一种认证协议防御拒绝服务攻击的设计方法

卫剑钒, 陈 钟, 段云所, 王立福
(北京大学计算机科学技术系, 北京 100871)

摘 要: 拒绝服务(DoS)攻击是一种阻碍授权用户正常获得服务的主动攻击, 大量认证协议和密钥建立协议存在着不同程度的 DoS 隐患. 本文提出一种新的解决方法, 用于无可信第三方认证协议和密钥建立协议防御 DoS 攻击, 该方法可动态调整 DoS 防御的强度, 并可减少并行会话攻击, 增强协议的安全性.

关键词: 认证协议; 密钥建立协议; 拒绝服务 (DoS); 工作量证明
中图分类号: TP393 文献标识码: A 文章编号: 0372-2112 (2005) 02-0288-06

A New Countermeasure for Protecting Authentication Protocols against Denial of Service Attack

WEI Jian-fan, CHEN Zhong, DUAN Yun-suo, WANG Li-fu
(Department of Computer Science and Technology, Peking University, Beijing 100871, China)

Abstract: Denial of service has become a major security threat in open communications networks. Authentication and key establishment protocols usually are vulnerable to network DoS attacks. This paper presents a new countermeasure to make authentication protocols without trusted third party resistant against DoS attack. By using this method, the strength of resistance can be adjusted dynamically and most parallel session attacks can be prevented.

Key words: authentication protocol; key establishment protocol; denial of service; proof of work

1 引言

拒绝服务攻击(DoS)是一种常见而有效的攻击手段, 它利用协议或系统中的缺陷, 采用欺骗或伪装的手段向服务提供方进行攻击, 试图通过耗尽服务方资源等方法使之无法向授权用户提供服务或造成对时间要求急迫类服务的延迟. 如 SYN-flood DoS 攻击通过大量发送 TCP SYN 连接请求但并不进行相继的交互, 导致服务器保存 TCP 半开连接的内存空间填满而无法接受新的连接请求.

大多数认证协议和密钥建立协议(由于密钥建立协议通常存在认证过程, 以下简称两者为认证协议)中存在着不同程度 DoS 攻击的威胁. 攻击者冒充发起方发起大量的认证请求, 如果认证协议在设计时没有考虑到 DoS 威胁或设计存在缺陷, 响应方就很容易耗尽受限资源而导致 DoS 攻击成功. 资源耗尽通常分为存储资源耗尽、计算资源耗尽以及带宽资源耗尽等类型, 如 SYN-flood 攻击就是一种典型的存储资源耗尽攻击. 使用公钥密码算法的认证协议, 由于用到模指数这样较为昂贵的计算, 则很容易遭受计算资源耗尽攻击.

本文提出了一种新的增强认证协议防御 DoS 能力的通用方法, 在增加两条消息的代价下, 使认证协议能够动态防御计算耗尽和存储耗尽的 DoS 攻击, 并可减少并行会话攻击发生的可能性. 文章在第 2 节给出了认证协议的 DoS 隐患示例以及相关研究, 第 3 节给出了本文研究的假设和目标, 第 4 节给出了 DoS 认证头的概念及其设计框架和设计需求, 并设计了一个满足需求的 DoS 认证头, 第 5 节给出了对无可信第三方认证协议进行 DoS 设计和改进的方法, 并用此方法给出了示例及分析.

2 认证协议 DoS 隐患示例及相关研究

下面以 CCITT X.509 认证协议(三条消息版本)[1] 为例, 来说明认证协议中普遍存在的资源耗尽 DoS 攻击隐患. 协议如图 1 所示, 描述采用传统的 Alice-and-Bob 标记方式, 每条消息(Msg)中, A, B 表示参与协议的主体, K_a, K_b 表示相应主体的公钥, K_a^{-1}, K_b^{-1} 表示相应主体的私钥, $\{X\}_K$ 表示对 X 用密钥 K 加密, 相应地, $\{X\}_{K^{-1}}$ 表示对 X 用私钥 K^{-1} 签名.

