基于可转换代理签密的SAML跨域单点登录认证协议

王冠众 张 斌 费晓飞 熊厚仁

(解放军信息工程大学三院 郑州 450001)

摘 要 可转换代理签密算法具有保护用户隐私、抗重放攻击、抗抵赖性等优势,基于该算法提出一种SAML跨域单点登录协议(SSPCPS)。通过用户与异构域服务器直接交互认证,简化了跨域单点登录认证过程。用户身份票据由双方公钥结合用户随机选取的参数而生成,以密文形式传输,攻击者即使窃取该令牌也无法调用服务。用户利用代理签名密钥对摘要进行签密,在减少计算量的同时也可保证用户隐私安全。SSPCPS协议基于DH算法协商会话密钥,简化了会话密钥分发过程并降低了管理成本。使用CK安全模型证明了本协议的安全性并进行了性能分析,结果表明协议具有前向保密性、消息完整性等特点,同时在生成票据计算量和计算时间方面优于SSPPS协议、Juang方案、Kerberos机制等。

关键词 代理签密,单点登录,安全断言标记语言,认证

中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.4.020

SAML Cross-domain Single Sign-on Authentication Protocol Based on Convertible Proxy Signcryption

WANG Guan-zhong ZHANG Bin FEI Xiao-fei XIONG Hou-ren

(The Third Institute, The PLA Information Engineering University, Zhengzhou 450001, China)

Abstract Convertible proxy signcryption algorithm has the advantages of protecting user privacy, anti-replay attack, anti-disavowal etc. A SAML cross-domain single sign-on authentication protocol (SSPCPS) was proposed based on the algorithm. Through user and heterogeneous domain server interacting and authenticating directly, the protocol simplifies the process of SSO authentication. User token is generated by combining selected random parameters with the public key, and is transferred in the secret form, improving the security of protocol. The attacker cannot use the service, even though the token is stolen. Proxy signature key is used to signcrypt the digest, reducing the amount of computation, and ensuring the privacy of user as well. Session key is negotiated based on DH algorithm, simplifying the distribution process as well as reducing the management cost. The security of the protocol was proved by CK security model and performance analysis was presented. The result indicates that the protocol holds the features of forward secrecy, message integrity, etc, and the amount of computation and the computation time of generating token are better than SSPPS protocol, Juang scheme and Kerberos scheme, etc.

Keywords Proxy signcryption, Sigle-sign-on, SAML, Authentication

1 引言

随着电子商务和网络技术的不断发展,在各种业务活动中用户往往需要访问不同的应用系统。为避免跨域访问服务时应用系统对用户的重复注册认证,提高用户访问服务效率,需要建立单点登录(Single Sign-on, SSO)系统。单点登录是指用户在网络中只需进行一次身份认证,即可访问所需应用系统,而无需再次进行身份认证[1]。基于安全断言标记语言(Security Assertion Markup Language, SAML)[2,3]的单点登录系统(SAML SSO)是当前单点登录的研究热点[4]。SAML可以在多个Web应用程序中实现身份认证,解决了用户在多个域中进行身份验证及身份属性信息共享的问题。

传统的SAML跨域单点登录系统[5]中存在安全令牌被窃取的隐患[6]和用户信息泄露等隐患。文献[6,7]提出的跨域单点登录模型,简化了单点登录的流程,认证效率较高,但在安全令牌的保护方面考虑不足,存在重放攻击隐患。文献[8]提出的以用户为中心的基于代理签名的SAML单点登录协议,通过限定用户的访问对象和限制访问时间,加强了跨域单点登录的安全性,但该协议在传送报文过程中由于公开传递用户信息(如访问服务列表),容易暴露用户隐私。传统的SAML认证方案中[5],用户若要访问异构域的服务,需要认证服务器之间进行多次交互,过程较为繁琐,灵活性不够。当大量用户同时发出访问请求时,可能导致该域认证服务器堵塞,无法及时响应。

Kerberos[9]是一种高安全性的身份认证机制,扩展性强,可在多域环境中使用,但在用户密钥管理方面,随着用户数增

到稿日期:2014-05-14 返修日期:2014-08-09 本文受河南省基础研究计划项目(142300413201)资助。

王冠众(1986-),男,硕士生,主要研究方向为SOA安全服务认证,E-mail:wgznavy@.126com;张 斌(1969-),男,教授,主要研究方向为网络信息安全;费晓飞(1977-),男,博士,主要研究方向为SOA安全;熊厚仁(1986-),男,博士生,主要研究方向为网络信息安全。

多，密钥的管理会越来越复杂，并且密钥的分发一般在离线状态下预先分配。

代理签密[10]是指原始签名人将他对全部或部分消息的签名权委托给代理签名人，代理签名人代表原始签名人对委托内容生成有效的签名，并对其中关键参数（用户隐私、敏感数据）进行加密。可转换代理签密算法[10]具有保护用户隐私、抵抗重放攻击等优势，代理签密比单独使用签名和加密生成消息的计算量更低[11]。

