# Kerberos协议版本的分析与比较

周 倜[1] 王巾盈[2] 李梦君[1] 李舟军[3]
(国防科技大学计算机学院 长沙 410073)[1] (北京跟踪与通信技术研究所 北京 100094)[2]
(北京航空航天大学计算机学院 北京 100080)[3]

**摘 要** Kerberos协议是由MIT开发出来的、广泛地用于对用户及网络连接进行安全认证服务的一种安全验证机制。基于共享密钥的方式,Kerberos协议保障了在不安全网络上进行服务会话的保密性和完整性。详细介绍了基于传统的保密性而建立起来的Kerberos协议版本4的基本原理,描述了Kerberos协议版本4的认证结构,指出了其不足之处。最后系统地阐述了版本5的消息交换过程和相关改进。

**关键词** Kerberos协议,安全,认证,缺陷

## Analysis and Comparison of the Kerberos Protocol's Versions

ZHOU Ti[1] WANG Jin-ying[2] LI Meng-jun[1] LI Zhou-jun[3]
(School of Computer, National University of Defence Technology, Changsha 410073, China)[1]
(Beijing Institute of Tracking and Communications Technology, Beijing 100094, China)[2]
(School of Computer, Beihang University, Beijing 100083, China)[3]

**Abstract** The Kerberos protocol developed by MIT has been widely used in the security authentication service for users and network connection. Based on the way of the communication with share-secret-key, Kerberos protocol protects the privacy and integrity of communication with those services in an insecure network. This paper introduced the principle of Kerbores 4 which was built up on the traditional security, described the structure and shortcomings of authentication in Kerberos 4, and presented how to interchange message and improve deficiencies in Kerberos 5.

**Keywords** Kerberos protocol, Security, Aauthentication, Deficiencies

## 1 引言

信息安全对今天的网络系统来说,是一个非常重要又非常严重的问题,而安全协议使用加密技术在开放的互联网上实现密文传送、身份认证等功能,它是实现安全电子商务的基本保证。因此,安全协议的分析、研究和验证是一个迫在眉睫的问题,大规模复杂协议的验证又是其中的重中之重。目前,越来越多的网络软件都开始使用Kerberos协议作为其验证协议,因而验证Kerberos协议成为安全协议领域的一个热点问题。

Kerberos是作为MIT(Massachusetts Institute of Technology)的Athena计划的认证服务而开发的、在Internet上长期被采用的、基于共享密钥方式的一种安全验证机制。Kerberos协议定义了一系列客户机/密钥发布中心(Key Distribution Center, KDC)/服务器之间进行的获得和使用Kerberos票据的通信过程。Kerberos还具有强化互操作性的优点。在一个多种操作系统的混合环境中,Kerberos协议提供了通过一个统一的用户数据库为各种计算任务进行用户验证的能力。即使在不同操作系统平台上通过KDC验证的用户,也可以通过KDC域之间的信任关系,获得无缝的网络服务访问。目前,Kerberos版本已成为网络上应用的一项标准。

在Kerberos发布的第一个报告[3]中阐述了Kerberos的4大需求:安全性、可靠性、透明性和可伸缩性。为了支持这些要求,Kerberos的整体设计方案是基于协议的可信任第三方认证服务的,客户和服务器均信任Kerberos认证服务器的认证。如果Kerberos协议设计充分,则认证服务的安全性取决于Kerberos服务器的安全性。

本文是在做安全协议验证时,对Kerberos 5进行分析、建模、最终实现对该协议的验证工作的一部分。在本文里,主要研究了Kerberos在域内认证的工作过程、版本4的缺陷和版本5的改进之处。

## 2 Kerberos 4简介

Kerberos协议是针对开放网络上多工作站都使用共同的系统软件而设计的。因此,在设计时有两个主要目标:1)让网络服务器(TGS或应用服务)能够证实票据使用者就是票据所有者;2)应用服务器应向用户证实自己身份。本节将具体分析Kerberos 4是如何实现这两个目标的。