Msg1 A → B: A, $\{T_a, N_a, B, X_a, \{Y_a\}_{Kb}\}_{K_a^{-1}}$
Msg2 B → A: B, $\{T_b, N_b, A, N_a, X_b, \{Y_b\}_{Ka}\}_{K_b^{-1}}$
Msg3 A → B: A, $\{N_b\}_{K_a^{-1}}$

图 1 CCITT X.509 认证协议(三条消息版本)

收稿日期: 2004-02-17; 修回日期: 2004-07-14
基金项目: 国家"八六三"高技术研究发展计划基金(No. 2002AA142160)

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

其中, T_a 和 T_b 为时间戳, N_a 和 N_b 为现时(nonce), X_a、Y_a、X_b 和 Y_b 都是用户数据. 协议设计用来向对方认证用户数据的来源及完整性, 并保证数据 Y_a 和 Y_b 的机密性. 本节暂不考虑协议本身存在的安全问题(如文献[2,3]中所指出的), 仅关注其防御 DoS 的能力.

攻击者可以冒充发起方 A 发出大量的虚假消息 Msg1(签名部分使用无意义数据填充), 响应方 B 接收到 Msg1 后, 不做任何判断(也不能做出), 就开始进行较为昂贵的签名验证, 这很容易造成计算资源耗尽; 另外, 由于 X.509 标准规定时间戳的检查在三条消息版本中是可选的, 攻击者可以保存大量窃听的 Msg1, 并在短时间内大量发送, B 对每条消息进行验证, 由于是 A 以前发送的消息, 签名验证会被通过, 然后 B 开始解密 Y_a, 并进行 Msg2 的产生(包含一次公钥加密和一次签名运算)和发送, 同时在存储器中保存这次认证的状态, 如保存 A、N_b、X_a、Y_a、X_b 和 Y_b 等这些状态信息, 这种情况下, 协议同时受到存储耗尽和计算耗尽 DoS 攻击的威胁.

作为对抗 DoS 攻击的方法, Karn 和 Simpson 最早在 Photuris 协议[4]中提出了 Cookie 机制, 并在 IKE 协议[5]中被采用, 其思路是在请求到来时, 服务器(server)端生成一个与客户(client)端相绑定的 Cookie, 然后发送给 client 并要求回送, 由于伪造网络地址(如 IP 地址)的攻击者很难伪造和篡改 Cookie 而继续运行协议, 从而达到防御 DoS 的目的, Cookie 的缺点是不能防御来自真实网络地址发起的攻击. 为防止垃圾邮件, Dwork 和 Naor 在文献[6]中提出了工作量证明(Proof of Work)的思路, 工作量证明要求 client 在发送每个邮件消息前向 server 证明自己已经耗费了一定量的资源代价, 使发送大量垃圾邮件的代价变得昂贵而效率低下. Aura 和 Leiwo 将工作量证明的思路用于认证协议中[7], 但这种方法仍需要在响应方保存额外的状态信息. 另外, Matsuura 和 Imai 在改进 IKE 积极模式防御 DoS 能力的方法中[8]提出了一种可用于防御 DoS 的特定签名算法, 其局限性在于不够通用.

本文提出一种新的方法, 采用工作量证明和会话标识相结合的方法, 能够不保存额外状态信息, 并可动态防御包括使用真实网络地址在内的三类攻击者(定义见第 3 节)发起的 DoS 攻击, 可通用于无可信第三方认证协议的 DoS 设计与改进.

3　假设与目标

本文考虑无可信第三方的认证协议和密钥建立协议, 协议只有发起方和响应方两个参与者且至少由两条消息组成.

假设 1　假设认证协议被 DoS 攻击的目标总为协议的响应方, 并且协议参与双方的网络地址(如 IP 地址)在一次认证协议的运行中(以下简称会话)中不变.

该假设在通常情况下总是满足的, 一个 DoS 攻击者总是主动发起协议运行, 服务器通常也总是网络协议的响应方; 绝大多数网络协议也总是要求会话中网络地址(如 IP 地址和端口号)不变.

假设 2　攻击者可以窃听、拦截、篡改、伪造和重放系统中传送的任何消息, 可以进行网络地址伪装.

该假设赋予了 DoS 攻击者在网络环境下的能力, 与目前绝大多数认证协议安全分析所刻画的攻击者能力相一致.

假设 3　假设 DoS 攻击者不具备任意延迟或篡改所有流入流出服务器端消息的能力; 服务器和网络有足够的带宽而不至于被带宽资源耗尽攻击; 有足够的物理安全措施不至于被物理破坏而导致无法向客户端提供服务.

