安全协议理论与方法

基于推理的结构性方法

逻辑类分析方法简介

- 逻辑:运用形式化方法研究和判定推理形式有效性。
- 形式化方法：将一套特制的人工符号应用于演绎体系以使其严格化、精确化的研究方法。包含符号化和系统化。
- 符号化：将命题(p, q, r)和常项(如命题连接词)用符号(可重复和可辨认的标记)表示。
- 系统化：在符号化的基础上将一定范围内所有有效的推理形式构成一个形式系统。

逻辑类分析方法简介

- 狭义形式系统:形式语言和演绎装置。
- 形式语言：系统的初始符号(字母表)，形成规则(如何使用符号组成公式)。
- 演绎装置：包括定义、公理和规则。
- 广义形式系统：增加语义部分。即对初始符号、公式和规则的解释。解释可把形式系统与一定模型(代表客观实际)连接起来，从而赋予初始符号和公式一定的实际意义。

逻辑--推理结构性方法简介

推理结构性方法：运用逻辑系统从用户接收和发送的消息出发，通过一系列的推理公理推证协议是否满足其安全说明。

- 基于知识：if you know P, then P is true.
- 基于信仰：不关心P的正确与否。
- 推理公理：从已知知识或信仰推出新知识或信仰。
- 典型：BAN逻辑、Kailer逻辑、RV逻辑。

逻辑--推理结构性方法简介

- 公式:具有一定形成规则,解释后有意义的符号串。
- 定理:表示系统内的重言式(或称永真式)。
- 规则:变形规则。从已有定理产生其他定理的方法。“从A得出B”或者“若A，则B”。A=>B。
- 蕴含：“若…，则…”关系：“实质蕴含”简称“蕴含”，记为“A→B”⇔ ((not A) or B)。

  含义：不是通常逻辑思考中的充分条件关系。

  误解：“伦敦不在英国”蕴含“巴黎在法国”。

逻辑--推理结构性方法特点

1：简洁直观，易于使用。相对于以后所讲的其他形式化分析方法。

2：理想化方法。由于相同的消息在不同的协议中可能代表不同的含义，因此分析协议之前必须对协议进行形式化处理，即用逻辑语言描述。具体的形式化方法没有给出，靠人的经验。

3：使用假设和推理规则。

假设不正确，不能得到正确的信念；

公理和推理规则是否合理和完备也影响性能。

BAN逻辑系统

- 定义：基于主体知识和信念推理的模态逻辑。
- 过程：通过推导主体是否能够从接收到的消息中获得信念来判断协议是否能够达到认证目标。

1. 首先给出一套形式化标记方法，用于对协议消息、初始假设、推理规则和主体信念进行描述。

2. 在公理和推理规则的作用下，从协议的初始假设和消息中蕴含的公式推导出主体的信念，来判断协议是否满足既定的目标。

BAN逻辑系统

消息描述：用形式化方法对协议消息进行描述。

初始假设：对协议外部特性的一些假设，如对可信第三方的信任假设，对共享密钥安全性的假设，对随机数新鲜性的假设等。

推理规则：逻辑类分析方法的核心。

主体信念：表明了主体对信念的理解，是逻辑分析方法的结果。

BAN逻辑系统---所用符号

- A, B, S等:泛指参与协议的主体。
- Kab, kbs, kas：主体之间的共享密钥。
- Ka, kb, ks：代表主体的公开密钥。
- $Ka^{-1}, kb^{-1}, ks^{-1}$：代表对应主体的秘密密钥。
- Na, Nb, Ns：代表主体各自生成的用于确认新鲜性的随机数Nonce。
- P, Q. R：代表主体变量。
- X, Y：代表公式变量。
- K：代表密钥变量。

BAN逻辑系统---所用符号

- sees(P,X)(P sees X)：P接收到X。
- said(P,X)(P said X)：P发送X。
- cont(P,X)(P cont X)：P拥有对X正确与否的判决权。
- fresh(X)(#X)：X是新鲜的。
- skey(P,K,Q)(SharedKey(K,P,Q)):K是PQ共享密钥。
- goodkey(P,K,Q):K是PQ共享的良好密钥。
- pubkey(P, K)(PK(K,P))：K是P的公开密钥。
- secret(P,X,Q)(sec(X,P,Q):X是P和Q的共享秘密。
- {X}k：用密钥K加密X得到的结果。
- <X>Y:X和Y的组合，Y是P和Q之间的秘密。
- P bel X：P相信X。