为了尽可能降低本域服务器在过多用户同时申请调用服务时造成的网络堵塞，提高认证效率，保护用户隐私安全，降低票据生成的计算量以及高效安全地传递身份票据，减少密钥管理成本，本文依据文献[10]提出的可转换代理签密算法，设计了一种基于可转换代理签密的 SAML 跨域单点登录认证协议（SAML Cross-domain Single Sign-on Authentication Protocol Based on Convertible Proxy Signcryption，SSPCPS）。利用 DH 协商会话密钥，使用扩展元素的 SAML 报文传递票据参数，并运用 CK 安全模型对其安全性进行证明。

2 基于可转换代理签密的跨域单点登录认证协议

2.1 协议设计

2.1.1 基本思想

基于可转换代理签密的跨域单点登录认证协议的主要思想是用户首先在本域进行登录认证，通过认证的用户与本域认证服务器协商产生代理签名密钥；随后用户与访问域双向认证，用户与访问域根据代理签密算法相互验证彼此身份，若均合法，则用户和该域协商生成会话密钥并调用服务。

2.1.2 协议描述

SSPCPS 协议基本认证流程如图 1 所示。

图 1 SSPCPS 协议认证流程

本文所用标识和符号如表 1 所列。

表 1 文中标识和符号

符号	描述	符号	描述
u	用户	t,t',t''	时间戳
$E_k()$	使用对称密钥 k 加密	$True_u$	u 是域 A 中的合法用户
$D_k()$	使用对称密钥 k 解密	L	用户授权信息
ID_u	U 的身份标识	n_u	u 的票据标识
URL_A	域 A 的身份标识	n_B	B 的票据标识
URL_B	域 B 的身份标识	(t_1,t_2)	(t_1 是生效时间，t_2 是失效时间)
$cert_x$	x 的代理身份证书		

(1)登录认证，代理密钥生成。

在域 A 注册的用户 u 通过认证后，用户 u 与 A 协商产生代理签名密钥，代理签名密钥根据 Nicolosi 等[12]提出的可证明安全的两方签名方案而生成。生成代理签名密钥的具体过程如下：

定义 $E(GF_q)$ 是有限域 $GF(q)$ 上的椭圆曲线，O 为椭圆曲线 $E(GF_q)$ 上的无穷远点，密钥生成中心 KGC 选取大素数 q，P 是 $E(GF_q)$ 的一个生成元，n 是选取的安全参数，KGC 负责生成域 A、用户 u 和域 B 的公私钥对：(x_i,y_i)。其中私钥 $x_i \in Z_q^*$，公钥 $y_i=x_iP$，$i\in A,u,B$。定义安全 Hash 函数 $H:\{0,1\}^{256}\rightarrow Z_q$（本文使用 SHA-256），符号“‖”表示消息连接操作。

第一步：域 A 为用户 u 创建代理身份证书 $cert_u=[ID_u,URL_A,true_u,(t_1,t_2),L]x_A$，随机选取 $j_0\in Z_q^*$，计算 $g_0=j_0P$，$MAC=H(g_0)$，将 $(MAC,cert_u)$ 发送至用户 u。

第二步：用户 u 接收到 $(MAC,cert_u)$ 之后，随机选取 $k_1\in Z_q^*$ 并计算 $g_1=j_1P$，将 g_1 发送给域 A。

第三步：域 A 计算验证 $qg_1=O$ 是否成立。若成立，则计算：

$$\begin{cases} g=g_0+g_1 \\ \delta=j_0+x_AH(g\parallel cert_u\parallel y_u\parallel y_A) \end{cases}$$

将 (δ,g_0) 发送给用户 u。

第四步：在生成代理签名公私钥对之前，需要验证 $(MAC,\delta,cert_u)$ 的正确性，用户 u 进行如下计算：

$$\begin{cases} g=g_0+g_1 \\ MAC=H(g_0) \\ MAC=H(\delta P-y_AH(g\parallel cert_u\parallel y_u\parallel y_A)) \end{cases}$$

若上式正确，那么用户 u 生成代理签名公私钥对：(x_m,y_m)，即

$$\begin{cases} x_m=\delta+g_1+x_uH(g\parallel cert_u\parallel y_u\parallel y_A) \\ y_m=g+H(g\parallel cert_u\parallel y_u\parallel y_A)(y_u+y_A) \end{cases}$$

其中 $\delta=j_0+x_AH(g\parallel cert_u\parallel y_u\parallel y_A)$

若验证未通过，则要求域 A 重新计算并发送 (δ,g_0)。

(2)用户 u 与域 B 双向认证，生成会话密钥。

第一步：用户 u 生成身份票据 $Ticket_u$。

用户 u 随机选取 $a,b\in Z_q^*$，t 为当前时间，计算：

$$\begin{cases} V_1=aP,V=bP,V_2=ay_B \\ c=a+x_mH(V_1\parallel y_B\parallel V_2\parallel t\parallel g\parallel V) \\ V_3=E_{V_2}(c\parallel cert_u\parallel URL_A\parallel y_A\parallel t\parallel g\parallel V) \end{cases}$$