如果攻击者具备在服务器节点处任意延迟或篡改消息的能力, 只需要简单地抛弃或破坏每个发往服务器的消息即可达到 DoS 目的, 其性质接近于物理破坏. 由本假设, 本文考虑的 DoS 攻击类型为存储资源耗尽攻击和计算资源耗尽攻击, 带宽耗尽, 物理破坏等其他方式导致的 DoS 不予考虑.

定义 DoS 防御目标之前, 先依据攻击者是否伪造网络地址和其能力的差异, 将其划分为如下三类: 第一类攻击者: 使用伪造网络地址发送消息, 但不能收到发往这个地址的消息. 这是最常见的 DoS 攻击者类型.

第二类攻击者: 使用伪造网络地址发送消息, 并可通过改变路由或攻破同一子网内主机并监听等方法接收服务器发往此伪造网络地址的消息.

第三类攻击者: 使用真实网络地址发送和接收消息. 对常规的攻击者来说, 为防止取证, 通常不使用这种方式, 但可能使用在被攻破的傀儡主机上, 这在分布式拒绝服务(DDoS)攻击中较为普遍.

综上, 本文定义如下的认证协议 DoS 防御目标:

目标 1　能够防御来自第一类攻击者发起的 DoS 攻击.

目标 2　能够防御第二类、第三类攻击者的 DoS 攻击, 并且防御强度可视攻击强度进行动态的调整.

目标 3　防御 DoS 机制所带来的服务器端资源消耗足够小, 以防止该机制自身被 DoS 攻击.

目标 4　增加了 DoS 防御机制后的认证协议, 其安全性不能低于原有协议的安全性.

4　DoS 认证头

本节讨论 DoS 认证头的概念和设计方法, 在第 5 节给出基于 DoS 认证头的认证协议 DoS 设计和改进方法.

4.1　DoS 认证头的基本概念

我们给出 DoS 认证头非形式化的定义如下:

定义 1　DoS 认证头是位于认证协议起始部分的一个针对 DoS 攻击的弱认证(相对于认证协议主体部分的强认证), 并满足以下两个条件:

(a)只有在确认发起方通过了该弱认证, 响应方才开始进行其后的强认证部分.

(b)为通过该弱认证, 发起方需耗费远大于响应方的资源代价.

由于每个参与者可能同时发起多个协议运行会话, 双方必须维护一个会话标识(session identifier). 然而在大多数的认证协议中, 都没有显式地在协议中标明会话标识, 而是交由实现来处理. 我们认为会话标识对认证协议的安全性有相当重要的作用, 应该被显式地标识在协议的消息中, 一方面会话标识用来在多个并发会话中区分特定会话; 另一方面, 可以减少

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

并行会话攻击发生的可能性.

会话标识必须保证唯一性,由发起方或响应方单方选取会话标识是不可行的,这是由于一方无法验证另一方选取的会话标识是否为一个重放.我们采用参与双方各产生会话标识一部分的方法来生成会话标识,每一方通过确保自己产生部分的新鲜性来保证整个会话标识的新鲜性.为描述方便,以下称发起方生成的部分为 SII,响应方生成的部分为 SIR.

工作量证明的思路是让客户端首先消耗资源,最早由 Dwork 和 Naor 在解决垃圾邮件问题时提出[6],他们建议在发送邮件时,发送者必须在发送每个消息前解决一个小的密码问题(puzzle),对于普通用户来说,这个代价可以接受,但对于发送大量垃圾邮件来说代价高昂而难以实施.文献[7]使用了同样的思路解决认证协议的 DoS 问题,但是需要在每个间隔时间 t 内,server 端保存每个通过验证的 puzzle、puzzle 解答以及用户标识,以防止攻击者通过重放攻击绕开 puzzle 机制.

我们采用会话标识和 puzzle 相结合的方式设计 DoS 认证头,如果设计满足下节定义的 DoS 认证头设计需求,则能够满足第 3 节中定义的防御 DoS 攻击的前 3 个目标,并且不需要额外保存每个 puzzle 及其相关信息,减少了 server 的资源负荷.在第 5 节中,将对目标 4 进行讨论.

412 DoS 认证头的设计框架与设计需求

DoS 认证头设计框架由三条消息组成.为方便分析协议双方的行为及运行代价,描述采用文献[9]中带注解的 Alice2and2Bob 协议描述规范,描述形如 A y B: T_1, , , T_k + M + O_1, , , O_n,其中 A 和 B 为协议运行方,M 为 A 发送给 B 的消息,T_1, , , T_k 是 A 在产生 M 时所作的一系列操作,O_1, , , O_n 表示 B 在处理 M 时采取的一系列操作.

