 成功的进行了会话,从而 b 会继续向 c 发送消息$\{a,b,nc\}_{k(b,c)}$,c 收到该消后用密钥 $k(a,c)$解密发现临时值 nc 以及主体 a、b 的身份标识,因此 c 会误认为它和 a、b 成功的进行了会话,而此过程已被入侵者控制,应此角色 C 的一致性声明事件和同步性声明事件不成立。

5　三方认证协议 BNV 的改进

根据第 3 章的分析可知,三方 BNV 认证协议存在缺陷的主要原因是在协议的第(1)、(2)、(3)步的消息中缺少发送临时值的协议主体的身份标识,入侵者可以冒充主体 A、B、C 的身份构造并发送临时值。为了防止这种攻击,在协议第(1)步的消息中加入 A 的身份标识,并用 A 和 B 对称密钥 $k(a,b)$加密 A 和临时值 N_a;在协议第(2)步的消息中加入 B 的身份标志,并用 B 和 C 对称密钥 $k(b,c)$加密主体 A、B 和临时值 N_a、N_b;在协议第(3)步的消息中加入 C 的身份标志,并用 A 和 C 的对称密钥 $k(a,c)$加密主体 A、B 和 C 以及临时值 N_a、N_b 和 N_c。改进后的协议如下:

(1)$A\rightarrow B:\{A,N_a\}_{k(a,b)}$

(2)$B\rightarrow C:\{A,N_a,B,N_b\}_{k(b,c)}$

(3)$C\rightarrow A:\{A,N_a,B,N_b,C,N_c\}_{k(a,c)}$

(4)$A\rightarrow B:\{A,C,N_b,N_c\}_{k(a,b)}$

(5)$B\rightarrow C:\{A,B,N_c\}_{k(b,c)}$

建立改进后 BNV 协议的操语义模型,将该模型转化成 Scyther 语言,并输入到 Scyther 系统中,运行后发现该协议的三个角色规格所定义的一致性(*Niagree*)声明事件和同步性(*Nisynch*)声明事件都满足。改进后的 BNV 三方协议验证结果,如表 2 所示。

表 2　改进后的三方 BNV 协议验证结果

性质	角色 A	角色 B	角色 C
Niagree	OK	OK	OK
Nisynch	OK	OK	OK

6　*n* 方 BNV 认证协议原型

从上面的分析可知,改进后的三方 BNV 协议能够对三个主体进行认证,但若一个协议有 n 个参与主体,则需要一个可靠的 n 方认证协议来确定彼此身份。为此,本文基于改进后的三方 BNV 协议,设计了一个用于确认 n 个参与主体证实身份的 n 方认证协议,其协议原型如下:

0 $R_0 \rightarrow R_1: \{R_0, N_0\}_{k(0,1)}$

1 $R_1 \rightarrow R_2: \{R_0, N_0, R_1, N_1\}_{k(1,2)}$

… …

$n-1$ $R_{n-1} \rightarrow R_0: \{R_0, N_0, R_1, N_1, R_2, R_2, \cdots, R_{n-1}, N_{n-1}\}_{k(n-1,0)}$

n $R_0 \rightarrow R_1: \{R_0, R_2, \cdots, R_{n-1}, N_1, \cdots, N_{n-1}\}_{k(0,1)}$

$n+1$ $R_1 \rightarrow R_2: \{R_1, R_2, R_4, \cdots, R_{n-1}, N_3, \cdots, N_{n-1}\}_{k(1,2)}$

… …

$2n-2$ $R_{n-2} \rightarrow R_{n-1}: \{R_1, R_2, \cdots, R_{n-1}, N_{n-1}\}_{k(n-2,n-1)}$

此协议共有 $2n-1$ 步，满足 n 个主体间相互认证所需的最少步骤[1]。若用 $Mesg(i)$ 表示协议的第 i 步消息，则根据此 n 方认证协议的消息结构特点可知

$$Mesg(i)=\begin{cases} Mesg\_1(i) & \text{if } 0\leqslant i\leqslant n-1 \\ Mesg\_2(i) & \text{if } n\leqslant i<2n-1 \end{cases}$$

若设 $x=i \bmod n$，$y=(i+1) \bmod n$ 则

$Mesg\_1(i)=\{R_0, N_0, R_1, N_1, \cdots, R_i, N_i\}_{k(x,y)}$

$Mesg\_2(i)=\{[N_0, \cdots, N_{n-1}][N_0, \cdots, N_x], [R_0, \cdots, R_{n-1}]/R_y\}_{k(x,y)}$

在此 n 方认证协议中，当 $0<i\leqslant n-1$ 时，协议主体 R_i 收到主体 R_{i-1} 发送的 $Mesg\_1(i-1)$ 消息后，生成临时值 N_i 再向主体 R_{i+1} 发送消息 $Mesg\_1(i)$；而当 $i=n$ 时，主体 R_0 收到主体 R_{n-1} 发送的消息 $Mesg\_1(n-1)$ 后，检验该消息中是否含有临时值 N_0 及其它 $n-1$ 个主体的身份标志，据此判断与其通讯的主体是否是信任的主体，若 R_0 成功完成了对这些主体的认证则会继续向主体 R_1 发送消息 $Mesg\_2(n)$；同样，当 $n<i<2n-1$ 时，当主体 R_x（$x=i \bmod n$）收到消息 $Mesg\_2(i-1)$ 后，检查该消息中是否含有临时值 N_x（$x=i \bmod n$）及其它 $n-1$ 个主体的身份标志，据此判断与其通讯的主体是否是信任的主体，其协议结构框架如图 5 所示。

