

文章编号:1001-9081(2009)06-1659-03

基于混合体制的 Kerberos 身份认证协议的研究

胡 宇¹, 王世伦²

(1. 四川师范大学 计算机科学学院, 成都 610068; 2. 四川城市职业学院, 成都 610101)

(huyu0833@163.com)

摘 要:对 Kerberos 身份认证协议方案进行了详细的分析,针对 Kerberos 协议本身存在的局限性,从系统安全性和实际执行性能角度出发,提出了一种混合加密体制的 Kerberos 改进协议。并且解决了 Kerberos 认证协议可能存在窃听通信双方会话的问题,从而防止内部攻击。

关键词:Kerberos 认证协议;安全性分析;混合加密体制;两方保密通信

中图分类号:TP393.09 **文献标志码:**A

Research on Kerberos identity authentication protocol based on hybrid system

HU Yu¹, WANG Shi-lun²

(1. College of Computer Science, Sichuan Normal University, Chengdu Sichuan 610068, China;

2. Urban Vocational College of Sichuan, Chengdu Sichuan 610101, China)

Abstract: Kerberos identity authentication protocol was analyzed in detail. According to its limitations, a new protocol about Kerberos based on hybrid system was proposed to improve the security and actual implementation performance of the system. Moreover, the problem that Kerberos authentication protocol maybe eavesdrop two-side communication conversations was solved in the system, thus inner attacks could be avoided.

Key words: Kerberos authentication protocol; safety analysis; hybrid encryption system; two-side privacy communication

0 引言

Kerberos 协议是以可信赖第三方为基础的认证协议,为网络中的所有实体提供一个集中的、统一的身份认证管理机制,广泛应用于 Internet 服务的访问。Kerberos 协议所采用的是对称加密体制有域内认证和域间认证两种模式,可提供安全的客体认证。本文详细分析了 Kerberos 认证协议的原理以及针对由对称算法所带来的局限性和可能存在的内部窃听安全隐患,利用混合加密体制和安全的 Diffie-Hellman 密钥协商协议对 Kerberos 协议进行一定的改进,进而更具有安全性和实用性。

1 Kerberos 认证协议

所谓身份认证就是指验证通信对象是原定的那位而不是冒名顶替者的技术。Kerberos 的安全不依赖于用户登录的主机或者应用服务器,而是依赖于几个认证服务器——Kerberos 服务器(也称之为 KDC 密钥分发中心)。它是整个认证系统的核心,其中维护了所有用户的账户信息,包括认证服务器(Authentication Server, AS)和票据授予服务(Ticket-Granting Service, TGS),主要是提供票据和会话密钥。AS 的作用是对用户的身份进行初始认证,若认证通过便发放给用户一个称为 TGT 的票据,凭借该票据用户可访问 TGS,从而获得访问应用服务器时所需的服务票据,票据(Ticket)是指能够证明客户端身份的凭证。Kerberos 把身份认证的任务集中在身份认证服务器(AS)上执行。Kerberos 认证过程中使用两种数据结构包含具有信任信息的信任状,分别是票据(Ticket)和认证单(Authenticator)。Kerberos 认证协议的基本结构如图 1

所示。

Kerberos 认证过程分三个阶段六个步骤,用公式符号化来表示 Kerberos 认证协议的工作流程。

第一阶段 客户端 C 请求认证服务器 AS 发给访问票据授予服务器 TGS 的门票 TGT。

1) C → AS: IDc, EKc(IDtgs, ADc, T1)

2) AS → C: IDc, EKc(Kc.tgs, TGT)

Kc 表示客户端用户的长期密钥(即用户口令的散列值);T1 表示时间戳;Kc.tgs 表示客户与 TGS 间通信的会话密钥;票据 TGT = EKtgs(Kc.tgs, IDc, ADc, Times, T2);Times 表示门票的有效期限时间信息,包含起始时间、结束时间、更新时间;T2 表示时间戳;Ktgs 为票据授予服务器的密钥。

