Kerberos 身份认证协议分析与改进①

游新娥，胡小红

(娄底职业技术学院，娄底 417000)

摘　要：对 Kerberos 协议及安全性进行了较详细地分析，针对该协议存在的缺陷，提出了一种改进的认证模型。该模型引入轻量级票据，采用混合密码体制和 USBKey 双因素认证，较好地解决了口令猜测攻击、重放攻击、密钥存储困难等问题，具有较好的安全性与易实现性。

关键词：Kerberos 协议；票据；USBKey 双因素认证

Analysis and Improvement of Kerberos Identity Authentication Protocol

YOU Xin-E, HU Xiao-Hong

(Loudi Vocational & Technical College, Loudi 417000, China)

Abstract: This paper discusses Kerberos protocol and its security in detail, proposes an improved authentication model according to its limitations as well. This model aims at solving the problems such as password guess attack , replay attack and key storage management by using lightweight ticket and hybrid cryptosystem and USBKey two-factor authentication, which is safe and easy to realize.

Key words: Kerberos Protocol; ticket; USBKey two-factor authentication

1　引言

Kerberos 身份认证协议是美国麻省理工学院(MIT)的 Athena 项目小组提出的基于可信的第三方高效认证机制，旨在为开放网络环境中的服务请求提供安全保障，既可用于身份认证，也可用于保证数据的完整性与保密性。至今，Kerberos 已经了五个版本，其中前三个版本是内部应用版本。Kerberos V4 是被公诸于众的第一个版本，Kerberos V5 针对 V4 存在的一些安全漏洞作了改进，但因 Kerberos 协议基于对称密码体制，仍然存在口令猜测攻击，密钥管理困难等方面的不足[1]。针对这些不足，人们提出了很多对 Kerberos 协议进行改进的方案，如文献[1]利用混合加密体制和安全的 Diffie-Hellman 密钥协商对 Kerberos 协议改进了改进，文献[2] 将一次性口令认证协议 EOTP 和 AES 相结合，给出了新的 Kerberos 改进协议，文献[3]将视觉密码技术融入到 Kerberos 协议当中，构建出了一套新的认证方案，文献[4]与文献[5]分别提出了基于身份的密钥协商协议和基于无证书密钥协商协议的 Kerberos 认证方案，文献[6]参考 SAML 中的 Artifact 概念对 Kerberos 中的票据进行了改进，这些方案从不同角度改善了 Kerberos 协议的安全性，各有其优点，但也有其不足之处[2]。本文在深入分析 Kerberos 协议的体系结构和认证过程的基础上，在文献[1]和文献[6]的基础上，通过引入轻量级票据，采用混合密码体制和 USBKey 双因素认证提出了一个改进的 Kerberos 协议认证模型，使其更加安全可靠且具有更高运行效率。

2　Kerberos认证协议及安全性分析

2.1 Kerberos 认证协议

Kerberos 的实现包括密钥分发中心 KDC(key distribution center)以及一个可供调用的函数库[3]。KDC 包括认证服务器 AS(authentication server)和票据服务器 TGS(ticket granting server)。AS 的作用是对用户的

① 基金项目:湖南省教育厅科研项目(09C1271)
收稿时间:2011-07-08;收到修改稿时间:2011-08-18

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[4]。

Kerberos 认证过程如图 1 所示分三个阶段六个步骤。

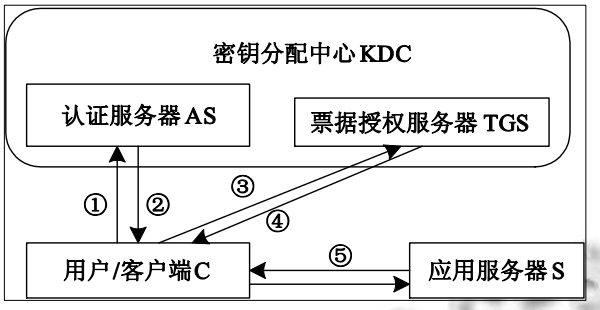

图 1 Kerberos 认证过程模型

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

1) C→AS：IDC‖IPC‖IDTGS‖timestamp1

客户端 C 向认证服务器 AS 发出访问 TGS 的请求，请求内容包括：客户标识 IDC、客户 IP 地址 IPC、TGS 标识 IDTGS、时间戳。

2) AS→C： KC[SKC,TGS‖IDTGS‖timestamp 2‖TC,TGS]