然后，用户 u 将票据 $Ticket_u=(R_1,R_3,URL_B,ID_u,n_u)$ 发送至域 B。

第二步：域 B 生成身份票据 $Ticket_B$。

域 B 收到 $Ticket_u$ 时，记录下当前时间 t'，进行如下计算：

$$\begin{cases} V_2=x_BV_1 \\ c\parallel cert_u\parallel URL_A\parallel y_A\parallel t\parallel g\parallel V=D_{R_2}(V_3) \end{cases}$$

解密 V_3 可获得用户信息和重要参数，根据 URL_A 可知证书是由域 A 生成的，通过公钥 y_A 验证 $cert_u$ 的合法性并解读证书内容，检查当前时间 t' 是否在 (t_1,t_2) 有效时限内，若未超出时限，则票据有效，通过 $true_u$ 可确认用户 u 在域 A 中已注册。根据用户拥有的权限 L，域 B 有选择地对用户 u 提供服务以及限定访问时间，从而避免了用户非法调用服务。

验证如下公式的正确性：$cP=V_1+H(V_1\parallel y_B\parallel V_2\parallel t\parallel g\parallel V)(g+H(g\parallel cert_u\parallel y_u\parallel y_A)(y_u+y_A))$

若该式成立，则证明票据是由域 A 中的合法用户 u 生成并发出。

域 B 随机选取 $e\in Z_q^*$，计算 $V_4=eP$，生成用户 u 与域 B 的会话密钥：$k_{pwd}=eR=ebP$，对会话密钥进行摘要计算：$S=H(k_{pwd})$，记录下票据生成时间 t''，域 B 私钥对相关参数进行签名：$M=[V_4\parallel t''\parallel S\parallel URL_B\parallel n_B]x_B$。最后，生成域 B 身份票据 $Ticket_B=(M,t'',URL_B,ID_u,y_B,n_B)$ 发送给用户 u。

第三步：生成会话密钥。

用户 u 收到域 B 发送的身份票据 $Ticket_B$ 后，使用域 B 公钥验证签名并解析得到 $V_4\parallel t''\parallel S\parallel URL_B\parallel n_B$，确认消息

来自域 B，计算 $k'_{pud}=bV_4=beP$，验证 $H(k'_{pud})=S$ 是否成立，若成立，由于只有域 B 使用自身私钥 x_B 才能计算 V_2，最终计算出 V，则用户 u 可以确认域 B 生成的共享密钥 $k_{pud}=eV=ebP$ 有效，否则停止对话。用户 u 和域 B 之间通过生成的对称密钥 k_{pud} 进行通信，且域 B 为用户 u 提供相应服务。

在用户申请调用服务时，用户直接和域 B 进行双向认证，域 A 和域 B 无需任何信息的交互，从而很大程度上简化了单点登录认证过程，也降低了大量用户同时访问域 A 认证服务器造成网络堵塞的隐患。

通过身份票据中的时间戳以及验证 SAML 断言中的票据标识与访问域缓存中的票据标识是否重复，可判断断言是否被重放，能够消除因 XML 安全性不足和通信层安全机制不完善导致的重放攻击等安全隐患。用户的代理身份证书、权限以及重要参数等信息均加密传输，整个认证过程中，攻击者无法获取这些敏感信息，保证了重要信息以及用户隐私的安全。票据令牌由用户维护，减少了服务器专门管理令牌的成本。协议也具有抗用户抵赖性，若用户 u 出现违法或抵赖行为，域 B 可以将关键信息 $(V,V_1,V_2,cert_u,c,g,t)$ 提交给可信权威机构进行仲裁，维护域 B 的合法权益。

2.2 报文设计

SAML 断言基于 XML 格式，通过 SOAP 协议进行传输。XML 具有良好的扩展性和开放性，依据该特性设计协议中用户 u、域 B 身份票据的 SAML 断言，其用来保证用户传递消息的高效性。

在 SAML 断言的基础上对其进行扩展，加入代理签密根元素：ps(Proxy Signcryption)，在 ps 下增加用户 u 身份票据元素 IdentityTicket，其分为两部分：IDTFirstPart 和 IDTSecondPart，其它元素格式不变。

基于 $Ticket_U=(V_1,V_3,URL_B,ID_u,n_u)$ 定义的用户 u 的身份票据断言表示如下：

```
〈saml:Assertion
Version="2.0"
ID="nu"
IssueInstant="T"
NotBefore=t1 NotOnOrAfter=t2〉
〈saml:Issuer〉IDA〈/saml:Issuer〉
〈ps:Proxy Signcryption〉
  〈ps:SignedInfo〉
〈ps:DigestMethod Algorithm="SHA-2"/〉
  〈ps:DigestValue〉〈ps:DigestValue〉
 〈ps:SignedInfo〉
〈ps:IdentityTicket〉
〈ps:IDTFirstPart〉V1〈ps:IDTFirstPart〉
〈ps:IDTSecondPart〉V3〈ps:IDTSecondPart〉
  〈ps:IdentityTicket〉
  〈ps:Proxy Signcryption〉
  〈saml:Subject〉
〈saml:NameID format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent"〉 URLB
 〈/saml:NameID〉
 〈/saml:Subject〉
 〈/saml:Assertion〉
```