BAN逻辑---推理规则

- 消息意义规则
- 随机数验证规则
- 仲裁规则
- 信念规则
- 接收规则
- 新鲜规则
- 传递规则

推理规则---消息意义规则

- 从加密消息所使用密钥以及消息中包含的秘密来推断消息发送者的身份(假设P和Q为不同的主体)
- 对于共享密钥， bel(P, goodkey(P, K, Q))and sees(P, {X}k)=>bel(P, said(Q, X))
- 对于私钥， bel(P, pubkey(Q, K))and sees(P, $\{X\}_k^{-1}$)=>bel(P, said(Q, X))
- 对于共享秘密有 bel(P, secret(P, X, Q))and sees(P, $(X)_Y$)=>bel(P, said(Q, X))

推理规则-随机数验证规则(现时检验规则)

- bel(P,fresh(X)) and bel(P,said(Q,X))
  =>bel(P, bel(Q,X))

- 解释：如果P相信X是新鲜的，并且P相信Q曾经发送过X，那么P相信Q相信X。

推理规则---仲裁规则

- 作用：拓展主体的推知能力，使主体可以在基于其他主体已有的信仰之上推知新的信仰。

- bel(P, cont(Q, X)) and bel(P, bel(Q, X))
  => bel(P, X)

- 解释： 如果P相信Q对X是有仲裁权的， 并且P相信Q是相信X的， 那么P相信X。

推理规则---信念规则

- 作用:反映信念在消息的级联与分割的不同操作中的一致性以及信仰在此类操作中的传递性。

$bel(P,X)\ and\ bel(P,Y) \Rightarrow bel(P,(X,\ Y))$

$bel(P,(X,Y)) \Rightarrow bel(P,X)\ OR\ bel(P,Y)$

$bel(P,bel(Q,(X,Y))) \Rightarrow bel(P,bel(Q,X))\ or\ bel(P,bel(Q,Y))$

推理规则---接收规则

■作用：定义了主体在协议运行中获取消息。

sees(P, (X,Y)) ⇒ sees(P, X)

sees(P, <X>Y) ⇒ sees(P, X)

bel(P, goodkey(P,K,Q)) and sees(P, $\{X\}_k$)⇒sees(P,X)

bel(P, pubkey(P,K)) and sees(P, $\{X\}_k$) ⇒sees(P,X)

bel(P, pubkey(Q,K)) and sees(P, $\{X\}_k^{-1}$) ⇒sees(P,X)

■解释：如果P接收到一个消息，P也就收到了这个消息的组成部分，以及P能够从中解出的消息。

推理规则---新鲜规则

bel(P, fresh(X)) $\Rightarrow$ bel(P, fresh(X,Y))

bel(P, fresh(X)) and

bel(P, said(Q,X)) $\Rightarrow$ bel(P, bel(Q,X))

解释：如果公式的一部分是新鲜的, 则整个公式也是新鲜的。以密文出现才有意义，密文中有一部分是新鲜的，则整个部分也是新鲜的。

推理规则---传递规则

bel(P, said(Q, (X,Y)) ⇒ bel(P, said(Q, X))

- 解释：如果P相信Q曾发送过整个消息，那么P相信Q曾发送过消息的 子部分。

BAN逻辑---若干假设

- 时间假设
- 密钥假设
- 主体假设
- 自身消息可识别假设

时间假设

- 协议分析中区分两个时间：

  current-time：起始于本次协议运行的开始阶段。

  past-time： current-time之前的时间。

- current-time：如果某一观点在协议开始时是成立的，那么在整个current-time中也是成立的。但是在past-time中成立的观点在current-time中却并不一定成立。

密钥—主体—假设

- 密钥不能从密文中推导出来。
- 不拥有正确密钥不能解密报文。
- 主体能够知道他是否正确地使用了解密密钥。正确的密钥解密得到的明文有意义，错误的密钥解密得到的明文没有意义。

