维普资讯 http://www.cqvip.com

一类具有"开端"结构的安全协议分析方法研究*)

董荣胜[1] 吴光伟[1,2] 骆翔宇[1] 古天龙[1]
(桂林电子科技大学计算机系 桂林 541004)[1] (中南林业科技大学计算机科学学院 长沙 410004)[2]

摘 要 开端协议(Open-ended Protocol)的分析是安全协议领域中一个待解决的重要问题,而 IKE 则是一个有代表性的具有"开端"结构的安全协议。本文基于串空间的 Athena 方法,针对 IKEv2 协议中的"开端"结构涉及的 DH(Diffie-Hellman)问题,增加了群、散列函数等原语,给出并证明了一个新的消减规则;针对"开端"结构,引入了集合的数学概念,建立了新的消息类型,重新定义了串空间中的消息项、替换、入侵者模型,以及 Athena 相应的内在项、目标和目标绑定,给出了一个新的替代关系。应用扩展后的方法,分析了协议,发现一个新的认证性缺陷,给出了解决该缺陷的方法。

关键词 开端协议,Diffie-hellman,Security association,串空间,Athena

Formal Analysis Approach for a Kind of Open-ended Protocols

DONG Rong-sheng[1] WU Guang-wei[1,2] LUO Xiang-yu[1] GU Tian-long[1]
(School of Computer Science,Guilin University of Electronic Technology,Guilin 541004,China)[1]
(School of Computer Science,Central South University of Forestry and Technology,Changsha 410004,China)[2]

Abstract The formal analysis of Open-ended Protocols is one of emerging areas of research. The IKEv2 protocol offers a complex example. Our work is based on Athena which is an efficient automatic checking approach. We introduce set theory to deal with Open-ended structure. The new notion of message terms, substitution and the penetrator model are given and the relevant propositions or theorem such as interm relation, goal and goal-binding are therefore modified and proved. A new relation is proposed which is used in order to keep the character of Athena approach. We also introduce the ability on modeling the DH problem to Athena according to IKEv2 protocol, a new pruning theorem is given and proved. Based on the research, we analyze IKEv2 protocol, and find a new weakness of authentication. An amendment is given.

Keywords Open-ended protocol, Diffie-hellman, Security association, Strand spaces, Athena

1 引言

就安全协议分析而言,一个基本的任务是刻画协议的消息结构。复杂协议复杂的消息结构直接导致其分析的困难性。为此,Meadows 将一类具有不固定数据结构的安全协议称为开端(Open Ended)协议,并将其列为目前安全协议领域待解决的重要问题。目前"开端"协议的分析还处于起步阶段,主要集中在群协议(Group Protocol)的分析中,并以定理证明为主,缺乏自动性[1-3]。

开端协议的结构不固定,不便展开工作。而针对一个具体的开端协议,则有助于研究工作的开展,并有可能从中提取解决这一类问题的一般方法。IKE 协议是一个具有代表性的开端协议,Meadows 在文献[1]中讨论了它的开端性质,并介绍了 Zhou 以及 Ferguson 和 Schneider 等人针对 IKE 中 SA (Security Association)使用非形式化的方法发现的一个认证性攻击[1,4,5]。IKEv2 是 IKE 的修改版,2002 年由 IETF 的 IPsec 工作组提出,协议更加紧凑和安全,避免了此类攻击[6]。

IKEv2 协议作为一个开端协议,其消息项结构复杂,且要处理 DH(Diffie-Hellman)密钥协商问题[7]。本文为了自动地分析 IKEv2 协议,选用了基于串空间的 Athena 方法,Athena 方法是一种基于串空间模型,结合模型检验和定理证明中各种技术的高效自动的安全协议分析方法[8]。

原始的 Athena 方法消息结构比较简单,没有抽象更多的密码学原语,因而无法处理较为复杂的协议。为此,本文增加了群、散列函数等原语,以处理 IKEv2 中涉及到的 DH 问题;使用集合等概念,增加新的数据项类型,使数据项具有一定程度的"开端"性质,以处理 IKEv2 中的 *SA* 开端问题。同时,对 Athena 方法中涉及的有关内容也进行了扩展。