DoS 认证头的设计框架如下:

(1) 协议发起方产生(gen)一个随机数作为发起方会话标识 SII,存储(store)SII,并作为第一条消息发给响应方.

Msg1: A y B gen SII, store SII + SII +

(2) 响应方收到 Msg1 后,产生 SIR.并根据当前的受攻击程度和剩余资源状况,产生一个 puzzle,发送给发起方.

Msg2: B y A gen SIR, gen puzzle + SIR, puzzle + store SIR

发起方收到 Msg2 后,存储 SIR,和 SII 共同形成会话标识(SII, SIR).注意响应方在此时尚未创建任何状态,因为并未保存 SIR 和 puzzle 或其他与发起方相关的信息.

(3) 发起方求解(solve) puzzle,发送会话标识及 puzzle 解答(solution).

Msg3: A y B retrieve (SII, SIR), solve puzzle + SII, SIR, solution + check SIR, check solution, store (SII, SIR)

响应方首先在无状态情况下验证 SIR 是否正确,如果正确,再验证 puzzle 解答是否正确,任何一项验证不通过时,协议停止运行.都通过时,开始创建状态,保存会话标识(SII, SIR),继续随后的认证协议并开始提供资源.

DoS 认证头的生成和处理应确保满足如下需求:

需求 1 在验证完 puzzle 的解答是否正确之前,响应方不创建任何和这个特定会话相关的状态.

如果每个会话都需要保存和这个会话相关的状态,即便是很小的存储量,也易遭受大量请求包带来的存储资源耗尽攻击,因为攻击者发送 Msg1 的代价非常低廉.Simpson 经过测试指出[10],即便是采用了废状态回收,IKE 所采用的用于防御 DoS 攻击的有状态 cookie 机制仍然很容易遭受到 DoS 攻击.

需求 2 为防止网络地址欺骗,SIR 的产生必须和发起方的网络地址相关,并且除响应方外,不存在任何其他实体能产生被响应方验证通过的 SIR 值.

需求 3 SII、SIR 以及 puzzle 解答必须有足够长度以保证会话标识重复的概率能够小到可以忽略不计,并能够防止被攻击者创建数据库进行攻击.

需求 4 发起方不能预先求解 puzzle,也无法根据一个或多个其他 puzzle 的求解降低某次 puzzle 求解的代价.

需求 5 发起方求解 puzzle 的代价可由响应方较为容易的调整,使求解代价可以从零到几乎不可能求解.求解代价为零也即不使用 puzzle 机制.

需求 6 SIR 和 puzzle 必须能够较快并且代价较低地产生和验证.

需求 7 响应方要有能防止攻击者重放 SIR 和 solution 的机制,如使用时间戳或通过在 SIR 和 puzzle 的产生和验证中加入时变的本地秘密来限制其使用时间.

一个使用上述框架的 DoS 认证头,如果满足需求 1,则为一个无状态的认证机制,在这个弱认证完成前,不会被攻击者通过发送大量消息而耗费存储资源,避免类似 cookie crump[10] 的攻击;如果满足需求 2,则第一类攻击者不能收到发往伪造地址的 SIR,也无法伪造一个可通过验证的 SIR,从而无法继续进行协议,达到目标 1;对于第二类和第三类攻击,攻击者可以收到 SIR,但必须求解 puzzle,这使得攻击代价增高,不同的 puzzle 求解难度会使响应方的负荷得到不同程度的减轻,从而有效地防御 DoS 攻击,如果 puzzle 的设计满足需求 4,并且求解代价可以较为容易地调整即满足需求 5,则可达到目标 2;需求 6 如果被满足,DoS 认证头的运行对响应方来说,代价低廉并且速度较快,又因为满足需求 1,整个过程是无状态的,所以 DoS 认证头本身可以防御 DoS 攻击,达成目标 3;最后,需求 3 和需求 7 的满足可以防止攻击者通过重放或构建数据库来绕过这个认证机制.所以,满足上述需求的 DoS 认证头可以达到第 3 节中提出的目标 1 到目标 3,目标 4 的达成将在第 5 节中给出.