TC,TGS=KTGS [IDC‖IDTGS‖IPC‖SKC,TGS]

当 AS 收到 C 的请求报文后，产生一个用于客户与 TGS 加密通信的随机会话密钥 SKC,TGS，并将客户标识、客户 IP、TGS 标识、会话密钥使用 TGS 的密钥 KTGS 进行加密作为客户访问 TGS 的授权票据 TC,TGS。客户接收到 AS 的响应后，用自己的密钥解密，得到授权票据和与 TGS 通信的会话密钥。

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：IDS‖TC,TGS‖AC ,TGS

AC ,TGS = SKC,TGS [IDC‖IPC‖timestamp3]

客户端对 AS 返回的信息进行解密，将得到的 TC,TGS 连同验证码 AC,TGS 与要访问的应用服务器标识 IDS 一起发送给 TGS。

4) TGS→C： SKC,TGS[IDS‖SKC,S‖timestamp 4‖TC,S]

TC,S =KS[IDS‖IDC‖IPC‖SKC,S]

TGS 比较 TC,TGS 和 AC,TGS 的信息是否一致来验证客户端的身份，如果身份被确认，TGS 返回访问应用服务器的票据 TC,S 和会话密钥 SKC,S 给客户端。

第三阶段：客户端 C 与应用服务器 S 间相互验证身份。

5) C→S：AC,S‖TC ,S

AC,S=SKC,S[IDC‖IPC]

客户端将接受到的消息进行解密得到会话密钥 SKC,S，并把 TC,S 和验证码 AC,S 一起发送给服务器。

6) S→C：SKC,S[timestamp4 +1]

服务器比较 TC,S 和 AC,S 是否一致来验证客户端的身份，如果身份被确认，就将时戳加一返回给客户端。客户端比较时戳的有效性实现对应用服务器的认证。

整个认证过程完成后，客户端和服务器端就用会话密钥 Kc,s 进行通信。

1.2 Kerberos 协议安全性分析

Kerberos 协议是目前分布式网络环境中应用最广泛的第三方认证协议，但其本身却存在固有的安全缺陷。从安全角度和认证过程来看，主要有以下几方面：

(1) 口令猜测攻击

在 Kerberos 协议认证过程中，用户向 AS 请求认证时，AS 会把相关信息用客户端的密钥 Kc 加密后发送给客户端。该密钥是根据客户输入的口令随机生成的。攻击者可以收集大量相关信息，通过计算和密钥分析来进行口令猜测。如果攻击者掌握了足够的信息，就有可能猜测出用户的口令。若用户选择的口令不够强，就不能有效地防止口令猜测攻击。

(2) 重放攻击与时钟同步问题

在 Kerberos 协议中，为了防止重放攻击引入了时间戳，这就要求在整个网络内的时钟实现同步，但由于变化的和不可预见的网络延迟的本性，不能期望分布式时钟保持精确的同步。一般来说，在认证过程中时间相差 5 min 就认为是新的消息[5]。这样，时戳就带来了重放攻击的隐患，在规定的时间内，攻击者完全可以事先把伪造的消息准备好，一旦得到票据就马上发出。这在规定的时间内是很难被察觉的。

(3) 密钥管理与维护

KDC 保存大量的共享密钥，密钥管理与分配较复杂，需要特别细致的安全管理措施，无论 KDC 攻破与否，都将付出极大的系统代价。

3 Kerberos协议的改进

基于以上所分析的 Kerberos 协议的安全缺陷，本文通过对票据进行改进，结合公钥密码体制、数字证书和 USBKey 双因素认证等安全技术来提高协议的安全性。

3.1 改进的票据模型

改进的票据模型分为票据和票据引用两部分[6]。当 AS 对用户身份认证成功后，为用户创建 TGT 票据和 TGT 票据引用。TGT 票据记录了用户的登录信息：用户 ID、用户 IP、用户数字证书、票据标识、票据创建时间、票据作废时间等。而 TGT 票据引用是轻量级的用户证书，即将票据标识使用 AS 私钥签名并用 TGS 公钥加密，其形式为：PUKTGS[SessionID，PRKAS[SessionID]]，其中 PUKTGS 为 TGS 的公钥，PRKAS 为 AS 的私钥，SessionID 为 TGT 票据标识。

