基于 DESX 的 Kerberos 协议的分析与改进

王崇霞[1],高美真[2]
(1.长治学院 计算机系,山西 长治 046010;
2.焦作师范高等专科学校 计算机与信息工程学院,河南 焦作 454000)

摘 要:Kerberos 协议是一种基于可信第三方的身份认证协议,针对 Kerberos 协议具有口令猜测攻击、重放攻击等缺陷,提出一种基于 DESX 算法和 SHA 函数的 Kerberos 协议改进方案,通过分析比较,改进协议不但摒弃了原 Kerberos 协议存在的缺陷,且以相对较小的开销使 Kerberos 协议认证过程更安全可靠。

关键词:Kerberos 协议;身份认证;DESX;安全性;票据
中图分类号:TP3 **文献标识码**:A **文章编号**:1674-7720(2013)13-0066-04

Analysis and improve of Kerberos protocol based on DESX

Wang Chongxia[1], Gao Meizhen[2]
(1.Department of Computer, Changzhi University, Changzhi 046010, China;
2.Department of Computer and Information Engineering, Henan Jiaozuo Teachers College, Jiaozuo 454000, China)

Abstract: Kerberos protocol is an identity authentication protocol based on the trusted third party, According to it has password guessing attacks and replay attacks defect, an improved scheme about Kerberos based on DESX algorithm and SHA function was proposed, after the analysis and comparison, the improved protocol not only abandon the original Kerberos protocol flaws, and with less computing cost enables Kerberos authentication process is more safe and reliable.

Key words: Kerberos protocol; identity authentication; DESX; security; ticket

Kerberos 协议是麻省理工学院(MIT)在 20 世纪 80 年代为 Athena 计划开发的一种基于可信第三方协助的统一身份认证协议,它可以在不安全的网络环境中实现用户对远程服务器的访问,并提供自动鉴别、数据安全性和完整性服务,其特点是用户只需输入一次身份验证信息就可以凭此验证获得访问服务器的票据。至今,Kerberos 已经有 5 个版本,前 3 个版本是内部应用版本,Kerberos V4 是被公诸于众的第 1 个版本,Kerberos V5 针对 V4 存在的不足作了改进。但由于 Kerberos 协议基于对称加密算法 DES 实现,仍然存在口令猜测攻击、密钥管理困难等不足[1]。针对这些不足,学者们提出了很多对 Kerberos 协议的改进方案,如参考文献[1]利用公钥密码体制对 Kerberos 协议进行了改进;参考文献[2]基于动态密码体制对 Kerberos 协议进行修正;参考文献[3]基于椭圆曲线的零知识证明方法解决字典攻击问题;参考文献[4]提出基于混合加密体制和 Daffier-Hellman 密钥协商对 Kerberos 协议的修正方案;参考文献[5]提出用 ECC 算法作为加密和签名工具对 Kerberos 协议进行改进。这些方案虽然从不同角度对 Kerberos 协议进行了修正和补充,弥补了 Kerberos 协议的不足,但也在很大程度上修改了 Kerberos 协议算法,增加了运算复杂度,提高了系统开销。

本文在深入分析 Kerberos 协议结构和认证过程的基础上,提出了基于 DESX 及 SHA-2 函数对 Kerberos 协议的改进方案,不但摒弃了 Kerberos 协议存在的不足,增加了 Kerberos 协议的安全性,而且由于 DESX 算法是 DES 算法的变型,在不改变 Kerberos 协议认证模型及开销较小的情况下弥补和修正了 Kerberos 协议的缺陷。

1 Kerberos 协议及其安全性分析

1.1 Kerberos 认证协议

Kerberos 体系结构由密钥分配中心 KDC(Key Distribution Center)、应用服务器和客户 3 个部分组成,

KDC 在整个认证系统中处于核心地位，由认证服务器 AS（Authentication Server）、票据授权服务器 TGS（Ticket Granting Service）及认证数据库组成。

Kerberos 协议以域为单位进行管理，每个域中包含若干客户、1 个认证服务器 AS、1 个票据授权服务器 TGS 和若干应用服务器。因此，Kerberos 认证协议分为两种认证模式：域内认证和跨域认证。本文主要从域内认证模式阐述对 Kerberos 协议的改进。Kerberos 协议认证流程共有 6 个步骤，如图 1 所示。