到稿日期:2008-03-20 本文受国家自然科学基金(60073001,60473057)资助。

**周 倜** 博士研究生,CCF会员,研究方向为安全协议形式化验证,E-mail:tzhou@nudt.edu.cn, tzhou99@gmail.com;**王巾盈** 女,硕士,工程师,研究方向为计算机网络、信息安全;**李梦君** 博士,讲师,研究方向为安全协议形式化验证技术;**李舟军** 博士,教授,博士生导师,研究方向为进程代数理论、安全协议形式化验证、数据挖掘技术。

Kerberos 协议希望工作站一经启动，用户只需输一次口令，也即当客户需要申请新的服务的票据时，不需再次输入口令。因此，Kerberos 协议定义了一个认证服务器(*AS*)，*AS* 向客户发放票据授权票据，这样当用户需要申请新的服务时就不用再次使用口令去取新的服务票据了。客户获取票据后，即可向服务授权服务器申请获得与应用服务器交互的会话密钥。

Kerberos 协议首先假定在同一个域内，有一个认证服务器(*AS*)、若干客户端(*C*)以及多个服务授权服务器(*TGS*)和应用服务器(*S*)。Kerberos 服务器有存放用户标识和用户口令的数据库，所有用户必须在 Kerberos 服务器注册。同时，Kerberos 服务器与每个应用服务器共享一个特定的密钥，所有应用服务器必须在 Kerberos 服务器注册。

在本文中定义以下简写：$AD_i$ 是 $i$ 的网络地址；$\{ABC\}K$ 表示用密钥 $K$ 将明文 $ABC$ 加密后的密文；$K_i$ 是客户 $i$ 与认证服务器之间的共享密钥；$K_{i,j}$ 是 $i$ 与 $j$ 之间的共享会话密钥；$Lifetime_i$ 是第 $i$ 个有效期(或称为“(当前票据的)生命期”)；$Ticket_i$ 是发给 $i$ 的票据，该票据已用 $i$ 与认证服务器之间的共享密钥加密，用以向 $i$ 证实与 $i$ 通讯的客户身份；$Authenticator_c$ 是 $C$ 发给服务器的认证信息，表明拥有票据的确实是客户 $C$ 本人，也称之为认证器。这些记号的意义在后面分析的时候会得到进一步的阐述。同时，本文只关注认证成功的情况，在认证失败时，服务器只需向客户端发送认证失败消息即可。

Kerberos 4 协议中，认证协议的基本结构如图 1 所示。

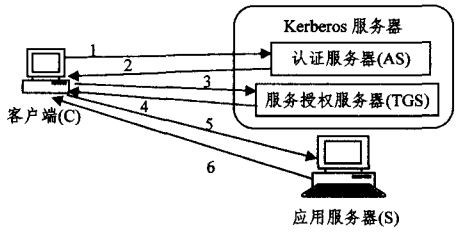


图 1　认证协议的基本结构

图 1 中发送的 6 条消息的内容和作用如下。

1)$C \rightarrow AS$：客户端向认证服务器发送认证请求 $KRB\_AS\_REQ$，其内容是：$\{c, tgs, TS_1\}$。其中，$c$ 表示客户标识，用于告知 *AS* 客户身份；$tgs$ 是 *TGS* 服务器标识，用于告知 *AS* 用户请求访问的 *TGS*；$TS_1$ 用于使 *AS* 能验证客户端时钟是否与 *AS* 时钟同步。

2)$AS \rightarrow C$：*AS* 向客户端发送认证回应 $KRB\_AS\_REP$，其内容是 $\{K_{c,tgs}, tgs, TS_2, Lifetime_2, Ticket_{tgs}\}K_c$。$K_{c,tgs}$ 是由 *AS* 随机产生的一个密钥；$TS_2$ 是票据发放的时间戳；$Ticket_{tgs}$ 是 *AS* 发给客户的用于访问 *TGS* 的票据授权票据，其定义为 $Ticket_{tgs} = \{K_{c,tgs}, c, AD_c, tgs, TS_2, Lifetime_2\}K_{tgs}$。