同理，ST 票据引用形式为：PUKS[SessionID，PRKTGS[SessionID]]，其中 PUKS 为 S 的公钥，PRKTGS 为 TGS 的私钥，SessionID 为 ST 票据标识。

2.2 改进的 Kerberos 协议认证模型

改进的认证模型引入了 CA 来进行证书颁发和管理工作，证书存储在 LDAP 目录服务器中，客户端数字证书同时存放在 USBKey 中。

第一阶段：客户端与认证服务器交互，获得票据许可票据。

1) C→AS：PUKAS[IDC‖IPC‖IDTGS‖CertC]

客户向 AS 发送请求与 TGS 建立联接的消息，消息包括客户 ID、客户 IP、TGS 的 ID、客户数字证书 CertC，USBKey 用 AS 的公钥 PUKAS 对消息进行加密。

2) S→C：PUKC[SKC,TGS‖SKC,TGS[TRC,TG S]]

TRC,TGS=PUKTGS[SessionID‖PRKAS[SessionID]]

AS 解密消息，得到 IDC、CertC，通过 IDC 查询 LDAP 服务器得到对应客户的数字证书副本，将 CertC 与副本进行比较，若相符，产生一个票据 TC,TGS、票据引用 TRC,TGS 及用于客户端 C 与 TGS 之间秘密通信的会话密钥 SKC,TGS。将票据存放到中心数据库中，以备以后验证，将会话密钥与加密的票据引用使用客户端的公钥再次加密并返回给客户端。

第二阶段：客户端与票据服务器交互，获得服务许可票据。

3) C→TGS： PUKTGS[SKC,TGS‖SKC,TGS[TRC,TGS]‖IDS]

客户用私钥解密 AS 传来的消息，得到与 TGS 的会话密钥 SKC,TGS 和密文票据引用 TRC,TGS。然后请求与 TGS 通信以获得对应用服务器 S 的访问权。客户发给 TGS 的信息包括：IDS、SKC,TGS 和用 SKC ,TGS 加密的 TRC ,TGS，这些信息使用 TGS 的公钥进行加密，以保证只有 TGS 才能解密。

4) TGS→C：SKC,TGS [TRC,S‖SKC,S]

TRC,S =PUKS[SessionID‖PRKTGS[SessionID]]

① TGS 用私钥解密客户发来的消息，得到 SKC,TGS 和加密的票据引用 TRC,TGS。

② TGS 用 SKC,TGS 解密得到明文引用，用自己的私钥 PRKTGS 解密 TRC ,TGS，得到访问 TGS 票据 TC,TGS 的 SessionID ，用 AS 的公钥解密 AS 对 SessionID 签名，比较两个 SessionID ，确定票据引用确实为 AS 所签发。

③ 通过 SessionID 在数据库中查找对应的票据，验证时间有效性。

④ 若符合，TGS 产生许可票据 TC,S 及其引用 TRC,S，以及客户和 S 之间的会话密钥 SKC,S，并用 SKC,TGS 将消息加密传给客户。

第三阶段：客户端与应用服务交互，获得服务。

5）C→S：PUKS[SKC,S‖SK C,S [TRC,S] ‖R1]

客户使用 SKC,TGS 解密 TGS 返回的消息，得到访问 S 的票据引用 TRC,S 和用于与 S 进行通信的会话密钥 SKC,S；客户产生一随机数 R1，将 SKC,S、用 SKC,S 加密的票据引用 TRC,S 及随机数 R1 一起使用 S 的公钥加密后发送给 S。

6) S→C：SKC,S["SUCCESS "‖R1]

① 应用服务器 S 收到消息后用自己的私钥解密，得到与 C 通信的会话密钥 SKC,S 和加密票据引用 TRC,S；

② S 使用 SKC,S 解开 TRC,S，再用私钥解开 TRC,S 得到 SessionID，并用 TGS 的公钥解密 TGS 对 SessionID 的签名，比较两个 SessionID ，确定票据引用确实为 TGS 所签发；