将 $Ticket_u$ 中包含有用户信息和协商会话密钥信息的 V_1 和 V_3 归于 IdentityTicket 元素中，以方便进行验证和协商生成会话密钥。

基于 $Ticket_B=(M,t'',URL_B,ID_u,y_B,n_B)$ 定义的域 B 的身份票据断言表示如下：

```
〈saml:Assertion
Version="2.0"
ID="nB"
IssueInstant="T0"
NotBefore=t1 NotOnOrAfter=t2〉
〈saml:Issuer〉URLB〈/saml:Issuer〉
〈ps:Proxy Signcryption〉
  〈Ps:SignedInfo〉
   〈ps:Signature〉M〈ps:Signature〉
  〈ps:SignedInfo〉
  〈ps:TicketTime〉t'' 〈ps:TicketTime〉
  〈ps:KeyInfo〉
〈ps:ReceiverKey〉yB〈/ps:ReceiverKey〉
  〈/ps:KeyInfo〉
 〈ps:Proxy Signcryption〉
 〈saml:Subject〉
〈saml:NameIDformat="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent"〉 IDu
 〈/saml:NameID〉
 〈/saml:Subject〉
 〈/saml:Assertion〉
```

在 ps 下增加票据生成时间 TicketTime 和验证公钥 KeyInfo，即接收者的公钥 ReceiverKey。域 B 对生成的消息进行签名，用户 u 在收到域 B 发送的消息后，首先进行签名验证，若正确则依据代理签密算法生成会话密钥。

3 SSPCPS 协议安全性与性能分析

CK 安全模型[13-15]是一种对密钥交换协议进行模块化分析的安全模型，本文使用 CK 安全模型证明 SSPCPS 协议的安全性。

3.1 CK 安全模型

CK 安全模型[13-15]主要由 3 部分组成：认证链路模型 AM、非认证链路模型 UM 和认证器。

(1)AM 模型

AM 模型可看作一种理想环境，主体只能由攻击者传递参与者产生的真实消息激活。消息由攻击者传递且只能传递一次，并且不能被篡改，最终保证传递到目的地。

(2)UM 模型

UM 模型可看作真实环境，UM 环境下的攻击者具备 AM 模型中的所有能力，并且能够激活一个主体与另一主体的会话，还可以任意重放和篡改消息。

在 CK 安全模型中，攻击类型主要有 3 种：合作攻陷(Party Corruption)、会话密钥查询(Session Key Query)和会话状态暴露(Session State Reveal)。

(3)认证器

定义 1[13] 编译器 CL(Compiler)是指将某协议在特定条件下转换成另一协议的算法。

将 AM 下的安全协议 ρ 转换成 UM 下安全属性相同的协议 $C(\rho)$ 的编译器成为认证器。

定义 2[13] 所谓消息传输(Message Transmission，MT)协议，即该协议只有一个功能，就是将消息由一个参与者传递至另一个参与者。

定理 1[13] 若 λ 为 MT-认证器，那么 C_λ 也是 MT-认证

器。

(4)会话密钥安全

攻击者 At 可以在已完成会话、未过期会话、未暴露会话中任意选取一个进行测试会话查询，公平抛硬币 μ，令 $\mu \xleftarrow{R} \{0,1\}$，$k$ 为会话密钥。若 $\mu=0$，则将 k 赋予 E；否则，从密钥生成空间随机选取数值 r 赋予 E，E 输出 μ 的猜想 μ'。

定义 3[13] 当且仅当符合如下两个条件，协议 ρ 即可产生安全会话密钥：

条件 1 协议 ρ 可保证任意两个可靠实体在执行协议后得到相同会话密钥。

条件 2 在 UM 下，设 ε 是一个可忽略的概率(称为"优势")。攻击者 At 能够猜出比特 μ 的概率为 ω，$\omega \leqslant 1/2+\varepsilon$。

定理 2[13] 在 AM 下，若协议 ρ 可生成安全会话密钥，λ 是 MT-认证器，则在 UM 下，协议 $C_\lambda(\rho)$ 也可生成安全会话密钥。

定理 3[13,15] 数字签名机制可以抗选择消息攻击，基于数字签名的 MT-认证器 Sig_λ 能够实现协议从 AM 下的状态向 UM 下的状态转换。

Sig_λ 定义如下：

设 k 为安全参数，m 为用户 u_i 发送给用户 u_j 的消息。

1)u_i 选取消息 m 并发送给 u_j；

2)u_j 接收 m 并随机选择 $\beta(\beta \xleftarrow{R} \{0,1\}^k)$，将 $\{m,\beta\}$ 发送给 u_i；

3)u_i 在接收消息 $\{m,\beta\}$ 之后，生成消息 $\{m, Sig(P_i,(m,\beta,P_j))\}$ 发送到 u_j；

4)在收到签名消息后，u_j 对签名进行验证，若正确，则可确认消息 m 来自 u_i。

定理 4[13,15] 如果 DDH 假设成立，那么 ρ 协议在 AM 环境下是安全的。

3.2 SSPCPS 协议安全性证明

为了证明 SSPCPS 协议的安全性，运用 CK 安全模型进行证明，首先将协议还原成 AM 下的协议状态，其次证明协议在 AM 下是安全的，最后结合相关定理证明协议在 UM 下的安全性。具体证明过程如下：