2 串空间模型的扩展

对 Athena 方法进行扩展,涉及到其基础串空间模型的扩展[9]。本文的扩展建立在 Dolev-Yao 模型的基础上。

2.1 DH 问题的描述

定义 1 设 G 为一个循环群,g 为群 G 的生成元,x,y 为集合$\{1,2,\cdots,|G|\}$中抽取的随机数,则 DH 问题的形式化描述如下:

(1)A 随机选择一个 x,计算 g^x 并发送给 B;

(2)在(1)的基础上,B 同样随机选择一个 y,计算 g^y 发送给 A;

(3)A 可以由所得的 g^y 以及自身所知的 x,通过计算 $(g^y)^x$ 得到 g^{xy};

(4)同样,B 可以由所得的 g^x 以及自身所知的 y,通过计

*)本文得到国家自然科学基金(60763004),广西自然科学基金项目(0542052)的资助。

维普资讯 http://www.cqvip.com

算$(g^x)^y$得到g^{xy}。

为使分析更具一般性，本文添加：

(5)任意主体C可以由所得的g,x,y，计算得到g^x,g^y,g^{xy}。

根据Dolev-Yao模型的假设，协议中采用的密码系统是完善的，不考虑密码系统被攻破的情况，假设：

公理1 $m_1,m_2\in A$且$K_1,K_2\in K$，那么$\{m_1\}_{K1}=\{m_2\}_{K2}\Leftrightarrow m_1=m_2\wedge K_1=K_2$；

公理2 $g^{x_1y_1}=g^{x_2y_2}\Leftrightarrow g^{x_1}=g^{x_2}\wedge g^{y_1}=g^{y_2}$；

公理3 $g_1^x=g_2^x\Leftrightarrow g_1=g_2\wedge x=y$。

在本文中，g^x,g^y,g^{xy}表示为变量DH值，计算g^{xy}用DH运算表示。通常，通信双方对g^{xy}进行散列计算，形成通信双方共享的对称密钥。散列函数是一种将任意长度的消息压缩到某一固定消息长度的消息摘要函数，是普遍采用的认证和数据完成性保护的密码算法。在本文中，为了便于讨论且不失一般性，采用钥控散列函数。

2.2 串空间模型的扩展

串空间模型的扩展包括三个方面：消息项的扩展、替换概念的扩展以及入侵者模型的扩展。

2.2.1 消息项的扩展

定义2 消息项A中原子项的集合扩展为三个部分：正文项T(Text terms)、密钥项K(Key terms)以及DH值D。

(1)正文项T包括各种不同类型项，用t表示，包括：

① 主体的名字；

② 现时值；

③ 集合$\{1,2,\cdots,|G|\}$中抽取的随机整数。

其中③为本文新增。这里，随机整数具有新鲜性。

(2)密钥项K包含了各种类型的密钥，用k表示，包括

① 对称密钥；

② 非对称密钥；

③ 求散列函数(hash)所需的密钥。

其中③为本文新增。在非对称的密钥系统，k是公共密钥簿上的加密密钥，k^{-1}则是与其对应的私有解密密钥。在对称密钥中$k^{-1}=k$。

(3)DH值D包括了通过计算g得到的各种DH值，用d表示。包括：g,g^x,g^y,g^{xy}等。

本文新增(3)。

在定义2基础上，本文新增加一个开端项U。

定义3 开端项U是一个集合，分别由以下三种单一种类的原子项组成：

①由正文项T中某种类型组成，用Ut表示；

②由密钥项K中某种类型组成，用Uk表示；

③由DH值D组成，用Ud表示。

集合中元素个数既可为空，也可为无穷。

定义4 消息项A中合成项集合可由下面操作所得：

(1)加密：$K\times A\rightarrow A,K\times U\rightarrow A$；

(2)密钥逆运算：$K\rightarrow K$；

(3)连接运算：$A\times A\rightarrow A,U\times U\rightarrow A,A\times U\rightarrow A$；

(4)散列函数运算：$A\times K\rightarrow K$(对称密钥)；

(5)DH运算：$D\times D\rightarrow D$。

加密运算用$\{m\}_k$表示，值域为E；连接运算用mm'表示，值域为C；散列函数运算用$H_k\{m\}$表示，值域为$H(H\in K)$。

公理4 $T\cap K=\emptyset,D\cap K=\emptyset,T\cap D=\emptyset$

这里引入一个新的自由假设公理：

公理5 $H_k(m)=H_{k'}(m')\Leftrightarrow k=k'\wedge m=m'$

公理6 $E\cap C=E\cap H=C\cap H=\emptyset,E\cap T=E\cap K=E\cap D=\emptyset,C\cap T=C\cap K=C\cap D=\emptyset$。