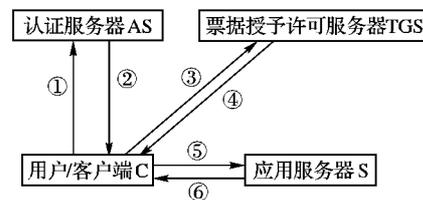


图 1 Kerberos 认证协议的结构

第二阶段 客户端 C 访问 TGS,获得访问应用服务器 S 的门票 ST。

3) C → TGS: TGT, Authenticator_c

4) TGS → C: IDc, EKc.tgs(Kc.s, ST)

Authenticator_c 表示用户提交的认证单,具体为 EKc.tgs(IDc, ADc, IDc, T3);Kc.s 表示客户与应用服务器间通信的会话密钥。票据 ST = EKs(Kc.s, IDc, IDc, ADc, Times, T4)。

第三阶段 客户端 C 与应用服务器 S 间相互验证身份

收稿日期:2008-12-31;修回日期:2009-03-06。

作者简介:胡宇(1984-),男,四川乐山人,硕士研究生,主要研究方向:网络、信息安全、数据库;王世伦(1969-),男,重庆人,教授,主要研究方向:计算机网络、数据库。

(挑战—应答方式)。

5) $C \rightarrow S: ST, Authenticator_{cc}$

6) $S \rightarrow C: IDc, EKc.s(T5 + 1)$

$Authenticator_{cc}$ 表示用户提交的认证单, 具体为 $EKc.s(IDc, ADc, T5)$ 。用户和服务器相互验证身份以后就利用会话密钥 $Kc.s$ 进行保密通信了。

Kerberos 的局限性:

1) 时钟同步较难。在整个认证过程中都要通过对时间戳的比较, 才能判定合法身份。这就要求在整个网络内的时钟实现准同步。但是由于变化的和不可预见的网络延迟, 不能期望分布式时钟保持精确的同步。同时, 时间戳也带来了重放攻击的隐患, 例如 Kerberos 系统中客户使用的认证单在有效期内可能被重放, 从而使资源服务器为非授权用户提供服务。

2) 口令猜测攻击。从认证过程来看, AS 并不能验证用户的身份。消息是用长期密钥 Kc 加密的, 而 Kc 由用户键入的口令导出, 这是 Kerberos 最薄弱的环节, 易被窃听和猜测攻击。还有 Kerberos 返回给用户的票据 TGT 也是用用户密钥加密的, 在 KDC 中必须存储用户的密钥。因此 TGT 的安全性依赖于 KDC 的安全性。如果一个攻击者获得了访问 KDC 用户密钥数据库的权限, 即使仅有只读的权限, 就可以获取 KDC 颁发的 TGT 中的会话密钥, 解密用户数据。

3) 密钥的存储和管理复杂。由于 Kerberos 协议采用的是对称加密机制, 必然会导致密钥存储和管理困难。在 KDC 中存储了与应用服务器、用户之间大量的共享密钥。对于大规模的应用来说通过一个 KDC 来完成, 可能会出现“瓶颈”问题。同时, 采用对称加密体制不支持数字签名和不可否认性的服务。

4) 恶意软件攻击问题。Kerberos 服务器保存了所有用户和服务器的密钥。一旦被黑客攻击成功, 那所有的密钥将会泄露。

基于上述 Kerberos 存在的问题, 在许多改进方案中都引入了 Kerberos RSA 验证协议为代表的公钥加密体制, 这样的确解决了上述存在的局限性。但是, 就 RSA 本身而言, 加解密速度较慢, 不适合大数据量传输。若整个流程都使用公钥加密体制, 必然会导致执行性能方面的效率的降低, 而且在实际应用中一般不直接使用公钥体制来加密传输的信息。所以应该合理的利用公钥加密技术。

2 基于混合体制的 Kerberos 认证协议

考虑上述存在的因素, 权衡安全性和执行方面的效率。在实际应用中, 尤其当需要加密大量的数据时, 通常采用是目前一种标准的方法——混合体制, 就是结合公钥加密体制和对称加密体制, 利用各自的优点(公钥系统易于密钥分配和对称密码系统的高效率)来实现新的加密方案: 用对称密钥来加密通信消息; 而使用公钥来加密一个用于对称密码加密的密钥, 该密钥即为发送方和接收方之间通信的会话密钥。在这个会话密钥的控制下, 采用对称加密体制对大量数据信息进行加密。这样即保证了安全性又在效率方面得到了缓解。所以, 仅在关键环节使用公钥的鉴别技术, 本方案仅在初始化和获取许可票据时采用公钥技术。