(1)将 SSPCPS 协议还原成 AM 下协议。由于生成的域 B 身份票据中使用了 Sig_λ，可去除 Sig_λ 得到 AM 下的协议 ρ。

(2)为了证明 SSPCPS 协议的安全性，首先证明在 AM 下协议 ρ 是安全的会话密钥协议。

在 AM 环境下，当两个活动实体执行协议后，分别得到没有被篡改的 bp 和 ep，由于实体标识 URL_B、ID_u 可将 bp 和 ep 与特定的会话匹配，因此可生成两个实体间相同的会话密钥，所以协议 ρ 符合定义 3 的条件 1。接下来证明协议 ρ 符合定义 3 的条件 2。

进行 DDH 假设，分析攻击者猜中会话密钥的概率，再结合相关定义进而证明协议 ρ 在 AM 下的安全性。

DDH 假设 $E(GF_q)$ 是有限域 $GF(q)$ 上的椭圆曲线，设 P 为循环群 G 上的生成元，用户 u 随机选取 $m \in Z_q^*$，域 B 随机选取 $n \in Z_q^*$，会话密钥 $k_{pwd}=mnP$，攻击者 At 猜中 $k_{pwd}=mnP$ 即成功，$W_1=k_{pwd}=mnP$ 表示会话密钥，$W_2=k'_{pwd}=m'n'P$ 表示随机数，其中 $m' \in Z_q^*$，$n' \in Z_q^*$，M_1，M_2 的概率分布在计算上不可区分。

假设在 AM 下，攻击者 At 在执行协议 ρ 的过程中能够以不可忽略的概率 ε' 区分出会话密钥和随机数，构造算法 Alg 能以一定的概率区分 W_1、W_2。设 Alg 的输入为 $W \in (W_1, W_2)$，W 为 W_1 或 W_2 的概率均为 1/2，算法 Alg 将攻击者 At 作为子程序，令 $\{1\cdots N\}$ 表示攻击者 At 在攻击过程中激发的会话次数，算法 Alg 描述如下：

1)在 AM 环境下，激活攻击者 At 与两个实体用户 u 和域 B 进行交互。

2)攻击者 At 激活某一实体创建新会话，算法 Alg 代表另一实体执行 ρ 协议。在交互过程中，若某次会话暴露或者某一实体被攻陷，Alg 将该事件的相关信息反馈至 A。若会话过期，那么该交互进程中的会话密钥将被参与者舍弃。

3)设 $n \in \{1\cdots N\}$，在激活第 n 次会话时，Alg 令用户 u 将身份票据 $Ticket_u$ 传输至域 B。域 B 收到用户 u 发送的消息后，Alg 令域 B 将其身份票据发送至用户 u。如果攻击者 At 将第 n 次会话作为测试会话查询，那么 Alg 将查询响应结果 W 发送给攻击者 At。

4)当攻击者 At 有如下情况之一时，Alg 结束活动并输出 $\beta' \xleftarrow{R} \{0,1\}$。

a)没有进行测试会话查询就终止活动；

b)选取其它会话进行测试会话查询；

c)第 n 次会话暴露。

5)当攻击者 At 结束攻击并输出 β' 时，Alg 同时结束，输出也为 β'。

由算法分析可知，若攻击者 At 选取第 n 次会话进行测试会话查询并且获得响应结果 W，当 Alg 的输入为 W_1 时，$W=W_1$，即攻击者 At 获得真实会话密钥；当 Alg 的输入为 M_2 时，$W=W_2$，即攻击者 At 获得一个随机数。由于 Alg 的输入 W 为 W_1 或 W_2 的概率均为 1/2，因此攻击者 At 的响应是真实会话密钥或随机数的概率为 $1/2+\varepsilon'$，因为 ε' 不能被忽略，那么攻击者 At 将以不可忽略的概率猜中会话密钥。由于 Alg 与攻击者 At 的输出均为 β'，因此 Alg 猜中它的输入来自 W_1 或 W_2 的概率也是 $1/2+\varepsilon'$。攻击者 At 若未选取第 n 次会话作为测试会话，那么所得的响应为随机数，Alg 猜中它的输入来自 W_1 或 W_2 的概率是 1/2。

攻击者 At 从 N 次会话中选中第 n 次会话的概率是 $1/N$，未选中的概率是 $1-1/N$，可以计算出 Alg 猜中它的输入来自 W_1 或 W_2 概率：

$$\lambda=(1/2+\varepsilon')\cdot 1/N+1/2\cdot(1-1/N)=1/2+\varepsilon'/N$$

因为 ε'/N 不能忽略，可得到如下结论：Alg 以 $1/2+\varepsilon'/N$ 的不可区分概率猜中它的输入来自 W_1 或 W_2，该结论与 DDH 假设矛盾。综上所述，协议 ρ 符合定义 3 中的条件 2，故协议 ρ 在 AM 下是安全的会话密钥协议。