3)$C \rightarrow TGS$：当 $C$ 在第 2 步中获得 *AS* 认证，取得票据授权票据后，$C$ 向服务授权服务器(*TGS*)申请服务票据。发送的消息定义为 $KRB\_TGS\_REQ = \{s, Ticket_{tgs}, Authenticator_c\}$，其中 $s$ 是用户期望访问的服务器 $s$；$Authenticator_c = \{c, AD_c, TS_3\}K_{c,tgs}$ 是 $C$ 发送的一条认证消息(也称之为认证器)，用于向 *TGS* 声明 $C$ 确实已经得到会话密钥 $K_{c,tgs}$(因为 *TGS* 可以通过解密 $Ticket_{tgs}$ 而获得 $K_{c,tgs}$，通过匹配 $Ticket_{tgs}$ 中的客户标识和 $Authenticator_c$ 中的客户标识即可验证客户身份)。需要注意的是，与票据的可重用性不同，此处的认证信息仅能使用一次且生命期非常短。

4)$TGS \rightarrow C$：当 *TGS* 验明了客户 $C$ 确实具有票据授权票据后，授予客户服务授权票据。发送的消息定义为 $KRB\_TGS\_REP = \{K_{c,s}, s, TS_4, Ticket_s\}K_{c,tgs}$。该消息包含一个由 *TGS* 随机产生的密钥 $K_{c,s}$、服务器 $s$ 的标识、票据时间戳以及服务授权票据。服务授权票据定义为 $Ticket_s = \{K_{c,s}, c, AD_c, s, TS_4, Lifetime_4\}K_s$，它的结构与票据授权票据的结构相同。

5)$C \rightarrow S$：当客户拿到服务授权票据后，客户向应用服务器发送服务授权票据和认证单(即 $Ticket_S, Authenticator_c$)，请求允许使用服务。$Authenticator_c = \{c, AD_c, TS_5\}K_{c,s}$ 结构同前，只是时间戳更新了。

6)$S \rightarrow C$：$S$ 发送 $\{TS_5 + 1\}K_{c,s}$ 给 $C$，该消息是双向认证消息，让客户确认是他所期待的服务器收到了他的服务请求。

表 1 形式化地总结了 Kerberos 4 的消息交换过程。Kerberos 协议的思想在现实世界中也经常使用。例如，$A$ 想去乘坐 $B$ 航班(即需要申请使用某个资源)，则他首先需要去公安局获取身份证(即取得票据授权票据)；其次，他需要拿着身份证到航空公司买机票(即获取服务授权票据)；最后，在机场出示身份证和机票才能登机(即获取服务)。身份证、机票等都是有时间戳、有效期等时间信息的，在有效期内不需要再次认证，这也是 Kerberos 协议一次认证、多次登录情形的现实写照。

表 1　Kerberos 4 的消息交换

| | |
|---|---|
| 第一阶段　服务认证交换：获得票据授权票据 | |
| (1) C→AS： | $\{c, tgs, TS_1\}$ |
| (2) AS→C： | $\{K_{c,tgs}, tgs, TS_2, Lifetime_2, Ticket_{tgs}\}K_c$ |
| 第二阶段　服务授权票据交换：获取服务授权票据 | |
| (3) C→TGS： | $\{s, Ticket_{tgs}, Authenticator_c\}$ |
| (4) TGS→C： | $\{K_{c,s}, s, TS_4, Ticket_s\}K_{c,tgs}$ |
| 第三阶段　客户/服务器认证交换：获取服务 | |
| (5) C→S： | $\{Tickets, Authenticator_c\}$ |
| (6) S→C： | $\{TS_5 + 1\}K_{c,s}$ |

## 3　Kerberos 4 的缺陷

版本 4 是基于开发 Athena 计划而设计的。由于工程上的因素较多，在设计时以特定情况作为设计的出发点，因而导致了以下一些环境缺陷。

1)加密系统依赖性：该版本只采用数据加密标准(DES)来加密消息，限制了协议与其他加密算法的结合使用。

2)Internet 协议依赖性：该版本只使用 IP 地址，不支持其它地址使用。

3)消息字节顺序任意性：由消息发送者用标记说明规定消息的字节顺序，而不遵循已有的惯例。

4)票据的有效期有限性：该版本的有效期用 8 位表示，每一个单位代表 5min。因此最大有效期为 21h 多一点。

5)不具有向前认证性：应用服务器认证了客户后，并不能将该认证信息告知其他应用服务器或其它客户，从而导致其它相关操作不能进行。例如，客户端申请邮件服务器将某个文件作为附件发送，而需要利用客户端证书访问文件服务器得到该文件，在该版本中，邮件服务器不能请求文件服务器与之协作。