主体—假设

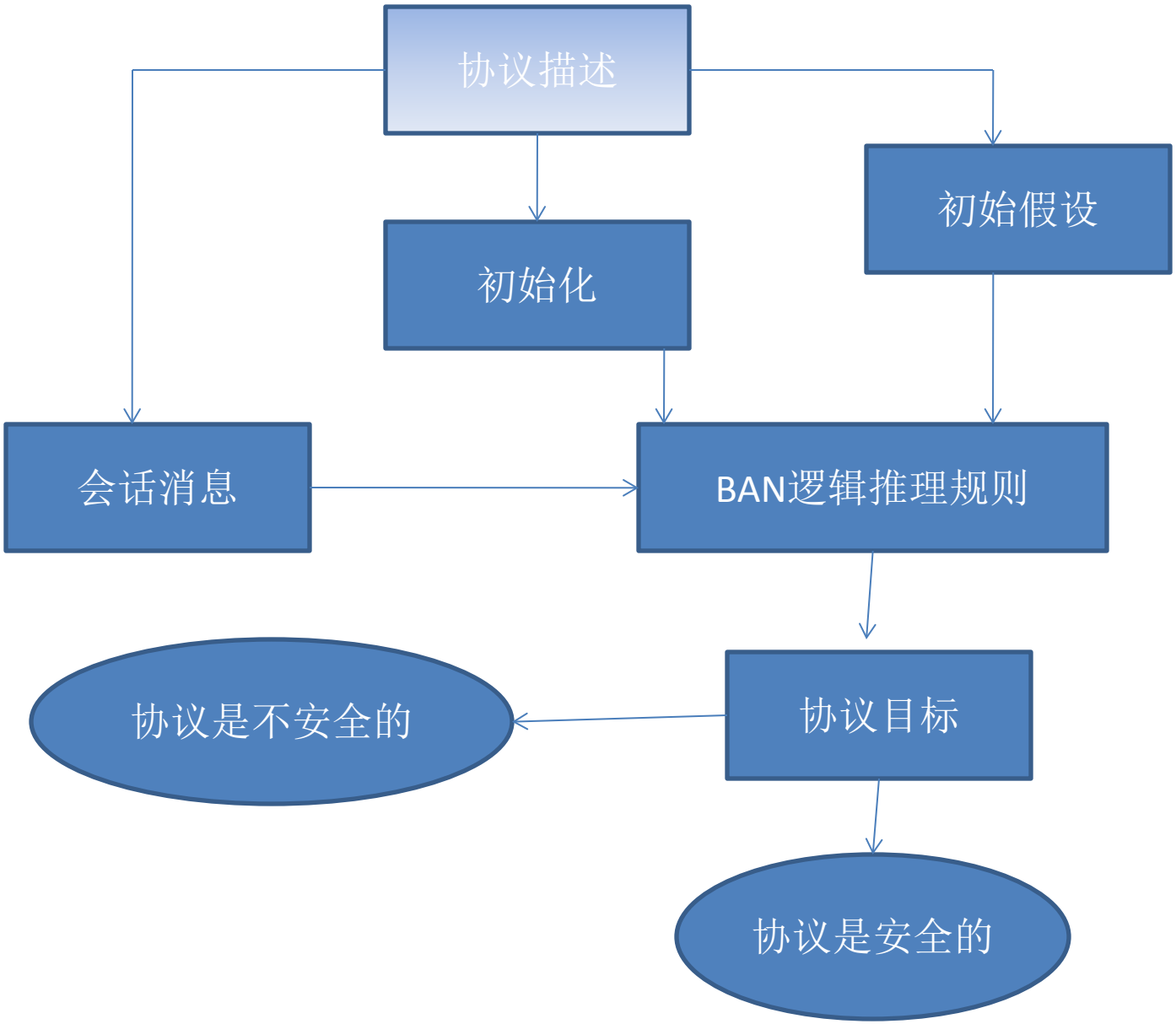
- 假设参与协议运行的主体都是诚实的。

自身消息可识别—假设

- 假设接收方能分辨接收到的消息是否为自己发送过的消息。使得消息含义规则的成立有合理性。

如何应用BAN逻辑

1. 对协议进行理想化预处理（初始化）。
2. 给出协议初始状态及其所基于的假设。
3. 形式化说明协议将达成的安全目标。
4. 运用公理和推理规则以及协议会话事实和假设，从协议的开始进行推证直至验证协议是否满足其最终运行目标。