③ 如果验证通过，则由 SessionID 在数据库中查找相对应的票据，并验证时间的有效性。

④ 如符合，S 用 SKC,S 返回一个消息，该消息包括一个确认符号和随机数 R1，用于 C 验证 S 的身份。

⑤ 当 C 验证 S 的身份通过后，C 便可以获得 S 的服务资源了，至此整个认证协议交换过程结束。

4 改进模型安全性分析

(1) 避免了口令猜测攻击

原 Kerberos 认证模型中 AS 发送给客户端的信息是使用客户端的密钥进行加密的，而该密钥是根据客户输入的口令生成的，攻击者可以收集大量信息，通过计算与分析进行口令猜测攻击。

在改进的认证模型中，认证服务器 AS 与客户端 C 通信时，使用 C 的公钥对消息进行加密，从而保证了只有 C 才能解密，有效避免了口令猜测攻击。同时，在客户端使用 USBKey 实现双因素认证，每个 USBKey 都有一个 PIN 码，客户登录时必须同时有 PIN 码和 USBKey，二者缺一不可，这样即使客户在不安全的环境中登录时造成了口令的泄露，而攻击者因不拥有 USBKey 也无法实现攻击。

(2) 减少了对时钟同步的依赖，防止了重放攻击

原 Kerberos 认证模型对时钟同步的依赖性较高，在分布式网络环境中，要保证系统各机器的时间一致相当困难。虽然在票据中含有有效时间，但攻击者可事先在这个时间内将伪造消息准备好，并截取客户与 AS 之间的会话密钥，从而得到 TGT 票据进行重放攻击。

在改进的认证模型中，采用随机数代替时戳，减少了对时钟同步的依赖；同时客户端与 AS 之间的通信采用公钥加密，只有客户用自己的私钥解密才能得到与 TGS 之间的会话密钥，而客户私钥存放在客户的 USBKey 中，有效地避免了攻击者截取会话密钥进行重放攻击。

(3) 减轻了 KDC 密钥存储管理负担

原 Kerberos 认证模型使用对称密钥密码体制，KDC 存储大量的共享密钥和客户密钥，管理维护复杂。

在改进的认证模型中，采用公钥密码体制，由 CA 负责公/私钥产生和数字证书的签发，USBKey 存放客户的私钥，KDC 只存储公钥信息，密钥管理和分配任务大大降低，同时即使攻击者攻破 KDC 的数据库得到的也只是公钥，不能形成有效的攻击，大大提高了系统的安全性。

(4) 减轻了网络传输负载，提高了票据的安全性

在原 Kerberos 认证系统中，在客户与 TGS 及客户与应用服务器之间传送的都是票据，而票据中包含重要客户认证信息，当这些包含重要客户认证信息的票据在安全性较低的客户浏览器和 Web 服务器之间进行传播时，不但增加了信息泄漏的可能性，而且加重了网络传输负载。

在改进的认证模型中，引入了轻量级票据(即票据引用)，在认证过程中传输的是票据引用而不是票据，一方面保证了票据的安全，另一方面降低了网络传输负载。

4 结语

本文在充分分析 Kerberos 协议认证过程的基础上，指出该协议存在的缺陷，提出了一种改进的认证模型。在改进的认证模型中为了有效地避免口令猜测攻击与重放攻击，在客户端使用 USBKey 存储客户端数字证书及私钥，由 USBKey 与 PIN 码实现双因素认证，采用基于数字证书的强认证方式，安全性较高，但硬件成本也较高，在实际应用中，可将传统的“用户名/密码”认证方式与数字证书认证方式结合起来，根据实际安全需求，灵活采用两种认证方式中的一种。通过实践，，改进的模型易于实现，具有较高的机密性、完整性和认证性。

参考文献

1 胡宇,王世伦.基于混合密码体制的 Kerberos 身份认证协议的研究.计算机应用,2009,29(6):1659–1661

2 张利华,杨秀青.Kerberos 协议的安全性增强方案.计算机工程与设计,2009,30(9):2124–2126.

3 胡志刚,曾巧平.基于视觉密码的 Kerberos 改进协议.计算机工程,2009,35(18):159–163.

4 邬春学,刘柳生.基于身份的密钥协商协议对 Kerberos 的改进.上海理工大学学报,2010,32(4):305–308.

5 陈家琪,冯俊等.无证书密钥协商对跨域 Kerberos 的改进.计算机工程,2010,36(20):150–152.

6 刘铮.基于改进 Kerberos 协议的单点登录系统研究与实现.重庆:重庆大学,2010.