若通过一次协议运行，该 n 方认证协议中所有的协议参与主体都成功地对其它的参与主体进行了认证，则称该 n 方协议满足认证性。

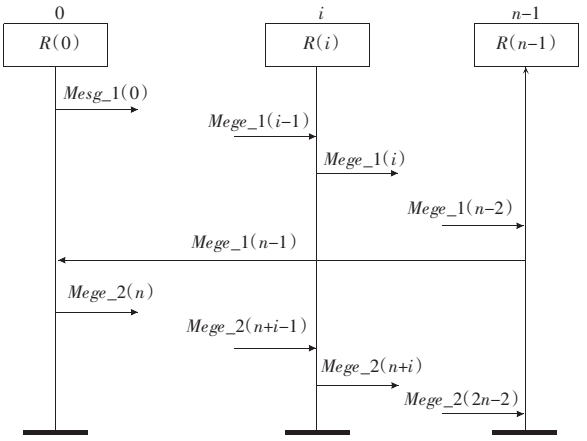

图 5 n 方认证协议结构框架

7 结束语

本文基于文献[1,2]的工作，进一步分析了三方 BNV 认证协议缺陷产生的原因，给出了针对此协议缺陷的改进方案，结果表明改进后的三方认证协议不存在原协议的缺陷，满足一致性与同步性。通过对改进后 BNV 的消息结构扩展，设计了一个 n 方 BNV 认证协议。限于篇幅，本文没有对该 n 方认证协议的安全性作进一步分析，但该 n 方认证协议的消息结构是通过改进后的三方 BNV 认证协议扩展得到的，保留了其安全性质。

参考文献：

[1] Buttyán L, Nagy A, Vajda I.Efficient multi-party challenge-response protocols for entity authentication[J].Periodica Polytechnica, 2001,45(1):43-64.

[2] Cremers C, Mauw S.Generalizing needham-schroeder-lowe for multi-party authentication[EB/OL].[2006].http://people.inf.ethz.ch/cremersc/publications/index.html.

[3] Lowe G.A hierarchy of authentication specifications[C]//Proceedings of the 10th IEEE Computer Security Foundations Workshop, 1997: 31-43.

[4] Cremers C, Mauw S.Operational semantics of security protocols[C]//Scenarios: Models, Transfor mation and Tools, International Workshop Models, 2005: 66-89.

[5] Cremers C, Mauw S, Vink E P, et al.Defining authentication in a trace model[C]//Proceedings of the First International Workshop on Formal Aspects in Security and Trust, 2003: 131-145.

（上接 79 页）

更的部分，从图中可以看出，活动 A2 移至与 A3 并行处理，根据处理状态 S2/S3 的值决定流转至 A4 或 A1；并且把活动 A1 的下一个处理变为活动 A5，活动 A5 的输出流转则根据输出状态 S2/S3 决定。由于改进后的元模型进行了状态和约束的分离，整个流程调整后只需部分修改连接器 C1、C2、C3 和状态连接转移 SCT1、SCT2、SCT3 的部分参数（图 4 中的下划线部分），无需修改活动本身的具体事务处理，从而使得元模型具备了柔性和可复用的特征。

5 结束语

在对现有文献分析的基础上，针对业务流程柔性变更的需求，从过程定义元模型的角度出发，把状态和控制类型（汇聚、分支和约束）从 WFMC 定义的工作流过程定义元模型的活动和转移条件中分离出来，重新构建了一个改进的工作流过程定义元模型，使得系统对一些无法预计的动态变更因素，能够在系统执行时进行柔性变更，提高了工作流系统的应变能力和适应能力，具有一定的现实意义。

在此过程定义元模型的指导下，要使一个工作流管理系统得到实现，需研究如何使应用程序根据流程变更自动调整连接器和状态连接转移的 WPDL 描述的算法。这些是我们要进一步解决的问题。

参考文献：

[1] Hollingsworth D.Workflow management coalition—the workflow reference model.Hampshire, U K Winchester, 1995.

[2] Momotko M, Subieta K.Process query language: a way to make workflow processes more flexible[C]//LNCS, Springer-Verlag Heidelberg, 2004: 306-321.

[3] Kumar A, Wainer J.A framework for coordination, exception handling and adaptability in workflow systems[C]//LNCS, 2004, 3095: 13-27.

[4] 孙瑞志，史美林.支持工作流动态变化的工作流元模型[J].软件学报，2003，14(1)：62-67.

[5] 赵文，胡文蕙.工作流元模型的研究与应用[J].软件学报，2003，14(6)：1052-1059.

[6] Workflow Management Coalition.Interface 1: process definition interchange process model[EB/OL].WfMC TC 1016 P, [1999-10]. http://www.WfMC.org.