协议描述
初始假设
初始化
会话消息
BAN逻辑推理规则
协议是不安全的
协议目标
协议是安全的

应用BAN逻辑发现NS协议漏洞

1).A→S: A,B,Na

2).S→A: {Na,B,Kab,{Kab,A}Kbs}Kas

3).A→B: {Kab,A}Kbs

4).B→A: {Nb}Kab

5).A→B: {Nb-1}Kab

NS协议是1978年提出的基于共享密钥体系的协议。使通信双方能互相证实对方身份,并为后继的加密通信建立一个新会话密钥。

NS(Needham-Schroeder)假定

1)三个主体：A,B及其所信赖的认证服务器S。

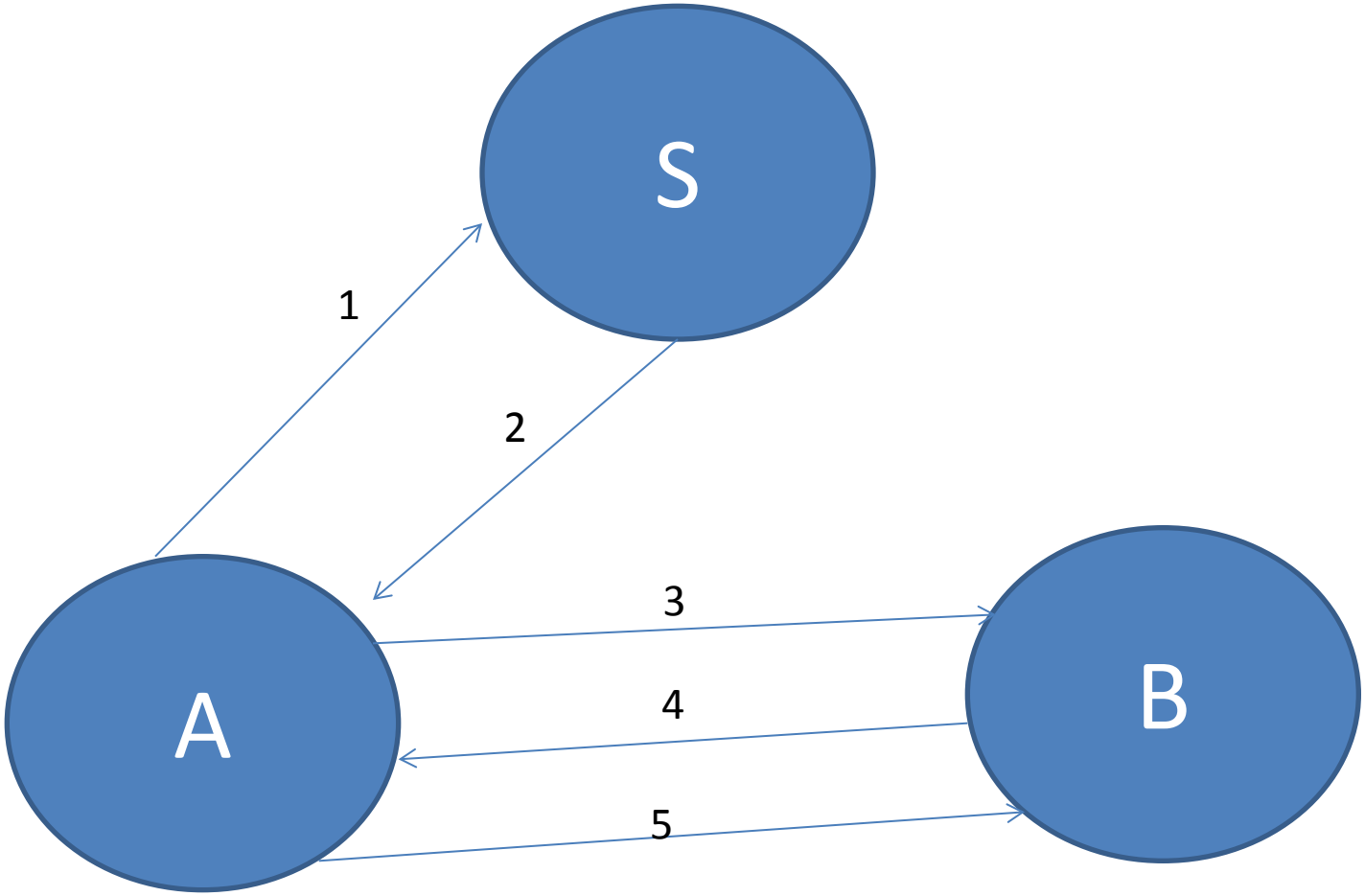
2)Kas：A与S之间的长期共享密钥。

Kbs：B与S之间的长期共享密钥。

Kab：S生成的A与B之间的临时密钥。用于A、B双方认证之后的加密通信-会话密钥。

3)Na,Nb：A和B生成的随机数。

加标识的目的是A可能与多个主体通信。



NS协议

NS协议的目标

在成功地完成协议后，A和B就能准确地确信它们之间拥有了一个仅为它们和可信服务器知道的会话密钥。

但NS协议被证明是有漏洞的，只是不易察觉。

BAN逻辑证明NS协议

(1)理想化协议

M2:S→A: {Na, skey(A,fresh(Kab),B),
{skey(A,Kab,B)}$_{kbs}$}$_{kas}$

M3:A→B: {skey(A,Kab,B)}$_{kbs}$

M4:B→A: {Nb}$_{Kab}$

M5:A→B: {Nb-1}$_{Kab}$

此理想化NS与原协议区别

NS协议的第一个消息被忽略。BAN认为明文是无效的，因为攻击者可编造明文。

将理想化协议改为逻辑公式

P1: sees (A, {Na,skey(Kab,A,B), fresh(skey(Kab,A,B)), (skey(Kab,A,B))kbs}kas

P2: sees(B, (skey(kab,A,B))kbs

P3: sees(A, {Nb,skey(Kab,A,B)}Kab

P4: sees(B, {Nb, skey(Kab,A,B)}Kab

BAN逻辑证明NS协议续

(2) 初始状态及假设-密钥的有效性

(A1) bel(A,goodkey(A,Kas,S))

(A2) bel(B,goodkey(B,kbs,S))

(A3) bel(A, cont(S, skey(Kab,A,B)))

(A4) bel(B, cont(S, skey(Kab,A,B)))

(A5) bel(A, cont(fresh(S, skey(Kab,A,B))))

(A6) bel(A, fresh(Na))

(A7) bel(B, fresh(Nb))

BAN逻辑证明NS协议续

(3) 协议目标

G1: bel(A, goodkey(Kab,A,B))