(3)根据以上证明，依据定理 2、定理 3 和定理 4 可以得出 SSPCPS 协议在 UM 环境下是安全的。

通过对 SSPCPS 协议的安全性证明，可知该协议同时具有如下特性。

消息完整性：在生成域 B 身份票据时使用了数字签名，保证了数据的完整性。

双向密钥控制[16]：用户 u 和域 B 的会话密钥根据双方给定的参数、基于 DH 密钥交换协商而产生，b 和 e 均由双方随机选取，故双方均无法单独生成会话密钥 k_{pwd}，并且第三方无法得到相关参数。

前向保密性[17]：由于会话密钥的生成基于 DH 密钥交换，对于攻击者而言，即使窃取域 B 的私钥，面对 DH 数学难

题也无法求解出使用过的会话密钥，会话密钥的产生具有安全性，因此协议具有前向保密性。

3.3 性能分析

本文与 SSPPS 协议[8]、Juang 方案[18]进行传输效率对比。3 种方案中的认证过程均需两轮交互，假设代理签名密钥已生成并分发到用户，在对比分析时均只考虑用户与异构域认证过程中生成票据的大小和生成用时两方面的性能，签名采用椭圆曲线 ECC[19]实现，杂凑函数选用 SHA-256。

SSPPS 协议首先由身份提供者 IDP 为用户生成并分发身份票据 IDT_{uid}，其中，uid 的长度为 50bit，m 的长度为 128bit，URL_{idp} 的长度为 50bit，ACL_{uid} 的长度为 128bit，nc 的长度为 6bit，(t_1,t_2) 的长度为 12bit，$cert^*$ 的长度为 384bit，故 IDT_{uid} 的长度为 858bit。其次，用户生成的访问令牌 $ACST_{uid}$ 中，uid 的长度为 50bit，pk_{uid} 的长度为 128bit，ACL_{uid} 的长度为 128bit，Y^* 的长度为 128bit，σ 的长度为 128bit，URL_{us} 的长度为 128bit，故 $ACST_{uid}$ 的长度为 690bit。这两个过程共传输 1548bit。

Juang 方案中需要 2 轮认证共 4 次票据传输，第一轮交互是异构域对用户的认证，需要两次票据传输，第一次票据 $(N_1,Q,r_H,R,S_U,y_U,y_w,ID_H)$ 的长度为 128+50+164+164+128+128+128+50=940bit，第二次票据 $[N_1 \| Q \| y_U \| n_2]_{KVH}$ 中 $N_1 \| Q \| y_U \| n_2$ 的长度为 128+50+128+50=356bit，故需要 3 次 128bit 对称密钥加密，票据长度为 384bit；第二轮交互是用户对异构域认证，同样需两次票据传输，第一次票据 $[h(N \| Q \| y_{Unew} \| n_2),c,l]_{KVH}$ 中 $h(N \| Q \| y_{Unew} \| n_2),c,l$ 的长度为 256+50+50=356bit，故需 3 次 128bit 对称密钥加密，票据长度为 384bit；第二次票据 $(h(N \| Q \| y_{Unew} \| n_2),n_2)$ 长度为 256+50=306bit。传输总长度为 2014bit。

在本文的协议中，用户身份票据 $Ticket_u$ 为 526bit（V_1 为 164bit，V_3 为 256bit，URL_B 为 50bit，ID_u 为 50bit，n_u 为 6bit），域 B 身份票据 $Ticket_B$ 的长度为 722bit（M 为 446bit，t' 为 6bit，URL_B 为 50bit，ID_u 为 50bit，y_B 为 128bit，n_B 为 6bit），传输的总长度为 1252bit。具体对比如表 2 所列。

表 2 协议票据大小对比

协议	第一轮认证	第二轮认证	总计
SSPPS	858	690	1548bit
Juang 方案	940+384	384+306	2014bit
SSPCPS	526	722	1252bit

另外，对 SSPCPS 协议与 Kerberos[9]跨域认证机制中用户在认证服务器 AS 进行认证时的计算量进行比较，由于生成授权票据是在认证之后，故该部分不参与对比。Kerberos 跨域认证共分为 3 阶段：第一阶段，用户向异构域发出认证请求 (ID_c,ID_B,N_c)，长度为 50+50+50=150bit；第二阶段，异构域与本域中的认证服务器 ASR 进行交互，验证用户身份并生成用户身份票据，该阶段需要传输的消息分为两部分。消息 $[ID_c,ID_{TGS},N_c]_{kfed}$ 中 ID_c,ID_{TGS},N_c 的长度为 50+50+50=150bit，故需要 2 次 128bit 对称密钥加密，即长度为 256bit；消息 $[K^B_{c,TGS},T_{c,TGS},[K^B_{c,TGS},T_{c,TGS}]_{kpud},N_c+1]_{kfed}$ 中 $K^B_{c,TGS},T_{c,TGS},[K^B_{c,TGS},T_{c,TGS}]_{kpud},N_c+1$ 的长度为 128+384+512+50=1074bit，故需要 9 次 128bit 对称密钥加密，即长度为 1152bit；第三阶段 AS 向用户发送身份票据和通信密钥，传输的消息为 $[K_{c,TGS},T_{c,TGS}]_{kpud}$，其中 $K_{c,TGS},T_{c,TGS}$ 的长度为 128+384=512bit，故需要 4 次 128bit 对称密钥加密，即长度为 512bit，因此传输消息的总长度为 150+256+1152+512=2070bit。可见，Kerberos 比本文协议认证过程复杂且计算量要大。