改进后的 Kerberos 认证协议的工作流程:

1) $C \rightarrow AS: IDc, IDtgs, Cert_c, EKRC(R1)$

2) $AS \rightarrow C: EKUC(Kc.tgs, IDtgs, R1', EKRC(IDc, TGT)), Cert_{as}$

其中: KUc, KRc 为客户端 C 公钥和私钥; $R1$ 为随机数用来替代原协议中的时间戳, 并用 C 的私钥签名。利用公钥技术生成票据 TGT, 具体表示为 $EKUtgs[EKRAs(IDc, ADc, Times, Kc.tgs)]$, 先用 AS 的私钥加密(签名), 再用 TGS 的公钥加密, 使得只有 TGS 才能打开。采用基于公钥证书的身份认证方案, 而不使用用户口令来产生密钥, 防止口令猜测攻击。

3) $C \rightarrow TGS: TGT, Authenticator_c$

4) $TGS \rightarrow C: IDc, EKc.tgs(Kc.s, IDtgs, EKRTgs(IDc, ST), R2')$

其中 $Authenticator_c$ 表示 C 的认证单, 具体为 $EKc.tgs[EKRc(IDc, IDs, R2)]$, C 用私钥对其签名, 再用与 TGS 间的共享密钥加密; 票据 ST 同样利用公钥技术来生成, 具体表示为 $EKUs[EKRtgs(IDc, ADc, IDs, Times, Kc.s)]$, 用 TGS 的私钥加密(签名), 再用 S 的公钥加密。

5) $C \rightarrow S: ST, Authenticator_{cc}$

6) $S \rightarrow C: EKc.s(R3')$

$Authenticator_{cc}$ 用户 C 身份认证单具体表示为 $EKc.s[EKRc(IDc, IDs, R3)]$ 。

改进后的新方案中关键是引入了数字签名验证的机制, 对票据和认证单都进行签名验证, 从而防止被篡改和伪造, 提供了不可否认性的服务。

按照上述流程, 到目前为止可能认为 C 与 S 利用 $Kc.s$ 可以进行保密通信了。但是事实上, 并没有达到真正的两方面的保密通信。因为在 C 和 S 间进行密文传输的过程中, Kerberos 可以截获到此密文, 并用 $Kc.s$ 解密此密文, 轻松阅读 C 和 S 的会话, 原因在于 Kerberos 也共享了密钥 $Kc.s$, 即会话密钥 $Kc.s$ 为客户端 C、TGS 和应用服务器 S 三者共享。尽管 Kerberos 认证系统被认为是可信赖的第三方认证协议, 但是从信息保密的角度考虑, C 和 S 间的通信内容仅希望是彼此双方知道。例如: 客户端 C 预发送消息 M 给 S, 如图 2 所示。

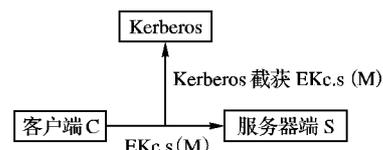


图 2 Kerberos 窃听通信双方

在 Kerberos 认证系统中, 由于客户 C 和服务器 S 之间的会话密钥是由 Kerberos 产生的, 因此该协议也就存在一个漏洞: Kerberos 可以窃听客户和服务端之间的会话而不会被举证, 因为 Kerberos 完全可以抵赖他所做的一切。所以需要产生一个仅由 C 和 S 专门共享的会话密钥来实现两方的安全保密通信。为此, 可以使用 Diffie-Hellman 密钥交换协议, 但是由于该协议没有提供对协议消息源的认证服务, 所以存在中间人攻击是可能的。为解决该问题引入了基于混合加密的 Diffie-Hellman 密钥交换协议来改进此漏洞, 发送方使用接收方经 CA 认证后的公钥来加密用于通信的对称会话密钥。该方法可以避免中间人攻击, 但是却存在这样的两个缺陷问题:

1) 会话密钥的生成。一般地, 会话密钥是由一方(消息的发送方)生成的, 另一方(消息的接收方)不得不完全依赖于发送者为安全通信而生成密钥的能力(或诚实), 这样实际上对另一方并不公平。而且, 这在某些环境下可能不是理想的, 例如, 在 SSL 协议的客户端—服务器环境下, 客户端(即发

送者)单方面生成用于通信的随机会话密钥,其安全性不高。这样的缺陷不满足密钥交换中的密钥控制安全性原则。

2)存在“非瞬间性”的缺陷。在混合加密体制中,能够强迫接受者出示其私钥的搭线窃听者,就能够恢复所有的有效信息。这个缺点称之为缺乏“前向保密安全性”。所谓前向保密性是指无论通过分析还是强迫,搭线窃听者都不可能由以前发送的密文在将来的时间恢复出明文消息。^[9]

3 安全的 Diffie-Hellman 密钥交换协议

综上,提出了一种安全的 Diffie-Hellman 密钥交换协议,即混合加密方案的公钥密码部分采用 Diffie-Hellman 密钥交换协议,就能够克服这两个缺点。在 C 和 S 双方运行的 Diffie-Hellman 密钥交换协议中,共享的会话密钥 $K = g^{ab}$ (g 为一个素阶群, a, b 为双方各输入的一个随机数,且满足 $g^a \neq 1$, $g^b \neq 1$) 包含双方的随机输入,假设 C 输入的是 a , S 输入是 b , C(S) 能够确信只要对方用了一个随机的整数,从 g^{ab} 推导的共享秘密会话密钥就是随机的。这是因为映射 $g^b \mapsto (g^b)^a$ 和 $g^a \mapsto (g^a)^b$ 在问题中是群的一个置换,所以均匀分布的指数(小于群的阶)把 $g^a (g^b)$ 映射为一个均匀分布的群元素 g^{ab} 。这样就能保证生成的会话密钥具有随机性和公平性。使用 Diffie-Hellman 密钥交换协议的混合加密方案就具有前向保密性,会话密钥 g^{ab} 的生成是双方的随机指数生成的。即使一方的私钥泄露,也不能根据其私钥推断得到以前的会话密钥,因为每次的会话密钥中都包含了随机生成的信息,保证了新密钥与其他密钥之间的相互独立,提供了密钥独立性和安全前向保密性。为了谨慎地运行 Diffie-Hellman 密钥交换协议, C 和 S 应该在交换他们的会话密钥,并在协议完成后立即销毁 a 和 b , 同时为了正确进行以后的会话通信, C 和 S 还应该在会话结束后销毁他们的会话密钥,并适当地处理他们所通信的明文消息。如果他们遵循这种相当标准的程序,显然强迫手段也不会使搭线窃听者得到 C 和 S 所通信的明文内容。由计算 DH 问题基于离散对数的困难性,此协议具有前向保密性,搭线窃听者的分析也不会成功^[9]。并且,该方案还满足已知密钥安全性,即每一次密钥交换产生的密钥是不同的,当一次会话密钥泄露后,其他次产生的密钥不会因此而泄露。

结合上述思想将改进的 Kerberos 认证方案中流程的第 5、6 步改为:

$C \rightarrow S: ST, \text{Authenticator}_{cc}, EKc. s[EKRc(IDE, R, g^a)]$

$S \rightarrow C: EKc. s[R3', EKRs(R', g^b, K)]$

S 收到 C 的报文用 Kc. s 解密后,用 C 的公钥 KUc 验证 C

对 g^a 的签名—— $EKRc(g^a)$, 若通过验证, S 利用 $K = g^{ab}$ 生成与 C 的两方会话密钥 $K = (g^a)^b$, 再发给 C。

当 C 收到 S 的应答报文用 Kc. s 解密后,用 S 的公钥 KUs 验证 S 对 g^b 和两方会话密钥 K 的签名,若通过, C 同样利用 $K = g^{ab}$ 计算得到 $K' = (g^b)^a$, 然后与 K 比较,如果相同则说明 K 即为他们间的会话密钥。由于会话密钥 K 每次都是随机产生的,每次皆可能不同,所以保证了一次一密的原则,增强安全性。到此为止,就可以实现 C 和 S 间两方的保密通信了,而 Kerberos 并不知道会话密钥 K 故无法窃听 C 和 S 间的通信内容,从而防止了 Kerberos 系统的内部攻击。