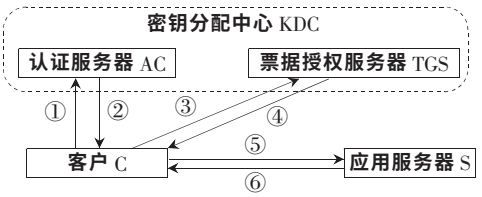

图 1 Kerberos 认证过程模型

本文用到的符号定义：

C：客户；

S：应用服务器；

ID_c：用户的身份标识；

AD_c：用户的地址；

K_x：X 与 AS 的共享密钥；

$K_{x,y}$：x、y 共享的会话密钥；

$Ticket_{x,y}$：访问时的票据；

Lifetime：生存时间；

TS：时间戳，信息的发送时间；

$Authentication_c$：由客户产生，用于验证用户所持票据的真实性；

$\{M\}K_c$：密钥 K_c 加密信息 M；

Option：可选项，用于请求在票据中设置相应的标志位，通过设置相应的标志，可以在协议中添加一些控制；

Nonce：一个保鲜数，它表示回应的信息是新鲜的，预防遭到重放攻击；

Times：客户请求在票据中设置时间，包括请求票据的起始时间、过期时间和过期时间的更新时间，从而允许票据拥有任意长度的有效期；

Subsky：子密钥，用以保护某一特定的应用程序会话；

Seq：可选项，说明在此次会话中服务器向客户发送消息的序列号，将消息排序可以防止重放攻击。

Kerberos 协议的信息交换过程共分 3 个阶段 6 个步骤[6-7]：

（1）客户 C 请求 AS 发放访问 TGS 的许可票据 $Ticket_{tgs}$。C 发送明文请求消息到 AS，AS 返回 1 张加密过的票据，加密密钥 K_c 由用户口令 Password 得到。当 AS 回应的信息到达客户端，客户提示用户 C 输入口令，由此产生密钥 K_c，并对收到的报文解密，得到 $Ticket_{tgs}$，若口令正确，票据正确解密，恢复加密的许可票据 $Ticket_{tgs}$，否则提示错误返回。

①$C \rightarrow$AS：$\{Options, ID_c, ID_{tgs}, Times, Nonce_1\}$

②AS$\rightarrow C$：$\{ID_c, Ticket_{tgs}\{K_{c,tgs}, Times, Nonce_1, ID_{tgs}\}K_c\}$

$Ticket_{tgs}=\{flags, K_{c,tgs}, ID_c, AD_c, Times\}K_{tgs}$

（2）客户 C 访问 TGS 获得访问应用服务器 S 的服务许可票据 $Ticket_s$。TGS 对收到的许可票据 $Ticket_{tgs}$ 解密，并核查解密后的信息，检查票据有效期 Times 和新鲜数 $Nonce_2$，核定该票据是否过期；比较票据中的用户信息与收到的数据包中的用户鉴别信息是否一致，确定用户是否为合法用户，如用户合法，发放访问应用服务器的许可票据 $Ticket_s$。用户鉴别信息 $Authentication_{c1}$ 由用户 C 与 TGS 之间的共享密钥加密 $K_{c,tgs}$。

③ $C \rightarrow$TGS：$\{Options, ID_s, Time_s, Nonce2, Ticket_{tgs}, Authentication_{c1}\}$

$Authentication_{c1}=\{ID_c, TS_1\}K_{c,tgs}$

④TGS$\rightarrow C$：$\{ID_c, Ticket_s, \{K_{c,s}, Times, Nonce2, ID_s\}K_{c,tgs}\}$

$Ticket_s=\{flags, K_{c,s}, ID_c, AD_c, Times\}K_s$

（3）客户 C 据许可票据 $Ticket_s$ 访问应用服务器，并进行相互身份验证。客户持许可票据 $Ticket_s$ 申请访问应用服务器 S，S 解密信息（实现客户对应用服务器身份确认），并根据 $Ticket_s$ 和用户鉴别信息 $Authentication_{c2}$，验证用户身份，构建回应信息包。客户产生与应用服务器间会话的子密钥 Subkey 和序列号 seq，来确保信息传输的安全性和信息传输的顺序性。