2.2.2 替换

本文在串空间中增加了开端项U，相应地，替换概念重新定义如下。

定义5 替换表示变量到项的映射(mapping)。映射分为两种情况：

①若项非U，替换$\sigma=[\vec{t}/\vec{x}]$表示变量$\vec{x}$到项$\vec{t}$的映射(mapping)。

②若项为U，替换$\sigma=[\vec{t}/\overrightarrow{x1x2\cdots xn}]$表示集合$\{x_1x_2\cdots x_n\}$到项$\vec{t}$的映射，$n\geqslant 0$。

定义6 对于项t和t'，如果有$\sigma(t)=\sigma(t')$，那么称替换σ为统一者(unifer)。

定义7 如果对项t和t'，替换σ是一个统一者，并且对其它所有的统一者，存在一个替换γ，使得$\sigma'=\sigma\circ\gamma$，那么称替换$\sigma$是最为普遍的统一者MGU(most general unifier)。

2.2.3 入侵者模型的修改

入侵者模型修改包括两个方面：入侵者初始知识的修改和入侵者角色的修改。

定义8 合法参与者初始知识由下面几个部分组成：

(1)所有主体的姓名；

(2)所有公共密钥、自身的私有密钥，以及作为协议运行主体，通过协议规则获得的对称密钥；

(3)作为合法参与者与其它参与者所共享的群。

本文新增(3)。

在定义8的基础上，给出入侵者初始知识的定义。

定义9 入侵者初始知识init-info(P)定义如下：

(1)入侵者作为合法参与者获得初始知识；

(2)入侵者之间的知识是共享的。

入侵者可以通过初始知识以及截获的知识，获取新的知识。

本文用Kp表示入侵者所知密钥，Dp表示入侵者所知DH值。

定义10 入侵者的角色如下，其中g和h是项，“－”表示接收，“＋”表示发送：

(1)$M[t]$ Atomic message：$\langle+t\rangle$，$t\in$ init-info(P)并且$t\in T$；

(2)$F[g]$ Flushing：$\langle-g\rangle$；

(3)$T[g]$ Tee：$\langle-g,+g,+g\rangle$；

(4)$V[g,h]$ Concatenation：$\langle-g,-h,+gh\rangle$；

(5)$R[g,h]$ Separation into components：$\langle-gh,+g,+h\rangle$；

(6)$K[k]$ Key：$\langle+k\rangle$，这里$K\in Kp$；

(7)$E[k,h]$ Encryption：$\langle-k,-h,+\{h\}_k\rangle$；

(8)$D[k,h]$ Decryption：$\langle-k^{-1},-\{h\}_k,+h\rangle$；

(9)$H[k,g]$ Hash：$\langle-k,-g,+H_k(g)\rangle$；

(10)$Dh[d]$ Dh：$\langle+d\rangle$，这里$d\in Dp$；

(11)f函数：$\langle-z1,-z2,-z3,\cdots,-zn,+d\rangle$，$z1,z2,z3,\cdots,zn$为值$Dp$或者是随机整数$x\in\{1,2,\cdots,|G|\}$，$d=f(z1,z2,z3,\cdots,zn)$，其中$f$计算时间的复杂度应在多项式时间内。