413 DoS 认证头示例及实验

图 2 示例了一个满足上节各需求的 DoS 认证头,记之为 PHC(partial hash collision)2DoS 认证头,其中: $s = \{0,1\}^k$ 表示长度为 k 的比特串,0^k 表示长度为 k 的全零比特串,$_R\{0,1\}^k$ 表示长度为 k 的随机数.

s[i]表示比特串中的第 i 个比特,s[1]表示比特串中最左边的位.

符号/ = (left b)0 表示两旁的两个比特串最左边 b 个比特相等,即:

x = (left b)y 表示 $P_{i=1,,,b}(x[i] = y[i])$.

Hash 表示一个单向散列运算(如 MD5),HMAC 表示相应散列算法的 HMAC 运算,| 表示连接符.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

SII= $_R\{0,1\}^{64}$, 长度为 64 比特
SIR= HMAC(secret, SII| IP$_i$), 长度为 64 比特
Secret: 响应方本地秘密, 定期更换(如 30 秒)
IP$_i$: 由 Msg1 中获取的发起方的 IP 地址
puzzle= k, 长度 8 比特, 取值从 0 到 64, 根据策略定期改动
puzzle 要求发起寻找 solution 满足:
Hash(SII| SIR| solution)= (left k) 0^{64} solution 长度为 64 比特
Msg1　A y B: gen SII, store SII+ SII+
Msg2　B y A: gen SIR, gen puzzle+ SIR, k+ store SIR
Msg3　A y B: retrieve(SII, SIR), solve puzzle+ SII, SIR, solution+
check SIR, check solution, store(SII, SIR)

图 2　PHC2DoS 认证头

在 PHC2DoS 认证头中, 发起方对每个新会话产生一个长度为 64 比特的随机数作为 SII 发给响应方. 响应方收到 SII 后, 用一个定期改变的本地秘密 secret 和 SII、IPi 做 HAMC 产生 SIR, 并根据受攻击程度和当前资源情况由安全策略决定 k 值, 和 SIR 一同在 Msg2 中发送给发起方. 发起方纪录 SII 和 SIR 作为会话标识, 求解 puzzle 使得 Hash(SII| SIR| solution) 的左边 k 比特为全 0, 这是一个求解部分散列碰撞(partial hash collision)的问题, 发起方只能通过暴力(brute force)破解的方法来寻求 solution, 显然, k 越大, 发起方求解的代价越大, k= 0 则表明不需要求解 puzzle. 求解完成后, 发起方送交 SII、SIR 和 solution 作为认证头的最后一条消息. 响应方收到 Msg3 后, 根据本地存储的 secret 和 k 对 SIR 和 puzzle 进行验证, 如果通过, 则开始提供资源, 记录会话标识(SII, SIR), 并继续进行认证协议.

下面对认证头是否满足各需求进行简要的分析, 容易看出, 响应方对每个会话唯一创建状态的动作 store(SII, SIR) 是在完成 SIR 和 puzzle 的验证之后进行的, 所以满足需求 1; SII、SIR 和 solution 的长度均为 64 位, 满足需求 3 中对会话标识的要求; 由 SIR 中本地秘密 secret 成分以及对 IP 的绑定, 满足需求 2, Hash 函数的单向性使得需求 4 被满足, 通过改变 k 的大小, 可以容易的调整发起方求解 puzzle 的代价也即满足需求 5, Hash 和 HAMC 都是运算较快而且代价较小的运算, 所以需求 6 被满足.

对于重放攻击(需求 7), 一个攻击者可能试图通过窃听有效 SII、SIR 和 solution 并重放 Msg3 来绕过 puzzle 求解, 由于 secret 每过一个时间段就进行一次改变, 攻击者必须在这个时间段内进行使用, 这防止了攻击者收集大量有效 puzzle 求解并集中发送进行 DoS 攻击的可能; 另一方面, 在实现中, 一个在时间段内重放的 Msg3, 若其中的会话标识(SII, SIR) 对应的原会话仍未结束, 会被立即检测出是重放而抛弃. 对于已经结束的会话, 则可通过限制时间段内每个 IP 的会话数来防止重放, 具体数目由系统的安全策略决定, 如在某系统中可规定一分钟内同一 IP 地址的用户不允许发起 3 次以上的认证.