⑤$C \rightarrow S$：$\{Options, icket_s, Authentication_{c2}\}$

$Authentication_{c2}=\{ID_c, TS_2, Subkey, Seq.\#\}K_{c,s}$

⑥$S \rightarrow C$：$\{TS_2, Subkey, Seq.\#\}K_{c,s}$

1.2 Kerberos 的安全性分析

Kerberos 协议是目前计算机网络环境中应用最广泛的第三方认证协议，Kerberos 协议虽然能够实现身份认证，并提供数据完整性和保密性服务，但却存在着一定的安全缺陷。参考文献[1]、[7]、[8]列举了 Kerberos V4 协议存在的安全问题，主要有以下几方面：

（1）口令猜测攻击：Kerberos 协议中，AS 并不直接验证用户的口令，只是通过用户口令产生的密钥能否解密来判断用户的合法身份，若攻击者捕获该消息，并尝试各种口令解密，如果解密成功，攻击者即可得到用户口令，进而冒充用户身份访问服务器。

（2）重放攻击：虽然 Kerberos V4 采用时间戳预防重放攻击，但在时间戳许可的有效时间内，攻击者若把事先准备好的伪造消息发出，服务器就很难判断信息是否是伪造的。

（3）票据有效期有限：Kerberos V4 用 8 bit 表示票据的有效期，因此其最大有效期约 256×5=1 280 min[8]。

除了以上所描述的缺陷，Kerberos V4 还存在消息字节顺序由发送者决定、域间认证不完善、采用非标准的 PCBC 加密模式、多次使用同一个会话密钥等不足。

Kerberos V5 在 V4 的基础上进行了修改，改用了标准的 CBC 加密模式、对消息字节的发送顺序也作了一定的规定，并且可以设置一些标志位，增加了 Nonce 保鲜数、机动设置票据时间、用户访问应用服务器时产生子密钥、添加发送消息的序列号等，解决了大部分 Kerberos V4 存在的问题，但 Kerberos V5 依然存在口令猜测攻击的致命缺陷。

2 Kerberos 协议改进方案

2.1 DESX 加密算法

数据加密标准 DES(Data Encryption Standard)是由 IBM 公司研制的一种分组对称加密算法，自发布以来，DES 在各行各业得到广泛应用。DESX 算法是 RSA 数据安全公司对 DES 算法的改进，它采用白化技术来掩饰输入和输出，除了有 DES 的 56 bit 密钥外，DESX 还有附加的随机密钥，将总密钥长度扩展到 184 bit，即使用 3 个 64 bit 的密钥 $K1$、$K2$ 和 $K3$ 对数据块加密，充分保证了数据的安全。DESX 加密过程分为 3 个步骤：(1)使用 $K2$ 对数据块 C 做 XOR 运算；(2)使用 $K1$ 对上一步结果做 DES 加密运算；(3)使用 $K3$ 对第二步的结果进行XOR 运算得到密文。即 $DESK_{k1,k2,k3}(C)=k3\oplus Des_{k1}(k2\oplus C)$(密钥 $K=k1.k2.k3$ 符号“.”表示级联)[9]。

与 DES 和 3DES 相比，DESX 只做了 1 次 DES 运算，计算开销和普通 DES 相当，但密钥长度提高到 184 bit，在保证安全性前提下，极大地提高了执行效率，且白化技术迫使攻击者不仅要猜出算法密钥，而且必须猜出 1 个白化键，因此 DESX 抵御各种网络攻击的能力更强。

当 DESX 的密钥 $K=k1.k2.k3=k1.0^{64}.0^{64}$ 时，DESX 与 DES 兼容[9]。

2.2 SHA-2 函数

Hash 函数可以将任意长的数据映射为定长的 Hash 码，也称为数据摘要，即 $h=H(M)$。它具有单向性、抗碰撞性等特点，在身份认证、数字签名和完整性检验等方面得到了广泛应用。

安全散列算法（SHA）是由美国国家标准与技术协会 NIST 设计发布的安全 Hash 函数，1995 年发布了 FIPS180-1，称之为 SHA-1；由于 SHA-1 存在安全隐患，2002 年，NIST 又发布其修订版 FIPS180-2，并称为 SHA-2，其中包含 3 种新的 Hash 函数，因其 Hash 值长度分别为 256 bit、384 bit 和 512 bit，故分别称为 SHA-256、SHA-384 和 SHA-512。

SHA-2 是一种迭代结构的 Hash 函数，它可以把任意长度的输入数据压缩成固定长度的数据摘要。SHA-2 不但具有 Hash 函数抗碰撞性、计算不可逆等特点，且 SHA-2 的数据分组和摘要信息都比 SHA-1 大，所以它具有更高的安全性[10]，其在信息安全领域的应用也越来越多。