角色(9)、(10)、(11)为本文新增。角色(1)包括了角色(6)和(10)，为了清晰，单独列出。(11)描述的是入侵者构造

维普资讯 http://www.cqvip.com

DH 值的能力。

3 Athena 方法的扩展

Athena 方法的扩展包括四个部分：内在项关系的扩展；目标、目标绑定的扩展；下一个函数的扩展；消减规则的添加。

3.1 内在项关系的扩展

定义 11 内在项关系$\sqsubset$中定义如下，其中 $n\geqslant 0$：

(1)$a\sqsubset t$，其中 $t\in T$，当且仅当 $a=t$；

(2)$a\sqsubset k$，其中 $k\in K$，当且仅当 $a=k$

(3)$a\sqsubset d$，其中 $d\in D$，当且仅当 $a=d$；

(4)$a\sqsubset u$，其中 u 为 Ut 类型，当且仅当 $a\in T$；

(5)$a\sqsubset u$，其中 u 为 Uk 类型，当且仅当 $a\in K$；

(6)$a\sqsubset u$，其中 u 为 Ud 类型，当且仅当 $d\in D$；

(7)$a\sqsubset H_k\{g\}$，当且仅当 $a=H_k\{g\}$；

(8)$a\sqsubset gh$，当且仅当 $a\sqsubset g\vee a\sqsubset h\vee a=gh$；

(9)$a\sqsubset\{g\}_k$，其中 g 不包含 U，当且仅当 $a=\{g\}_k$；

(10)$a\sqsubset\{guh\}_k$，其中 $u\in Ut$，当且仅当 $a=\{gt_1\cdots t_nh\}_k$，其中 t 为正文项的某种类型；

(11)$a\sqsubset\{guh\}_k$，其中 $u\in Uk$，当且仅当 $a=\{gk_1\cdots k_nh\}_k$，其中 k 为密钥项的某种类型；

(12)$a\sqsubset\{guh\}_k$，其中 $u\in Ud$，当且仅当 $a=\{gd_1\cdots d_nh\}_k$，其中 d 为 DH 值。

3.2 目标以及目标绑定的扩展

定义 12 目标是一个二元组(t,n)，这里，$\mathrm{sign}(n)=-$，$t\sqsubset\mathrm{term}(n)$，并且 t 不是项的连接及项的集合。从 C 中的目标集合是 C 中所有目标的集合，用 $G(C)$来表示。

定义 13 若 C 是一个丛，n'是下面集合的$\preceq_C$ 最小员：

$\Psi=\{m\in C|\ t\sqsubset\mathrm{term}(m),\mathrm{sign}(m)=+,\text{and } m\preceq_C n\}$

则称目标(t,n)绑定到节点 n'。

用 $n'\overset{t}{\twoheadrightarrow}n$ 表示目标绑定关系。节点 n'称为(t,n)的被绑定者(binder)。若 t 在上下文中是清楚的，通常可简写成 $n'\twoheadrightarrow n$。

3.3 后续状态函数的扩展

为了使扩展后的 Athena 方法依然具备自动高效的特点，本文提出了一个新的关系——替代，并在此基础上对后续状态函数做相应的扩展。

定义 14 替代表示的是变量到项的映射(mapping)。映射分为两种情况：

①若项非 U，替代 $\sigma=[\vec{t}/\vec{x}]$表示变量$\vec{x}$到项$\vec{t}$的映射(mapping)。

②若项为 U，替代 $\sigma=[\vec{u}/\overrightarrow{ux1x2\cdots xn}]$表示集合$\{ux_1x_2\cdots x_n\}$到项$\vec{u}$的映射，$n\geqslant 0$。

被替代的项含自由变量，可再次进行替换。

定义 15 位置是一个对$[r,i]$，这里 r 是一个角色(role)(参数化的串)，它代表了给定协议 P 中的合法参与者或者是人侵者，i 是 r 中节点的索引。换言之，位置指定(specifies)了一个特殊(particular)角色的一个特殊节点。

定义 16 对于$[r,i]$，若替代 σ 称为一个关于项 t 统一替代者，则相应的节点 $n=\langle r,i\rangle$有这样的形式 $n=\langle +t''\rangle$，对于 t''和 $t'(t'(\sqsubset t'')$，替代 σ 是统一替代者。换言之，节点 $\sigma(n)$发送了一个项，这个项包括了 $\sigma(t)$。

定义 17 对于$[r,i]$，若替代 σ 是一个关于 t 的最为普遍的统一替代者，则对其它的统一替代者 σ'来说，存在一个替代使得 $\sigma'=\sigma\circ\gamma$。

对于给定的协议 P 和项 t(term)，定义统一替代者的集合 $UP(t)$如下：

$UP(t)=\{([r,i],\sigma)|,\sigma$ 是关于 t 和$[r,i]$的一个最为普遍的统一替代者，其中$[r,i]\in P\cup\Pi\}$，