票据生成用时方面，T_a 代表计算 Hash 的时间，T_b 代表点乘计算时间，T_c 代表加解密算法计算时间。在同一实验环境下，SSPPS 协议中共有 10 次 Hash 计算，7 次点乘计算，4 次加/解密算法计算，所需时间为 $10T_a+7T_b+4T_c$；Juang 方案中共有 7 次 Hash 计算，9 次点乘计算，$N+6$ 次加/解密计算，所需时间为 $7T_a+9T_b+(N+6)T_c$。本文 SSPCPS 协议中共有 5 次 Hash 计算，7 次点乘计算，3 次加/解密计算，所需时间为：$5T_a+7T_b+3T_c$。具体对比如表 3 所列。

表 3 协议票据生成用时对比

协议	T_a	T_b	T_c	总计(ms)
SSPPS	10	7	4	$(10T_a+7T_b+4T_c)$
Juang 方案	7	9	N+6	$(7T_a+9T_b+(N+6)T_c)$
SSPCPS	5	7	3	$(5T_a+7T_b+3T_c)$

通过分析可知，SSPCPS 协议在生成票据计算量和计算时间方面优于类似协议。

结束语 本文提出的基于可转换代理签密的 SAML 跨域单点登录认证协议具有保护用户隐私及票据安全、抗重放攻击、不可抵赖性等优点；而且协议通过用户直接与异构域认证服务器交互认证，能够灵活地生成会话密钥，简化了密钥分发过程，降低了密钥管理成本，为缓解传统单点登录过程中本域认证服务器负载过重以致网络堵塞的问题提供了参考。最后，对协议的安全性和性能进行了分析。

参 考 文 献

[1] Armando A, Carbone R, Compagna L, et al. An authentication flaw in browser-based single sign-on protocols: Impact and remediations[J]. Computers & Security, 2012, 33: 41-58

[2] Lutz D J, Stiller B. Combining identity federation with payment: the SAML-based payment protocol[C]// 2010 IEEE/IFIP Network Operations and Management Symposium (NOMS). 2010: 495-502

[3] 唐利娟. SAML 及 SSO 研究与企业化 SSO 框架设计[D]. 济南：山东大学，2011

[4] 陈天玉. 基于 Web Service 的单点登录认证模型的研究与实现[D]. 长沙：湖南大学，2010

[5] 何倩，王芳，柴华昕，等. Web 服务认证技术综述[J]. 桂林电子科技大学学报，2013，33(3)：246-252

[6] 邱罡，张崇，周利华. 基于可信计算的 Web 单点登录方案[J]. 计算机科学，2010，37(9)：121-123

[7] 尹星. 基于 SAML 的单点登录模型及安全的研究与实现[D]. 镇江：江苏大学，2005

[8] 王曦，张斌. 基于代理签名的 SAML 单点登录协议[J]. 计算机工程，2012，38(16)：130-133

[9] 王亚弟，束妮娜，韩继红，等. 密码协议形式化分析[M]. 北京：机械工业出版社，2007：169-180

[10] 谢琪，吴吉义，等. 云计算中基于可转换代理签密的可证安全的认证协议[J]. 中国科学，2012，42(3)：303-313

[11] 孙华，郑雪峰. 一种可证安全的有效无证书签密方案[J]. 计算机科学，2013，40(11)：112-116

[12] Nicolosi A, Krohn M, Dodis Y, et al. Proactive two-party signatures for user authentication[C]// Proceedings of the Network and Distributed System Security Symposium. San Diego, 2003

(下转第 115 页)

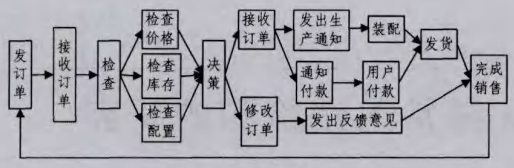

图 5　计算机销售过程的工作流程

因为算法 1 和算法 4 是基于模型 WSCPAM 提出的，所以首先需要生成测试用例对应的 WSCPAM 模型（见图 6）。该 WSCPAM 模型具有 19 个库所和 15 个变迁。