2.3 建议的 Kerberos 协议改进方案

通过 1.2 节对 Kerberos 协议安全性的分析，对 Kerberos 协议的改进主要从以下几个方面进行：

(1)针对加密算法，原 Kerberos 协议采用的 DES 算法密钥较短、强度较弱、存在互补对称性等缺陷，建议 Kerberos 协议以 184 bit 密钥的 DESX 算法对数据进行加密，提高了 Kerberos 协议的安全性。

(2)针对口令猜测攻击，取消认证过程中的直接口令认证，改为采用 SHA-2 函数计算的多参数密钥认证，并在数据通信过程中多次实现对用户的身份确认。

建议的 Kerberos 协议模型与原 Kerberos 协议模型相似，认证过程中每个阶段的修改主要有以下几点：

(1)C AS，客户 C 请求认证服务器 AS 发放访问票据授权服务器 TGS 的许可票据 $Ticket_{tgs}$。

客户 C 发送认证信息给认证服务器，认证服务器收到信息后反馈一个密钥 K_c 加密的信息及票据 $Ticket_{tgs}$，客户收到信息后，输入口令 Password 作为解密密钥 K_c。在原 Kerberos 协议中，密钥 K_c 即用户口令 Password，而建议的 Kerberos 协议中密钥采用 SHA-2 函数计算得到，即：

K_c=Hash(Password+用户信息+Times)

Password 是用户在系统中注册的口令信息，KDC 认证数据库中仅存储用户 Password，认证服务器在需要使用 K_c 加密数据时，通过 Hash 函数计算得到 K_c；用户信息关于用户的一些信息，也可以是 ID_c 或 AD_c 等；Times 为客户在请求票据中设置的时间，包括票据的起始时间、到期时间和到期时间的更新时间。

(2)C TGS 客户 C 访问 TGS 获得访问应用服务器 S 的服务许可票据 Tickets。

客户收到 K_{tgs} 加密的票据 $Ticket_{tgs}$，同 ID_c、Times、$Authentication_{c1}$ 等信息一起发送到票据授权服务器 TGS，其中用户鉴别信息 $Authentication_{c1}$ 的作用是验证用户所持票据的真实性，确保票据是被拥有者所持有。但是如果攻击者捕获该票据，并冒用 ID_c 从另一个工作站上发送消息，会误导票据授权服务器 TGS 授权给攻击者。所以建议的 Kerberos 协议在 $Authentication_{c1}$ 中添加用户身份信息，确保信息发送者是服务器授权的用户身份。

$Authentication_{c1}=\{ID_c, AD_c, TS_1\}K_{c,tgs}$

(3)S C，客户 C 访问应用服务器 S，并实现 C 和 S 的双向身份认证。

客户通过与应用服务器间会话的子密钥 Subkey 和序列号 Seq 来确保信息传输的安全性和顺序性。客户持授权票据 $Ticket_s$ 访问应用服务器 S，应用服务器通过客户所持票据 $Ticket_s$ 对其身份进行认证，同时在用户鉴别信息 $Authentication_{c2}$ 中添加用户信息 AD_c，防止攻击者冒充客户身份，实现应用服务器和客户的双向确认。

$Authentication_{c2}=\{ID_c, AD_c, TS_2, Subkey, Seq.\#\}K_{c,s}$

2.4 改进的 Kerberos 协议分析

(1)加密算法的改进

在建议的 Kerberos 方案中采用 DESX 加密算法，DESX 算法是为了克服 DES 算法容易受到口令猜测攻击而提出的改进加密算法，它采用当前密码学中流行的级联密码和白化技术，将密钥长度扩展到 184 bit，其密钥 $K2$、$K3$ 被称为白化键，它采用混合计算排列的输入和输出来掩盖密钥[9]。因此采用 DESX 加密算法的Kerberos 协议安全性更强。

(2)口令猜测攻击

在原 Kerberos 协议中，认证服务器通过用户是否能解密信息来判断用户的合法身份，且用户密钥是预先存储在 AS 中的不变的用户信息，由于通常用户所选择的密码比较简单或具有明显的特征，极易遭到口令猜测攻击。而建议的 Kerberos 方案中，用户密钥是通过多参数的 Hash 函数计算而来，且每次用户认证时，密钥会随当时认证时间、参数等信息不同而不同，由于 Hash 函数的不可逆性，攻击者通过口令猜测攻击能得到是不可能实现的。