Π是人侵者模型。因为任何的协议都有有限的正则串和人侵者串，并且每一个串都有有限个节点，集合 $UP(t)$有限。

计算协议 P 状态 $l=\langle S,G,\twoheadrightarrow\rangle$的后继状态：首先选定一个没有绑定的目标 $g\in G$，然后，计算关于项 t 的统一替代者集合 $UP(t)$。对任意一个项 $u=([r,i],\sigma)\in UP(t)$，构造下一个状态如下：

(1) 使得 $Su=(\Rightarrow^{-1})*[\sigma(\langle r,i\rangle)]$是一个(可能是部分)串，这个串以 $\sigma(\langle r,i\rangle)$为结束节点，并且在$\Rightarrow$上回溯闭包；它包括了 $\sigma(\langle r,i\rangle)$节点以及所有 $\sigma(r)$的前驱节点。

(2) 对于任意开始于初始状态 l 的串 $s\in S$，若存在一个替代 γ_s 使得 $\gamma_s(Su)\cap\sigma(s)\neq\phi$，即 $\gamma_s(Su)$和 $\sigma(s)$有相同的节点，那么 $\sigma(s)$是 Su 的一个实例化。构造出一个新的状态 $l'=\langle S',G',\twoheadrightarrow'\rangle$，这里：

1) $S'=\sigma(s)\cup\{\gamma_s(Su)\}$；

2)$\twoheadrightarrow'=\sigma(\twoheadrightarrow)\cup\{\langle s',i\rangle\overset{\sigma(t)}{\twoheadrightarrow}\sigma(g)\}$，这里 $S'=\sigma(s)\cup\{\gamma_s(Su)\}$

(目标 g 被串 s'的第 i 个节点绑定)；

3) G'根据上面两条进行更新。

(3) 将 Su 作为一个新的串，构造另一种情况的后续状态(additional next state)

$l''=\langle S'',G'',\twoheadrightarrow''\rangle$，

1) $S''=\sigma(s)\cup Su$；

2)$\twoheadrightarrow''=\sigma(\twoheadrightarrow)\cup\{\langle Su,i\rangle\overset{\sigma(t)}{\twoheadrightarrow}\sigma(g)\}$

(目标 g 绑定在 Su 上的最后一个节点)；

3) G''根据上面两条进行更新；

所有后继状态的集合定义为 $L'=\bigcup_{u\in UP(t)}L'u$。

3.4 消减规则的添加

针对扩展后的串空间，本文提出一个新的消减规则，即定理 1，并给出其证明。

定理 1 C 是一个丛，$d_a,d_b\in D$，若 a,b,d_{ab} 从未在正则串中产生，则对任意的人侵者节点 $p\in C$，$d_{ab}\not\sqsubset\mathrm{term}(p)$。

证明：假设 $d_{ab}\sqsubset\mathrm{term}(p)$，根据人侵者角色的定义 10，人侵者有两种途径获得 d_{ab}：

(1)人侵者节点或者是通过角色 F 直接截获 d_{ab}；

(2)人侵者根据截获的知识通过角色函数 f 构造出 d_{ab}。

先考虑途径(1)。根据假设，d_{ab}从未在正则串中产生，则人侵者通过 F 截获 d_{ab}是不可能的。考虑途径(2)，人侵者构造出 d_{ab}。根据定义 10，f 计算时间的复杂度应在多项式时间内，再根据定义 1、公理 2 和 3 可知，人侵者需要获得的知识可为$\{a,d_b\}$，$\{b,d_a\}$或$\{a,b,d\}$。这与假设相矛盾。

因此 $d_{ab}\not\sqsubset\mathrm{term}(p)$。

4 实例分析

IKEv2(Internet Key Exchange Protocol)协议是目前被广泛应用的互联网密钥交换协议。该协议具有丰富的密码学原语、复杂的“开端”数据结构。本文在保持 IKEv2 协议“开端”性质的基础上，对其认证性进行分析。

4.1 IKEv2 协议

维普资讯 http://www.cqvip.com

IKE_SA_INIT 交换和 IKE_AUTH 交换是 IKEv2 协议的核心部分。本文主要对 IKE_SA_INIT 交换和 IKE_AUTH 交换进行分析。首先对协议进行简化。

构成 IKE_SA_INIT 交换的两条消息如下：

(M1) $I \rightarrow R$: HDR, SA_i1, KE_i, N_i

(M2) $R \rightarrow I$: HDR, SA_r1, KE_r, N_r, [CERTREQ]