G2: bel(A, fresh(goodkey(Kab,A,B)))

G3: bel(B, goodkey(Kab,A,B))

G4: bel(B, fresh(goodkey(Kab,A,B)))

G5: bel(B, bel(A,skey(Kab,A,B)))

BAN逻辑证明NS协议续

(4) 逻辑推证

(1) 通过消息含义规则和A1、P2得到：

bel(A, said(S, Na, skey(kab,A,B),
fresh(skey(Kab,A,B)), skey(Kab,A,B)kbs)---P5

(2) 通过新鲜性规则和A6、P5得到：

(A6)： bel(A, fresh(Na))

新鲜性规则： bel(P, fresh(X)) ⇒ bel(P, fresh(X,Y))

bel(A, fresh({skey(Kab,A,B), fresh(Kab,A,B),
skey(Kab,A,B)kbs}))---P6

BAN逻辑证明NS协议续

(3) 通过P5、P6以及现时检验规则得：

bel(A, bel(S, {Na, skey(Kab,A,B),
fresh(skey(Kab,A,B),skey(Kab,A,B)kbs) ---P7

(4) 通过P7和信念连接规则得：

bel(A, bel(S, skey(Kab,A,B))) --- P8

BAN逻辑证明NS协议续

(5) 通过P7和信念连接规则得到：

bel(A, bel(S, fresh(skey(A, Kab,B))))-P9

(6)通过仲裁规则和P8式以及A3得到：

bel(A,skey(Kab,A,B)) ---G1

BAN逻辑证明NS协议续

(7) 通过仲裁规则和P9式以及A5得到：

bel(A,fresh(skey(Kab,A,B)))　---G2

G1,G2为主体A的两个目标。

BAN逻辑证明NS协议续

(8)通过消息含义规则和A2、P2得到：

bel(B, said(S, skey(Kab, A, B)))　---P10

结论：推导到此为止，B确信S发送过A、B之间的一个通信密钥，但他无法确信该密钥的新鲜性。因此该协议可能易受重放攻击。

BAN逻辑的局限性

- 实例Nessett
  1. A→B: (Na, Kab)ka^{-1}
  2. B→A: (Nb)kab

它基于公钥密码假设，不安全性显而易见。

理想化Nessett协议

实例Nessett

1. A→B: {Na, skey(Kab, A, B), fresh(skey(Kab, A, B)}ka^{-1}

2. B→A: {Nb, skey(Kab, A, B)}kab

将理想化协议重写为逻辑公式

- 实例Nessett

P1. sees(B, {Na, skey(Kab, A, B), fresh(skey(Kab, A, B)}ka^{-1})