除此之外，版本 4 还存在一些技术缺陷[5]，主要有：

1)两次加密：在表 1 中，*AS* 和 *TGS* 提供给客户端的票据内容都先用目标服务器的密钥加密(密文就是票据)，然后又将票据和其它给客户的消息用客户端的密钥进行加密。显然票据的第二次加密并不是必需的，而且第二次加密消耗了服务器的处理时间。

2)PCBC 加密：版本 4 使用 DES 的非标准模式——传播密文块连接模式(PCBC)[6]，但此种模式已被证明易受交换密码块攻击[7]。

3)口令攻击：从 *AS* 发往客户端的消息是用基于客户端的密钥加密的，攻击者可以捕获消息，并通过尝试各种口令解密。如果解密成功，则攻击者即可得到客户端口令，进而使用该口令获得认证证书。

## 4 从 Kerberos 4 到 Kerberos 5

尽管 Kerberos 5 在很多地方都做了修改，但是一些基本的思想都没有变。本节主要分析版本 5 在版本 4 的基础上做了修改的地方。表 2 形式化描述了 Kerberos 5 的消息交换过程。

表 2　Kerberos 5 的消息交换

| | |
|---|---|
| 第一阶段　服务认证交换：获得票据授权票据 | |
| (1) C→AS: | $KOpts, c, tgs, Times, n_1$ |
| (2) AS→C: | $c, Ticket_{tgs}, \{K_{c,tgs}, Times, tgs, TFlags, n_1\}K_c$ |
| | $Ticket_{tgs}=\{TFlags, K_{c,tgs}, c, AD_c, Times\}K_{tgs}$ |
| 第二阶段　服务授权票据交换：获取服务授权票据 | |
| (3) C→TGS: | $Ticket_{tgs}, Authenticator_c, TOpts, s, Times, n_2$ |
| (4) TGS→C: | $c, Ticket_s, \{K_{c,s}, Times, s, n_2, SFlags\}K_{c,tgs}$ |
| | $Ticket_{tgs}=\{TFlags, K_{c,tgs}, c, AD_c, Times\}K_{tgs}$ |
| | $Ticket_s=\{SFlags, K_{c,s}c, AD_c, Times\}k_v$ |
| | $Authenticator_c=\{c, TS_1\}K_{c,tgs}$ |
| 第三阶段　客户/服务器认证交换：获取服务 | |
| (5) C→S: | $SOpts, Tickets, Authenticator_c$ |
| (6) S→C: | $\{TS_2, Subkey, Seq\}K_{c,s}$ |
| | $Ticket_s=\{SFlags, K_{c,s}, c, AD_c, Times\}K_s$ |
| | $Authenticator_c=\{c, TS_2, Subkey, Seq\}K_{c,s}$ |

现在分别介绍表 2 中元素的意义。

表中阴影斜体部分表示可选项，这一部分增加了协议的功能。通过设置相应的标志域，可以在协议中采用不同的加密技术，允许协议使用任何类型的网络地址，决定是否启动向前认证功能等。Times 用于客户端请求在票据中设置时间，包括请求票据的起始时间、请求票据的过期时间和请求过期时间更新的时间，从而允许票据拥有任意长度的有效期。$n_i$ ($i=1,2$)是新鲜值，用于确保应答是新的，且未被攻击者使用。Subkey 是客户端选择的一个子密钥，以保护某一特定的应用会话。如果此域被忽略，则使用票据中的会话密钥 $K_{c,v}$。Seq 是用于说明在此次会话中服务器向客户端发送消息的序列号，可选。同版本 4，消息(6)是在双向认证时服务器的应答。与版本 4 不同的是，在版本 5 中，考虑到攻击者不可能在不知道正确密钥的情况下创建消息(6)，因此不需要对时间戳加 1。