HDR 是消息的头部，用于对消息进行标记，指明协议的版本。

SA_i1, SA_r1 分别表示由初始者提出的 SA 和和响应者接受的 SA，SA_i 中包含了一个或者多个提案(proposal)，提案规定了为建立和保护该 SA 而使用的散列函数以及使用何种 DH 群等(根据约定规则选择共享的群，而不是发送该群，目的是阻止入侵者破解获得共享群)；响应者根据 SA_i 及自己的能力作出选择，生成 SA_r1。SA 显然具有"开端"性质。

KE_i 和 KE_r 分别表示初始者和响应者选择的 DH 值。

N_i 和 N_r 表示现时值(nonce)。

CERTREQ 表示证书请求，加"[ ]"表示括号内的信息为可选。

协议简化如下：

(M1) $I \rightarrow R$: Uk_i, g^x, N_i

(M2) $R \rightarrow I$: k_i, g^y, N_r

其中，去除 HDR。根据定义 9，10 以及 IKEv2 协议，可知入侵者无法获得协议合法双方共享的群，无须选择共享群，因此将 SA_i1 简化成一组散列函数的密钥，该结构具有"开端"性质，用开端项 Uk_i 来表示。相应地，SA_r1 用散列密钥 k_i 进行替换。KE_i 和 KE_r 依然表示初始者和响应者选择的 DH 值 g^x 和 g^y。

在 IKE_SA_INIT 交换成功之后，通信双方分别根据所得消息项计算出各种密钥。IKE 协议的密钥计算十分复杂，目的是防止散列函数被破解。根据公理 7，散列函数无法被攻破。因此，将该密钥计算化简为

$SK = H_k(N_i, N_r, g^{xy})$

构成 IKE_AUTH 消息交换的两条消息如下：

(M3) $I \rightarrow R$: HDR, $sk\{ ID_i$, [CERT,] [CERTREQ,] [ID_r,] AUTH, SA_i2, TS_i, $TS_r\}$

(M4) $R \rightarrow I$: HDR, $sk\{ ID_r$, [CERT,] AUTH, SA_r2, TS_i, $TS_r\}$

因为 KE_SA_INIT 交换和 IKE_AUTH 交换中，IKE_AUTH 消息交换的 SA_i2, SA_r2 未被用到，本文将其考虑为非开端项，(M3)和(M4)可简化成：

(M3) $I \rightarrow R$: $\{ I, N_r, SA_i2, TS_i, TS_r\}_{SK}$

(M4) $R \rightarrow I$: $\{ R, N_i, SA_r2, TS_i, TS_r\}_{SK}$

4.2 认证性分析

首先，对密钥的秘密性进行证明，即证第三方无法获得密钥 SK，也就是证明密钥 $SK0 = H_{k1}(N_{i0}, N_{r0}, g^{x0y0})$ 不被第三方得到。因篇幅原因，仅对该证明进行简要说明。

根据定理 1，DH 值 g^{x0y0} 无法获得。因此，$SK0$ 无法被入侵者获得。根据定义 2，随机整数 x, y 具有新鲜性，于是其它正则节点无法构造出与 g^{x0y0} 相同的 g^{xy}，因此除通信双方外，其他主体无法获得该密钥 $SK0$。密钥的秘密性得证。

对 IKEv2 协议的认证性进行证明。假设响应者接收到的 Uk_i 为 (k_{i0}, k_{j0})，并选择其中 k_{i0} 为散列密钥，即证明在 $B[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, k_{j0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}]$ 串存在的情况下，$A[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, k_{j0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}]$ 串前三步必然存在。

修改后的 Athena 方法未改变其搜索步骤。分析步骤参照文献[8]。因篇幅的原因，本文做了一定的简化。

首先使用 init 规则，其中使用了替代的定义，形成状态 l_0。

l_0:

$$B[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, k_{j0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}]$$

$$n_1: \langle -(k_{i0}k_{j0})g^{x0}N_{i0} \rangle$$

$$\Downarrow$$

$$n2: \langle +(k_{i0})g^{y0}N_{r0} \rangle$$

$$\Downarrow$$

$$n_3: \langle -\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

$$\Downarrow$$

$$n_4: \langle +\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

第二步，使用 split 规则。绑定目标 $\langle n_3, -\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$ 出现两种情况：