P2. sees(A, {Nb, skey(Kab, A, B)}kab

Nessett协议目标

- 暂定为

G1: bel(B, skey(Kab, A, B)

G2: bel(B, fresh(skey(Kab, A, B)))

G3: bel(A, bel(B, skey(Kab, A, B)))

Nessett假设集合

- (2) 初始状态及假设-密钥的有效性

(A1) bel(B, PK(A,Ka))

(A2) bel(A, skey(Kab,A,B))

(A3) bel(A, fresh(skey(Kab,A,B)))

(A4) bel(B, fresh(Na))

(A5) bel(B, cont(A, skey(Kab,A,B)))

(A6) bel(B, cont(A,fresh(skey(Kab,A,B))))

(A7) bel(A, fresh(Nb))

BAN逻辑证明Nessett协议

(4) 逻辑推证

(1) 通过消息含义规则和A1、P1得到：

bel(B, said(A, {Na, skey(kab,A,B),

fresh(skey(Kab,A,B)})) ---P3

(2) 通过新鲜性规则和A4得到：

bel(B, fresh({Na, skey(Kab,A,B),

fresh(skey(Kab,A,B))} ---P4

BAN逻辑证明Nessett协议续

(3) 通过P3、P4以及现时检验规则得：

bel(B, bel(A, {Na, skey(Kab,A,B), fresh(skey(Kab,A,B)})) ---P5

(4) 通过信念连接规则和上式P5得：

bel(B, bel(A, skey(Kab,A,B))) -- P6

BAN逻辑证明Nessett协议续

(5)通过仲裁规和上式得到：

bel(B, skey(Kab,A,B)) --- G1

(6)同样可以得到：

bel(B, fresh(skey(Kab,A,B))) ---G2

结论：B相信了Kab是A和B的共享密钥，而且是一个新鲜的密钥。

BAN逻辑证明Nessett协议续

(7)运用消息含义规则和P2、A2得到：

bel(A, said(B, skey(Kab,A,B)))　---P7

(8) 运用现时验证规则、P7和A3得到：

bel(A, bel(B, skey(Kab,A,B)))　---G3

结论：Kab是A和B之间的良好的会话密钥。

BAN证明Nessett的问题

- 问题

  协议第一条消息暴露了自己的密钥，因此bel(A,skey(Kab,A,B))的假设与BAN逻辑的前提是不一致的。

- 解释

  BAN逻辑只考虑认证性，不考虑秘密性。

BAN逻辑理想化的问题

1省略掉对于推知主体信仰无用部分，如明文。理想化后的消息形式为 {X1}k1,…,{Xn}kn。

2 协议的理想化过于依赖分析者直觉。

3 Woo和Lam认为协议的理想化使得原始协议与理想化协议间存在语义鸿沟。

4 协议的理想化是将协议过程语言中对协议主体行为的描述解释为用逻辑语言描述的主体的知识和信仰，并以此来表示协议说明的语义。现有的逻辑形式化分析系统很难解决此问题。

5 BAN证明没有问题，并不能保证该协议没有问题。

BAN逻辑省略明文的缺陷

挑战-响应协议：

1. B→A:　{Nb}Kab
2. A→B:　Nb

解释：1中的为挑战，2为响应，B判断Nb是否和自己发送的相同，来发现问题。若省略明文，则不能用BAN进行分析了。

BAN逻辑缺陷-不合理的假设

- BAN逻辑系统认为参与协议运行的主体都是诚实的。每个主体相信它所发送的消息、随机数验证规则正是基于此，而这一假设并不是非常合理的。一些合理的协议并不满足主体诚实性假设。
- 诚实的界定模糊(应由消息内容决定)。
- 诚实是相对的。在一个环境中诚实，在另一个环境中不一定诚实。

BAN逻辑缺陷-无法检查协议运行的违规现象

- 不能检查协议并发运行带来的攻击

例：BAN-Yahalom密钥分配协议存在并发运行的攻击，但用BAN逻辑分析却是安全的。

- BAN缺少一个良好定义的语义，良好的语义体现在消息意义规则中：bel(P, goodkey(P,K,Q)) and sees(P,$\{X\}_K$)$\Rightarrow$bel(P, said(Q,X))