从形式上看，在第(2)、(4)两条消息上版本 5 与版本 4 的差别体现在编码上。在 Kerberos 认证服务器发送给客户端消息 KRB_AS_REP 的时候，因为票据授权票据已经用票据授权服务的口令加过密了，所以不再用用户的口令加密一次。这和票据授权服务的票据被用来获取服务授权票据的时候直接就在网络上传输的道理是一样的。这个类似的改变也应用到服务授权票据交换阶段；从票据授权服务返回的票据也不再用票据授权服务的口令来加密了，因为它所包含的票据已经被对应的服务的口令加过密了。

从两个版本的协议在相关实现上讲，因为验证器几分钟的有效期不足以防止攻击者进行重放(例如，攻击者用一个程序自动地截取票据和验证器并进行重放)，所以在 Kerberos 5 中，服务器用'重放缓冲区'保存了最近一次提交的验证器的信息，使得验证器真正地只能用一次。如果攻击者试图截取验证器并重用它，'重放缓冲区'会发现验证器已经被提交了。尽管口令攻击无法杜绝，但是版本 5 提供了一种称为预认证的机制，使得口令攻击更为困难。版本 5 还提供了精确的完整性检查机制，并用标准的密码字组链接(Cipher Block Chaining)模式加密。由于向前认证功能的存在，用户可以把他们的一部分认证权转给服务器，这样服务器就可以作为用户的代理。

在安全性保障方面，由于 Kerberos 协议主要关心的是认证性，因此它没有直接关注相关的认证和业务管理的安全功能。为了实现这些保障，Kerberos 5 在票据中提供了认证数据域以提供一些限制要求。同时，Kerberos 5 允许客户端与应用服务器协商使用子会话密钥，这体现在消息(5)、(6)的 Subkey 上。采用子会话密钥是因为在多次连接中，票据的会话密钥会被重用，所以版本 5 允许为特定会话生成新的子会话密钥，保护会话秘密。

**结束语**　尽管 Kerberos 认证服务目前还没有被广泛应用，但是它可以使服务器具备高效验证、相互验证、授权验证、简化信任管理、协同工作能力强等优点，因而也越来越受到各计算机厂商的青睐。例如，Windows® 2000 就使用公钥验证的扩展来实现 Kerberos 5。在不久的将来，该协议的使用范围会越来越广，且技术不断成熟完善。同时，我们也可以看出 Kerberos 认证协议比 Needham-Schroeder 公钥认证协议等经典的认证协议复杂，是一个大规模的复杂的安全协议。对复杂协议的形式化验证正是目前的热点问题。通过对该协议的分析和研究，可以使我们在复杂协议的建模、验证方面取得进一步的突破。

进一步的工作包括具体分析时间戳在进程代数描述中的表达、可选项的描述等；描述 Kerberos 的认证行为；用进程代数方法形式化描述 Kerberos 协议的 3 个阶段；用形式化方法验证 Kerberos 协议的正确性等。

## 参 考 文 献


[1] Butle F, Cervesato I, Jaggard A D, et al. A Formal Analysis of Some Properties of Kerberos 5 Using MSR// the Fifteenth IEEE Computer Security Foundations Workshop. 2002

[2] Stallings W. Cryptography and Network Security Principles and Practices. Third Edition. Pearson Education

[3] Steiner J, Neuman C, Schiller J. Kerberos: An Authentication Service for Open Networked Systems// Proceedings of the Winter 1988 USENIX Conference. February 1988

[4] Bryant W. Designing an Authentication System: A Dialogue in Four Scenes. Project Athena Document. February 1988


(下转第 128 页)

$dex_{A_table}$代表$A_table$的索引个数。

3.2.3 性能分析

通常分类器的分类字段主要集中于IP帧相关字段上，经过IP分类后规则的个数大大缩小，因此BH算法的时间和空间复杂度主要集中在IP分类结构之上。假设有$n$个规则，$m$阶B树共有$p$个结点，$k$个端点(关键码)，其深度为$h$，则有$k\leqslant 2n$。根据B树的定义，可知：

$$h\leqslant \log_{\lceil m/2\rceil}\left(\frac{k+1}{2}\right)+1\leqslant \log_{\lceil m/2\rceil}\left(n+\frac{1}{2}\right)+1 \qquad (4)$$