(1)被绑定的目标是入侵者节点 P，形成状态 l_1；

l_1:

$$B[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, k_{j0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}]$$

$$\langle -(k_{i0}k_{j0})g^{x0}N_{i0} \rangle$$

$$\Downarrow$$

$$\langle +(k_{i0})g^{y0}N_{r0} \rangle$$

$$\Downarrow \qquad\qquad P$$

$$\langle -\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle \leftarrow \langle +\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

$$\Downarrow$$

$$\langle +\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

(2)被绑定的目标为正则节点，其中使用了替代的定义，形成 l_2。

l_2:

$$B[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, k_{j0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}] \qquad A[A, B, N_{i0}, N_{r0}, g^{x0}, g^{y0}, (k_{i0}, Uk_{i0}), k_{i0}, SA_{i0}2, SA_{r0}2, TS_{i0}, TS_{r0}]$$

$$\langle -(k_{i0}k_{j0})g^{x0}N_{i0} \rangle \qquad \langle +(k_{i0}, Uk_{j0}), g^{x0}, N_{i0} \rangle$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\langle +(k_{i0})g^{y0}N_{r0} \rangle \qquad n_5: \langle -(k_{i0})g^{y0}N_{r0} \rangle$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\langle -\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle \leftarrow \langle +\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

$$\Downarrow$$

$$\langle +\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0} \rangle$$

从密钥的秘密性可知，入侵者无法伪造消息 $\{ A, N_{r0}, SA_{i0}2, TS_{i0}, TS_{r0}\}_{SK0}$，状态 l_1 被消减。

第三步，继续使用 split 规则，对 $\langle n_5, -(k_{i0})g^{y0}\ N_{r0} \rangle$ 进行绑定。本文作出一定的简化。需要绑定的目标项有三个，分别为 k_{i0}, g^{y0}, N_{r0}。协议发起者 A 与协议响应者 B 已经形成了通信的密钥 $SK0$，根据密钥的形成规则，根据密钥的秘密性，绑定目标必绑定到响应者 B 串中，形成状态 l_3。

l_3:

维普资讯 http://www.cqvip.com

$B[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$n_6:\langle-(k_{i0}k_{j0})g^{x0}N_{i0}\rangle$

$A[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},Uk_{j0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$\langle+(k_{i0},Uk_{i0}),g^{x0},N_{i0}\rangle$

$\Downarrow$ $\Downarrow$

$\langle+(k_{i0})g^{y0}N_{r0}\rangle \twoheadrightarrow \langle-k_{i0}g^{y0}N_{r0}\rangle$

$\Downarrow$ $\Downarrow$

$\langle-\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle \twoheadleftarrow \langle+\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

$\Downarrow$

$\langle+\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

第四步，对 n_6 中(k_{i0},k_{j0})继续使用 split 规则(目标项不能为连接项，为节省篇幅，这里将目标 k_{i0},k_{j0} 一起讨论)。形成多个状态，如 l_4,l_5。

l_4：

$B[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$\langle-(k_{i0}k_{j0})g^{x0}N_{i0}\rangle$

$A[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0},Uk_{i0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$\langle+(k_{i0}k_{j0},Uk_{i0}),g^{x0},N_{i0}\rangle$

$\leftarrow$

$\Downarrow$ $\Downarrow$

$\langle+(k_{i0})g^{y0}N_{r0}\rangle \twoheadrightarrow \langle-k_{i0}g^{y0}N_{r0}\rangle$

$\Downarrow$ $\Downarrow$

$\langle-\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle \twoheadleftarrow \langle+\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

$\Downarrow$

$\langle+\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

l_5：

$I[I,B,N_i,N_r,g^x,g^y,(k_{i0},k_{j0},Uk_{j0}),$
$ki,SA_i2,SA_r2,TS_i,TS_r]$
$\langle+(k_{i0},k_{j0},Uk_{i0}),g^{x0},N_{i0}\rangle$

$\twoheadrightarrow$

$B[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$\langle-(k_{i0}k_{j0})g^{y0}N_{i0}\rangle$

$A[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},Uk_{j0}),$
$k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$
$\langle+(k_{i0},Uk_{i0}),g^{y0},N_{r0}\rangle$