BAN解释：K是否良好的密钥在于：如果一个密钥K被用于发送消息，那么它就是好的密钥，而不涉及密钥的安全性，这显然不严密。如果其他主体发送这些消息的备份，根据消息意义规则，BAN逻辑是无法发现这类重放攻击的。

BAN逻辑缺陷-未考虑协议运行的顺序

- Snekenss协议

1. $A \rightarrow B: \quad q$
2. $B \rightarrow A: \quad \{Nb, q, a\}kb^{-1}$
3. $A \rightarrow B: \quad \{Nb\}Ka^{-1}$

解释：在控制器A和传感器B之间，A定期地发送询问q给B，B给出响应A，A负责检查Nb的新鲜性。

BAN逻辑缺陷-未考虑协议运行的顺序

Snekenss协议的问题

攻击者可以位于A和B之间，每次截获B给A的第2条消息，如果觉得a不满意，就冒充A多次发送q，直到从第2条消息中获得满意的a，再将其转发给A。结果是蒙骗A，使A不能获得真实和及时的a。

BAN逻辑缺陷-未考虑协议运行的顺序

- Snekenss协议改进

1. A→B:　q
2. A→B:　$\{Na\}Ka^{-1}$
3. B→A:　$\{Na, q, a\}kb^{-1}$

解释：随机数Na的检验由B负责，在控制器A和传感器B之间，A定期地发送询问q给B，B给出响应A，A负责检查Nb的新鲜性。

BAN逻辑改进的方向

- 确立一个可靠的语义, 用以验证初始假设的和确保推理的可靠性。
- 减少理想化步骤模糊度, 进而消除理想化步骤。
- 建立计算机化的自动分析过程，将各类攻击模型化并进行分析。
- 在协议设计阶段，就引入分析从而避免可能发生的设计错误，并确立好的协议设计方法和规则等。

BAN逻辑的设计准则

1）认证协议中认证主体用会话密钥加密另一个认证主体所知的明文消息，来表示已获得了会话密钥。注意：会话密钥必须以新鲜性为保证，避免密钥重放。

2）在传递包含关键的敏感信息的消息中，尽可能以显式表现出参与协议的各个主体，使接受的主体能清晰地意识到其他主体存在。

BAN逻辑的设计准则

- 3）协议中同一个认证主体生成发送的信息结构和接收并能看到的信息结构不要相同。

- 4）在协议中，可信服务器的信仰很重要，因此区分服务器作用的强弱很重要。

- BAN逻辑结构是开放的，因此演化出很多改进的其他逻辑。

GNY逻辑

- 取消了一些全局假设，增加BAN分析能力。如主体诚实性、消息源、可识别性等假设。
- 引入了可识别性的概念，用于描述主体对其所期望的消息格式的识别能力。
- 区分断言集合和符合集合。
- 引入了若干fresh及其变形判断法则。
- 将主体信念集和拥有集加以区别，使逻辑系统显得更为自然。

GNY逻辑的简单计算模型

- 信仰集：主体现有所有的信仰。
- 所有集：主体可得到的所有公式，尤其是包括主体收到的一切信息及生成的信息，如随机数等。主体从一定的最初信仰和初始的所有信息开始会话，然后主体从接受到的信息中扩展自己的所有集和信仰集。
- 在一次会话中，信仰和所有集是单调的。

GNY逻辑的语法和语义

1) (X,Y)表示公式X和Y的合取式。
2) $\{X\}_K$和$\{X\}_K^{-1}$分别表示用对称密钥对消息X进行加密，解密。
3) $\{X\}_{+K}$和$\{X\}_{-K}$分别表示用公开密钥和私有密钥对消息X进行加密和解密。
4) H(X)单向Hash 函数H对消息X作用的值。
5) F(X1，…，Xn)局部转换函数，对任一变量，当其他变量的值固定后所得的函数为一一对应函数。
6) rec(P,X):某一主体向P发送了包含消息X的消息。

GNY逻辑的语法和语义

7) poss(P, X): 主体P拥有消息X。
8) send(P, X): P发送了包含有公式X的消息。
9) bel(P, fresh(X))： P相信X是新鲜的。
10) bel(P,ϕ(X)): P可通过X内容和格式识别X。
11) bel(P,goodkey(P,K,Q))： P相信K是为P与Q共享的良好会话密钥。