对图 6 所示 WSCPAM 模型执行结构验证算法得到的输出结果如图 7 所示。

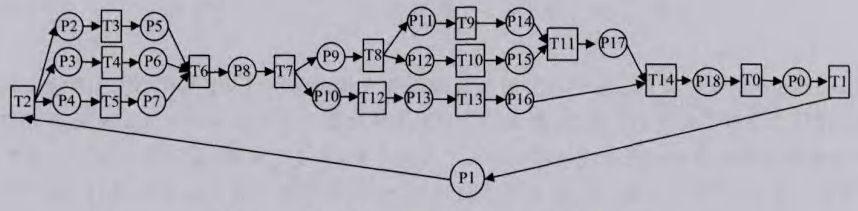

图 6　计算机销售过程对应的 WSCPAM 模型

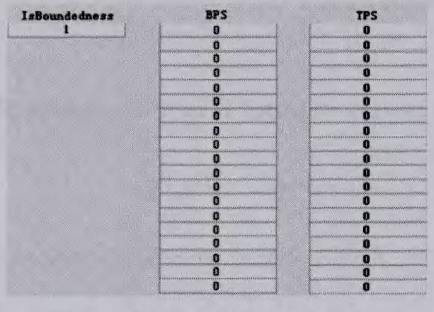

图 7　结构验证测试用例的输出结果

图中变量 *IsBoundedness* 的值为 1 表示 WSCPAM 模型是有界的，为 0 表示 WSCPAM 模型不满足有界性；数组 *BPS* 用来存储死锁覆盖的库所集，$BPS[i](0 \leqslant i \leqslant 19)$ 的值为 1 表示编号为 i 的库所在死锁的覆盖范围内，为 0 表示编号为 i 的库所不在死锁的覆盖范围内；数组 *TPS* 用来存储陷阱覆盖的库所集，$TPS[i](0 \leqslant i \leqslant 19)$ 的值为 1 表示编号为 i 的库所在陷阱的覆盖范围内，为 0 表示编号为 i 的库所不在陷阱的覆盖范围内。从图 7 所示的结果来看，图 5 所示的 Web 服务组合是有界的并且不存在死锁和陷阱。根据实际运行结果得出的结论与预期结论相同。

结束语　为了保证服务组合具有优良结构，本文在基于随机 Petri 网的 Web 服务组合模型 WSCPAM 的基础上，提出了对模型进行有界性、死锁、陷阱验证的必要性、意义和算法实现。

参 考 文 献

[1] 王晶，戎玫，张广泉，等. 基于概率模型检测的 Web 服务组合验证[J]. 计算机科学，2012，39(1)：120-123

[2] 雷丽晖，段振华. 一种基于扩展有限自动机验证组合 Web 服务的方法[J]. 软件学报，2007，18(12)：2980-2990

[3] 骆翔宇，谭征，苏开乐，等. 一种基于认知模型检测的 Web 服务组合验证方法[J]. 计算机学报，2011，34(6)：1041-1061

[4] Wombacher A, Fankhauser P, Mahleko B, et al. Matchmaking for business processes based on choreographies [C] // 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service. IEEE, 2004：359-368

[5] 李景霞，闫春钢. 一种基于扩展颜色 Petri 网的 Web 服务组合验证机制[J]. 计算机科学，2009，36(10)：146-149

[6] 陈丁剑，吴健，马满福，等. 基于 Petri 网的 Web 服务组合建模[J]. 计算机科学，2006，33(5)：128-130，135

[7] Narayanan S, McIlraith S. Analysis and simulation of web services[J]. Computer Networks, 2003, 42(5)：675-693

[8] Overdick H, Puhlmann F, Weske M. Towards a formal model for agile service discovery and integration[OL]. http://frapu.net/pdf/icsoc2005.pdf

[9] 廖军，谭浩，刘锦德. 基于 Pi-演算的 Web 服务组合的描述和验证[J]. 计算机学报，2005，28(4)：635-643

[10] 林闯. 随机 Petri 网和系统性能评价(第 2 版)[M]. 北京：清华大学出版社，2005

[11] 何炎祥，沈华. 随机 Petri 网模型到马尔可夫链的转换规则与实现[J]. 计算机科学与探索，2013，7(1)：55-62

[12] 何炎祥，沈华. 一种基于随机 Petri 网的 Web 服务组合性能瓶颈定位策略[J]. 计算机学报，2013，36(10)：1953-1966

[13] 吴哲辉. Petri 网导论[M]. 北京：机械工业出版社，2006：28-30，88-91

[14] 李扬. 求解线性方程组及不等式组的 ABS 方法[D]. 大连：辽宁师范大学，2007

[15] Murata T. Petri nets: Properties, analysis and applications [J]. Proceedings of the IEEE, 1989, 77(4)：541-580

(上接第 110 页)

[13] Chen L, Chen Z, Smart N P. Identity-based key agreement protocols from pairings[J]. Int J Inf Sec, 2007, 6：213-241

[14] Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels[C]// Advances in Cryptogy(EUROCYPT'01). London: Springer-Verlag, 2001：453-474

[15] Bellare M, Canetti R, Krawczyk H. A modular approach to the design and analysis of authentication and key-exchange protocols [J]. 30th STOC. 1998：419-428

[16] Mitchell C J, Ward M, Wilson P. Key control in key agree- ment protocols[J]. Electronics Letters, 1998, 34：980-981

[17] Guhe C G. An identity-based key-exchange protocol[C]// Proceedings of the Eurocrypt 89. Belgium, 1990：29-37

[18] Juang W S, Chiu J Y, Chang H Y. A secure and efficient delegation-based authentication scheme in public clouds[C]// The 1st Cross-Straits Conference On Information Security. Hangzhou, 2011：96-102

[19] Clom, Michael. Pairing Calculation on super singular Genus Curves[C]// Proceedings of the 13th International Conference on Selected Areas in Cryptography(SAC'06). 2006