我们对 PHC2DoS 认证头进行了实验, 使用 SHA21 作为 Hash 函数, 在 P32933MH2windows2000 系统环境下, 测得计算 Hash(SII| SIR| solution) 的平均速度为 452500 次/秒. 理想情况下暴力破解 puzzle 需要探测的次数为 2^k. 在 k 为 17, 18, 19, 20, 21 时, 测得 100 次求解 puzzle 平均的/Hash 次数/求解时间0 分别为 137816/0. 3118s, 272368/0. 6168s, 548107/1. 2427s, 984221/2. 2368s, 2000567/4. 4883s, 验证了 PHC2DoS 认证头可以通过调节 k 容易地改变 puzzle 求解时间. 正常网络环境下不需启动 puzzle 机制, 也即将 k 置为 0. 当检测到 DoS 攻击后, 随攻击强度增大而逐渐增大 k, 如采用 k= 18 时, 意味着发起方平均需要约 2^{18}次测试约才能求解 puzzle 并继续运行协议, 在上述测试环境中平均需要 0. 6 秒左右. 对于每秒可以发送 10 万个消息的攻击者, 攻击强度比不使用 puzzle 时下降了 6 万倍左右. 对于使用傀儡机的 DDoS 攻击或者征集良性受害者求解 puzzle 的攻击方法[11], 则意味着为达到同等效果, 需要征集参与攻击的主机数目为以前数目的上万倍之多, 而这几乎是不可行的.

5　基于 DoS 认证头的认证协议防御 DoS 攻击的方法

511　认证协议 DoS 防御设计与改进步骤

给定一个存在 DoS 攻击威胁的无可信第三方认证协议 AP(即 AP 不含 DoS 防御设计或 DoS 防御设计有缺陷), 我们提出如下的方法, 在满足第 3 节给出的假设的前提下, 修改 AP 使之具备防御 DoS 攻击的能力, 达到本文提出的防御目标. 其中, Msg(P, n) 表示协议 P 的第 n 条消息, len(P) 表示协议 P 中消息的个数, T(P) 表示协议 P 中消息的原子成分(包括文本、时戳和密钥等)集合, | 表示消息连接符号.

步骤 1　设计 DoS 认证头 DoS2Hdr. 要求 DoS2Hdr 满足第 4 节 DoS 认证头的设计框架和设计需求, 并且 DoS2Hdr 与 AP 协议中原子成分相独立, 即 T(DoS2Hdr) H T(AP)= ^{a}.

步骤 2　按下面的方法将 DoS2Hdr 与 AP 合并得协议 AP0.

Msg(AP0, i)= Msg(DoS2Hdr, i)　i= 1, 2;　Msg(AP0, 3)= Msg(DoS2Hdr, 3)| Msg(AP, 1);

Msg(AP0, i)= Msg(AP, i- 2)　3< i[len(AP)+ 2; 合并后, len(AP0)= len(AP)+ 2.

步骤 3　分析并处理 AP0 中出现的冗余部分.

有的认证协议已经有防御 DoS 的机制, 但由于使用不当, 仍然会有 DoS 隐患, 加入 DoS 认证头后, 应当移除或根据第 4 节中的设计需求修改原先的 DoS 防御机制.

步骤 4　在 DoS 认证头后的每条消息首部增加会话标识成分, 并进行完整性保护, 得到改进后的认证协议 AP1.

Msg(AP1, i)= Msg(AP0, i)　1< i< 3;　Msg(AP1, i)= SII| SIR| Msg(AP0, i)　3< i[len(AP0).

其中 Msg 为添加会话标识完整性认证后的消息, 大多数认证协议通过签名或伪随机函数运算(或 HMAC) 的方式认证消息中内容的完整性, 此时将会话标识加入签名或伪随机函数运算的内容中即可; 如果原消息没有完整性认证部分, 则添加对会话标识的完整性认证.

此时可得到改进后的认证协议 AP1. 由于在步骤 1 中保证了 DoS 认证头与 AP 中的成分相互独立, 所以组合后协议 AP1 不会降低 AP 的安全性, 满足目标 4 的要求. 综合 4. 2 节

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

中对 DoS 认证头的抵御 DoS 能力分析, AP1 在满足第 3 节假设的前提下, 达到本文提出的认证协议 DoS 防御目标. 改进后协议 AP1 比 AP 多两条消息, 认证头后每条消息增加了会话标识以及对其的认证, 协议效率略有下降. 另外, 从第 5.2 节的分析中可看出, 进行了会话标识的完整性认证后, 能够在一定程度上防止并行会话攻击的发生, 增强了协议 AP 的安全性.