GNY逻辑的语法和语义

12)bel(P,pubk(Q,+K)):P相信+K为Q的公开密钥。

13)C1,C2: 表示命题C1，C2的合取式。

14) bel(P,C):P相信命题C为真。

15) cont(P,X)： P对X有管辖权。

16) rec(P,*X)： P可识别X不是由自己首发的。

17) extern(X,C): 命题C是X的消息扩展。

GNY逻辑的推理规则(41条)

- 被告知推理规则(6条)
- 拥有规则(8条)
- 新鲜性规则(11条)
- 识别规则(6条)
- 消息解释规则(7条)
- 管辖规则(3条)

GNY逻辑对协议的分析

GNY逻辑对协议进行分析的步骤如下：

- 1) 对消息标识“不是由此首发”标记*，并对消息做出逻辑可以理解的解释。
- 2) 对系统的初始状态进行描述，给出初始化假设。
- 3) 运用GNY逻辑的推理规则对协议进行形式化分析。

用GNY逻辑发现NS协议漏洞

1).A→S: A,B,Na

2).S→A: {Na,B,Kab,{Kab,A}Kbs}Kas

3).A→B: {Kab,A}Kbs

4).B→A: {Nb}Kab

5).A→B: {Nb-1}Kab

■ NS协议使通信双方能互相证实对方身份，并为后继的加密通信建立一个新会话密钥。

对NS协议消息进行逻辑标注

1) rec(S,(*A, *B, *Na))
2) rec(A,(exten(*{Na,B, *Kab, exten(*{Kab,A}$_{kbs}$, bel(S,goodkey(A,Kab,B))}$_{kas}$,bel(S,goodkey(A,Kab,B)))
3) rec(B,extent(*{*Kab, *A}$_{kbs}$, bel(S,goodkey(A,Kab,B))))
4) rec(A, *{*Nb}kab)
5) rec(B， exten(*{*F(Nb)}$_{kab}$, bel(A,goodkey(A,Kab,B))))

第二步：初始假设

A1: poss(A,Kas) A2: poss(B,Kbs) A3: poss(S,Kas)
A4: poss(A,Na) A5: poss(B,Nb) A6: poss(S,Kbs)
A7: poss(S,Kab)
A8: bel(A,goodkey(A,Kas,S)) A9: bel(B,goodkey(B,Kbs,S))
A10: bel(S,goodkey(A,Kas,S)) A11: bel(S,goodkey(B,Kbs,S))
A12: bel(S,goodkey(A,Kab,B))
A13: bel(A, fresh(Na)) A14: bel(B, fresh(Nb))
A15: bel(A, ϕ(Na)) A16: bel(B, ϕ(Nb))
A17: bel(A, cont(S, goodkey(A,K,B)))
A18: bel(B, cont(S, goodkey(A,K,B)))
A19: bel(A,cont(S, bel(S, *))) A20: bel(B,cont(S, bel(S, *)))
A21: bel(A,cont(B, bel(B, *))) A22: bel(B,cont(A, bel(A, *)))

第三步：运用GNY推理规则对协议进行分析

- GNY逻辑对NS协议的分析结果是[GONY90]：
  - poss(S,(A,B,Na))
  - poss(A,Kab), bel(A,goodkey(A,Kab,B))
  - poss(B,Kab)。B无法判断Kab的新鲜性，并且无法相信消息(3)是由S发出的。其原因在于没有时效性标识。
  - poss(A,Nb)。A无法从这条消息判断B的信念。
  - B无法判断Kab是否为良好的密钥，所以不能推断出A的信念。

因此，最终运用GNY逻辑对协议进行分析只能得到：

- poss(A,Kab), bel(A,goodkey(A,Kab,B)), poss(B,Kab)

AT逻辑

- 评估一个逻辑最关注的两个问题是:

1) 是否能够得到所有想得到的(完备性)。
2) 是否能够回避所有不想得到的(合理性)。很多逻辑是基于生成所有可能的协议漏洞。

AT对BAN的改进

1. 从语义的角度对某些基本概念进行了修改。

   例如BAN对良好密钥定义需要进一步区分。P相信K是有一个良好会话密钥 ，并不意味P拥有K。
2. BAN逻辑关于主体诚实假设不完善，例如主体转发自己无法知道的真实内容的消息。
3. 引入全部命题连接词。
4. 区分公式与一般表达式。

AT逻辑的缺陷

1. 没有提供基于公钥体制的分析机制。
2. 有些公理存在缺陷。比如密文如果是对临时加密得到的，那么即使主体拥有密钥，也无法确定所收到的消息的内容。