(3)重放攻击

Kerberos V4 协议中，在时间戳允许的范围内，攻击者可以把事先准备好的伪造消息发出，进行重放攻击。在建议的 Kerberos 方案中不仅设置了时间戳 TS、生存期 Times 和保鲜数 Nonce，并且在认证过程中协商产生序列号 Seq，杜绝了重放攻击的发生。另外在与 TGS 和 AS 的交互过程中，鉴别信息 $Authentication_{c1}$ 和 $Authentication_{c2}$ 中添加用户身份信息 AD_c，防止攻击者冒充客户身份访问服务器，更加强了对重放攻击的预防。

(4)用户身份确认

在建议的 Kerberos 协议的实现过程中，每个阶段都对用户的身份进行确认，第一次，通过密钥解密信息来确认用户身份，第二次和第三次分别是客户与 TGS 和 AS 交互过程中，通过在加密了的鉴别信息 $Authentication_{c1}$ 和 $Authentication_{c2}$ 中添加用户身份信息 AD_c，实现对用户身份的确认，保障认证过程中每个阶段对客户身份的鉴别，防止攻击者冒充客户身份。

建议的 Kerberos 改进方案采用 DESX 作为加密算法，DESX 算法的计算开销和普通 DES 相当，且协议认证过程仅需执行一次 Hash 计算，因此在协议实现复杂度与加解密速度上，较参考文献[1]、[2]、[3]、[4]、[5]提议的 Kerberos 改进方案[11-12]，存在明显的改善。

本文详细分析了 Kerberos 认证协议，并针对 Kerberos 协议存在的安全隐患，提出了一种基于 DESX 加密算法和 SHA-2 函数的 Kerberos 改进方案。通过分析比较，建议的 Kerberos 改进方案弥补了原 Kerberos 协议存在的口令猜测攻击和重放攻击等缺陷，提高了协议认证过程的安全性，加强了协议认证过程中各阶段对客户身份的鉴别，有效防范了攻击者冒充客户身份，并且建议的协议方案与原协议实现模型类似，在 Kerberos 协议逐渐淘汰 DES 加密算法的过程中，具有很好的实用意义。

参考文献

[1] RAVI GANESAN. Yaksha: Augmenting Kerberos with Public Key Cryptography[J]. Network and Distributed System Security, 1995(6):132-143.

[2] 卢小良，袁丁.对一种基于动态密码体制的 Kerberos 协议的改进[J].四川师范大学学报，2006，29(2)：239-242.

[3] 张健，戴威岭，郝善勇.基于椭圆曲线的零知识证明方法对 Kerberos 系统的改进[J].计算机工程，2002，28(10)：143-144.

[4] 胡宇，王世伦.基于混合体制的 Kerberos 身份认证协议的研究[J].计算机应用，2009，29(6)：1659-1661.

[5] 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.

[6] FREDERICK BUTLER, ILIANO CERVESATOB, AARON D. Jaggarde, Andre Scedrovd, Christopher Walstadd, Formal analysis of Kerberos 5[J]. Theoretical Computer Science, 2006:57-87.

[7] STEVEN M. BELLOVIN, MICHAEL MERRITT. Limitations of the Kerberos Authentication System [J]. http://hdl.handle.net/10022/AC:P:9123, 1999.

[8] 周倜，王巾盈，李梦君，等.Kerberos 协议版本的分析与比较[J].计算机学报，2009，36(2)：119-123.

[9] JOE KILIAN. How to Protect DES Against Exhaustive Key Search (an Analysis of DESX)[J]. Journal of Cryptology, 2001(14):1,17-35.

[10] CHU HSING LIN, YI SHIUNG YEH, HUNG SHENG CHIEN, et al. Generalized secure hash algorithm: SHA-X[J]. International Conference on Computer as a Tool (EUROCON), IEEE, 2011:27-29.

[11] 莫燕，张玉清，李学干.对 Kerberos 协议的攻击及对策研究[J].计算机工程，2005，30(10)：66-69.

[12] NEUMAN, KERBEROS B C. An authentication service for computer networks[J]. Communications Magazine, IEEE. 1994,32(9):33-38.

(收稿日期：2013-03-10)

作者简介：

王崇霞，女，1970 年生，硕士，副教授，主要研究方向：计算机网络与信息安全，密码学。