5.2 协议改进示例及分析

按照 5.1 节所述的步骤, 使用 4.3 节所示的 PHC2DoS 认证头, 我们对 CCITT X.509 协议进行改进, 得改进后协议如图 3 所示:

Msg1 A y B SII

Msg2 B y A SIR, k

Msg3 A y B SII, SIR, solution, A, $\{SII, SIR, T_a, N_a, B, X_a, \{Y_a\}K_b\}\overline{K_a}^{1}$

Msg4 B y A SII, SIR, B, $\{SII, SIR, T_b, N_b, A, N_a, X_b, \{Y_b\}K_a\}\overline{K_b}^{1}$

Msg5 A y B SII, SIR, A, $\{SII, SIR, N_b\}\overline{K_a}^{1}$

图 3 使用 PHC2DoS 认证头进行 DoS 防御改进后的 CCITT X.509 认证协议

图 3 中的 Msg1、Msg2 和 Msg3 中的前三项(SII, SIR, solution)即为 DoS 认证头, 如果要防止攻击者篡改 k, 可将 k 改成 $\{k, T_b\}\overline{K_b}^{1}$, 即将 k 与时间戳 T_b 相绑定并签名, 签名以防止伪造, 绑定时间戳以防止重放. 但这不是必需的, 因为根据假设 3, 攻击者不具备修改服务器所有发出消息的能力.

由第 2 节分析可知原 X.509 协议不能防御存储资源耗尽和计算资源耗尽 DoS 攻击. 改进后, 响应者收到 Msg3 后, 首先按照 PHC2DoS 认证头中的方法验证 SIR 和 solution 是否通过, 如果不通过, 由于 Dos 认证头是无状态的, 响应者在验证不通过前并不建立任何与会话相关的状态, 故能够防御存储耗尽攻击; 又根据 4.3 节的分析, 此前的计算都是代价较小的运算, 也构不成计算资源耗尽攻击.

并行会话攻击是指攻击者在两个或多个协议运行(即会话)并发的环境中, 利用某个会话产生的消息来生成另一个会话的消息, 从而达到攻击的目的. 由于在 CCITT X.509 协议标准中规定时间戳的检查可选, 导致其存在严重的安全缺陷[2], 使得协议存在如图 4 所示的并行会话攻击(此外, CCITT X.509 三条消息版本还存在一个安全缺陷[3], 即加密后签名的问题, 由于超出本文讨论的范畴, 这里不做讨论).

Msg1 C(A) y B: a, $\{T_a, N_a, B, X_a, \{Y_a\}_{Kb}\}\overline{K_a}^{1}$

Msg2 B y C(A): B, $\{T_b, N_b, A, N_a, X_b, \{Y_b\}_{KA}\}\overline{K_b}^{1}$

Msg1c A y C: A, $\{Tc_a, Nc_a, C, Xc_a, \{Yc_a\}_{Kc}\}\overline{K_a}^{1}$

Msg2c C y C: C, $\{Tc_c, Nc_b, A, Nc_a, Xc_c, \{Yc_c\}_{KA}\}\overline{K_c}^{1}$

Msg3c A y C: A, $\{N_b\}\overline{K_a}^{1}$

Msg3 C(A) y B: A, $\{N_b\}\overline{K_a}^{1}$

图 4 CCITT X.509 协议中存在的并行会话攻击

攻击由两个会话组成, 在图 4 中用. 号标识第 2 个会话, Msg1 是 C 冒充 A(用 C(A) 表示)重放一条 A 以前发送过的消息, 由于并不检查时间戳 T_a, B 返回 Msg2, 这时 C 得到 N_b, 并通过某种方式触发 A 开始对 C 的会话, C 用得到的 N_b 返回 Msg2., 得到 A 的回应 $\{N_b\}\overline{K_a}^{1}$, 然后 C 将之转发给 B 完成冒充 A 的会话. 使 B 相信 Msg1 是 A 最近发出的消息, 从而认可过时的数据 X_a 和 Y_a, 达到攻击目的.

并行会话攻击是对认证协议的一种较为常见的攻击方式, 产生原因是由于协议的认证部分缺乏必要的绑定. 对存在并行会话攻击的不同协议, 人们提出了不同的解决方案(对 X.509 这个缺陷, 文献[2]提出的解决方案是在 Msg3 的签名内容中添加 B 的标识). 我们认为绑定会话标识可以作为协议设计和改进的一个原则, 因为会话标识由每个参加协议的主体共同产生, 比起单方的检查时间戳或者 Nonce, 检查会话标识更有利于一个协议主体判断某条消息是否来自本会话, 从而大大减少并行会话攻击的可能性.