$\Downarrow$ $\Downarrow$

$\cdots$

$\langle+(k_{i0})g^{y0}N_{r0}\rangle \twoheadrightarrow \langle-k_{i0}g^{y0}N_{r0}\rangle$

$\Downarrow$ $\Downarrow$

$\langle-\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle \twoheadleftarrow \langle+\{A\ N_{r0}\ SA_{i0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

$\Downarrow$

$\langle+\{B\ N_{i0}\ SA_{r0}2\ TS_{i0}\ TS_{r0}\}_{SK0}\rangle$

根据状态 l_4,l_5，协议的认证性不成立，即在 $B[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0}),k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$ 串存在的情况下，串 $A[A,B,N_{i0},N_{r0},g^{x0},g^{y0},(k_{i0},k_{j0}),k_{i0},SA_{i0}2,SA_{r0}2,TS_{i0},TS_{r0}]$ 前三步不一定存在。认证性不成立的原因在于发起者提出的 SA 无法被认证，根据状态 l_3，协议只能保证发起者提出的 SA 包含响应者采用的 SA。该缺陷可能导致某些攻击，比如，入侵者删除发起者 SA 集合中某些元素，导致响应者无法采用较为适合的 SA，而协议的发起者和响应者无法意识到这种改变，这便于入侵者对密钥进行破解。为了避免这种攻击风险，本文对密钥的计算进行重新定义，密钥计算化简后为：

$$SK=H_k(N_i,N_r,SA_i1,g^{xy})$$

在密钥计算的中添加 SA_i1 参数，对协议发起者提出的 SA 进行认证。

结束语 本文针对"开端"结构，扩展了基于串空间理论的 Athena 方法，并用扩展后的方法对 IKEv2 协议进行了分析。分析发现，IKEv2 协议中 SA 结构的"开端"性将导致一种原来协议(IKE)中没有的认证性缺陷，即发起者提出的 SA 无法被认证。为此，本文对密钥的计算进行了重新定义，添加了 SA_i1 参数，以便对协议发起者提出的 SA 进行认证。

本文的工作与文献[4,5]的不同，文献[4,5]针对 IKE 中 SA 的具体结构，采用非形式化的方法，发现响应者所选择的 SA 无法被认证的缺陷，IKEv2 协议工作组为此在 SA 中增加了一个新的组件解决了 IKE 存在的这一问题[1,6]。

本文扩展后的方法不仅继承了原有方法的性质，而且还为进一步应用的扩展提供了一个可以借鉴的模式。

参 考 文 献

[1] Meadows C. Formal methods for cryptographic protocol analysis: Emerging issues and trends. IEEE Journal on Selected Areas in Communication, 2003, 21(1): 44-54

[2] Bella G, Massacci F, Paulson L C, et al. Formal verification of cardholder registration in SET // Cuppens F, eds. Computer Security-ESORICS 2000. Springer LNCS 1895, 2000: 159-174

[3] Bryans J, Schneider S. CBS, PVS, and a recursive authentication-protocol // Proceedings of the DIMACS Worksho Pon Design and Formal Verification of Security Protocols. 1997: 3-5

[4] Zhou J. Fixing a security flaw in IKE protocols. Electronics Letters, 1999, 35(13): 1072-1073

[5] Ferguson N, Schneier B. A security evaluation of IPSec // Blaze M, Ioannides J, Keromytis A, et al, eds. The IPSec Papers. Addison-Wesley, to appear

[6] Kaufman C. Internet Key Exchange (IKEv2) Protocol. Internet Draft. http://www. ietf. org/internet-drafts /draft-ietf-ipsec-ikev2-11. txt

[7] Boneh D. The decision Diffie-Hellman problem // Proceedings of the Third Algorithmic Number Theory Symposium, number 1423 in Lecture Notes in Computer Science. Springer-Verlag, 1998: 48-63

[8] Perrig A, Song D, Berezin S. Athena: a novel approach to effcient automatic security protocol. analysis. Journal of Computer Security, 2001, 9(1/2): 47-74

[9] Thayer F J, Herzog J C, Guttman J D. Strand spaces: Why isa security protocol correct // Proceedings of 1998 IEEE Symposium on Security and Privacy. 1998: 160-171