4 结语

改进后的 Kerberos 认证协议安全性较高,但还存在需要进一步完善的地方:该混合体制方案中采用的是 RSA 公钥加密体制和 DES 对称加密体制的组合。然而,采用 RSA 其加解密速度较慢,发送方传输前要加密两次,接收方收到后要解密两次。C 和 S 在使用共享密钥 Kc. s 进行加解密时采用的是 DES 对称加密方式, DES 本身的安全性也较弱。所以,为了提高系统的执行性能,可以考虑使用公钥加密体制中的 ECC (椭圆曲线加密体制) 替代 RSA 和对称加密体制中的 AES (Rijndael 算法) 替代 DES。还有, Kerberos 目前还不支持用户注册认证,只提供认证服务,至于系统内的访问权限和授权只有通过其他途径来解决。Kerberos 系统的客户之间的通信仍需要事先交换密钥,如果使用环境增大必然会加重网络的负担等。

参考文献:

- [1] RFC1510, the kerberos network authentication service (V5) [S]. 1993.
- [2] BELLOVIN S M, MERRITT M. Limitation of the Kerberos authentication system [J]. ACM SIGCOMM Computer Communication Review, 1990(5): 119 - 132.
- [3] 许先斌, 陈凡, 苏剑. Kerberos 协议的改进和证明[J]. 计算机工程与应用, 2002, 38(8): 157 - 158.
- [4] 姚传茂. Kerberos 认证系统的研究与改进[J]. 安徽建筑工业学院学报: 自然科学版, 2006, 14(2): 85 - 87.
- [5] 刘克龙, 卿思汉, 蒙杨. 一种利用公钥体制改进 Kerberos 的方法[J]. 软件学报, 2001, 12(6): 872 - 877.
- [6] 张红旗, 车天伟, 李娜. Kerberos 身份认证协议分析及改进[J]. 计算机应用, 2002, (12): 25 - 27.
- [7] 汤卫东, 李为民, 周永权. 利用 ElGamal 算法改进 Kerberos 协议[J]. 计算机工程与设计, 2006, 27(11): 2063 - 2065.
- [8] 文铁华, 谷士文. 增强 Kerberos 协议安全性的改进方案[J]. 通信学报, 2004, 25(6): 76 - 79.
- [9] 毛文波. 现代密码学——理论与实践[J]. 北京: 电子工业出版社, 2004.
- [4] 袁亚飞, 廉玉忠. 3G 认证与密钥分发协议逻辑化分析[J]. 信息工程大学学报, 2004, 5(4): 15 - 17.
- [5] 田荣明, 李方伟. 一种改进的密钥分配与认证协议[J]. 电讯技术, 2004, 44(2): 68 - 71.
- [6] 朱里奇, 黄本雄. 3G 认证和密钥分配协议的形式化分析及改进[J]. 电子工程, 2004, 30(5): 21 - 24.
- [7] 郑宇, 何大可, 梅其祥. 基于自验证公钥的 3G 移动通信系统认证方案[J]. 计算机学报, 2005, 28(8): 1327 - 1331.
- [8] CHENG SHU - MIN, SHIEH SHIUH - PYNG, YANG WEN - HER. Designing authentication protocols for third generation mobile communication systems [J]. Journal of Information Science and Engineering, 2005, 21(2): 361 - 378.
- [9] 刘锋. 第三代移动通信系统中认证和密钥协商协议的应用研究[D]. 重庆: 重庆大学, 2005.
- [10] 姚惠明, 隋爱芬, 杨义先. 3GPP 网络 AKA 协议中若干算法的设计[J]. 北京邮电大学学报, 2002, 25(3): 98 - 102.
- [11] 刘佳潇, 曾光, 韩文报. 改进的 3G 认证与密钥协商协议[J]. 信息工程大学学报, 2006, 7(4): 318 - 322.
- [12] 杨士平, 李祥. BAN 逻辑在协议分析中的密钥猜测分析缺陷[J]. 计算机工程, 2006, 32(9): 126 - 127.

(上接第 1627 页)