图 3 所示的改进后的 X.509 协议由于绑定了会话标识, 能够防御图 4 所示的并行会话攻击, 分析略.

6 结语

认证协议正如其他网络协议一样, 存在着 DoS 安全隐患, 如果设计时缺乏对 DoS 防御的考虑, 很容易造成存储资源或计算资源的 DoS 耗尽攻击, 导致无法正常地向合法用户提供认证服务. 随着网络 DoS 攻击数量和规模的增加, 认证协议中的 DoS 问题正成为认证协议研究的关注方向之一[12].

本文提出一种通用的方法, 用于增强无可信第三方认证协议和密钥建立协议防御 DoS 攻击的能力, 给出了认证协议防御 DoS 攻击的假设与目标, 采用会话标识与工作量证明相结合的方法解决 DoS 问题, 在增加两条消息的代价下, 使认证协议能够同时防御计算资源耗尽和存储资源耗尽的 DoS 攻击. 同时减少了并行会话攻击发生的可能性, 增强了认证协议的安全性能. 如何减少假设, 并提高改进后认证协议的效率, 都是需要进一步研究的问题.

参考文献:

[1] CCITT Recommendation X.509. The Directory2Authentication Framework. CCITT[S]. 1988.

[2] Michael Burrows, Martin Abadi, Roger Needham. A logic of authentication[J]. ACM Transactions on Computer Systems, 1990, 8(1): 18-36.

[3] Colin Anson, Chris Mitchell. Security Defects in the CCITT Recommendation X.509 : The Directory Authentication Framework[J]. Computer Communication Review, 1990, 20(2): 30- 34.

[4] Phil Karn, W A Simpson. Photuris: session2key management protocol. RFC 2522[S]. IETF Network Working Group, 1999.

[5] Dan Harkins, Dave Carrel. The Internet key exchange (IKE). RFC 2409[S]. IETF Network Working Group, 1998.

[6] C Dwork, M Naor. Pricing via processing or combatting junk mail[A]. Proc. CRYPTO. 92[C]. Berlin: Springer, 1992. 139- 147.

[7] Tuomas Aura, Pekka Nikander, Jussipekka Leiwo. DOS2resistant authentication with client puzzles[A]. Proc. of Security Protocols Workshop[C]. Berlin: Springer, 2000. 170- 177.

[8] K Matsuura, H Imai. Modification of internet key exchange resistant a2

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

gainst denial2of2Service [A]. Pre2Proc. of Internet Workshop 2000 (IWS2000) [C]. NSW, Australia, 2000. 167- 174.

[9] C Meadows. A formal framework and evaluation method for network de2 nial of service[A]. Proc. of the 12th IEEE. Computer Security Founda2 tions Workshop[C]. Mordano, Italy, 1999. 4- 13.

[10] W A Simpson. IKE/ ISAKMP Considered Dangerous[DB/OL]. Internet Draft, draft2simpson2danger2 isakmp2 01. txt, 1999.

[11] Geraint Price. A general attack model on hash2based client puzzles [A]. Cryptography and Coding, 9th IMA International Conference[C]. Berlin: Springer, 2003. 319- 331.

[12] 卿斯汉. 安全协议 20 年研究进展[J]. 软件学报, 2003, 14(10): 1740- 1752.

作者简介:

卫剑钒　男, 1974 年 2 月生于山西运城, 分别于 1995 年、1998 年获得空军工程大学工程学院工学学士和硕士学位, 现为北京大学计算机科学技术系博士研究生, 主要研究方向为网络与信息安全, 安全协议形式化分析与设计. E2 mail: weijf@infosec. pku. edu. cn.

段云所　男, 1967 年 5 月生于云南大理, 北京大学计算机科学技术系副教授, 主要研究领域为网络信息安全, 电子商务安全.

陈　钟　男, 1963 年 4 月生于江苏徐州, 北京大学计算机科学技术系教授, 博士生导师, 主要研究领域为网络与信息安全, 嵌入式系统, 软件工程.

王立福　男, 1945 年 4 月生于辽宁瓦房店, 北京大学计算机科学技术系教授, 博士生导师, 主要研究领域为软件方法学, 软件质量保证, 网络与信息安全.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net