IP分类有5个字段，需要查找5棵B树，然后进行2次Hash表查找，则IP分类的时间复杂度近似为：

$$O(5(\log_{\lceil m/2\rceil}\left(n+\frac{1}{2}\right)+1)+2)$$

即$\log_{\lceil m/2\rceil}(n)$。当$m$取值越大时，时间复杂度越低。

其中算法在更新时可能会修改多个结点，并在最差情况下会分裂或者合并结点。由于根结点至少有一个端点，其他各非失败结点至少有$\lceil m/2\rceil-1$个端点，则B树至少有$1+(\lceil m/2\rceil-1)(p-1)$个端点，则平均分裂结点数$S$为：

$$S=\frac{\text{分类结点总次数}}{k}\leqslant\frac{p-2}{1+(\lceil m/2\rceil-1)(p-1)}<\frac{1}{\lceil m/2\rceil-1} \qquad (5)$$

当有$n$个规则时，在最坏的情况下，每个B树端点最多为$2n$，表A_table最多有$2n\times 2n$个表项，表P_table最多有$2n\times 2n\times 2n$个，表AP_table则有$O(n^5)$。与RC-FST性能的对比如表4所列，通过改变$m$的取值可以对算法的时间空间性能进行调整。

表4 性能比较($d=5$)

| | 时间复杂度 | 空间复杂度 |
|---|---|---|
| RC-FST | log(n) | $n^5$ |
| BH | $\log_{\lceil m/2\rceil}(n)$ | $n^5$ |

**结束语** 本文通过分析和对比常用报文分类算法的性能，针对HFC宽带网络接入技术的CableModem系统QoS体系提出了一种BH分类算法。分析表明，该算法具有时间和空间复杂度小的优点，满足实际嵌入式开发的要求，并且通过控制B树的参数，能对算法的时间空间性能进行调整，从而满足不同应用的需求。本算法对于宽带无线接入网802.16协议的QoS分类器、无线局域网802.11e、路由器等网络协议QoS分类算法的软/硬件实现具有参考借鉴意义。本研究的下一步工作将对BH算法结构进行优化，进一步提高其查找和更新性能。

## 参考文献


[1] 林锐，单志广，任丰原，等. 计算机网络的服务质量. 北京：清华大学出版社，2004

[2] Yu Lei, Deng Ya-ping, Wang Jiang-bo, et al. A Novel IP Packet Classification Algorithm Based on Hierarchical Intelligent Cuttings// The 6th International Conference on ITS Telecommunication Proceedings. 2006: 1033-1036

[3] 田珂，朱清新，向培素. 一种改进的多维高速报文分类算法. 计算机应用研究，2007(2): 27-32

[4] Hsu Chia-ren, Chen Chien, Lin Chun-Yuan. Fast Packet Classification Using Bit Compression. IEEE Globecom, 2005: 739-743

[5] Abdelghani M, Sezer S, Garcia E, et al. Packet Classification Using Adaptive Rules Cutting// Proceedings of the Advanced Industrial Conference on Telecommunications. 2005

[6] 汪伟，孙翌. 报文分类算法的设计与实现. 上海电力学院学报，2006, 22(1): 63-70

[7] Yu Lei, Deng Ya-ping, Wang Jiang-bo, et al. A Novel IP Packet Classification Algorithm Based on CrossProduct and Hash Tree // The 6th International Conference on ITS Telecommunication Proceedings. 2006: 1037-1040

[8] Tan Xing-ye, Zhan Yong, Leit Zhen-ming. A New Fast Packet Classification Algorithm: RC-FST. IEEE, 2005: 462-466

[9] Zheng Kai, Liang Zhiyong, Ge Yi. Parallel Packet Classification via Policy Table Pre-Partitioning. IEEE Globecom, 2005: 73-78

[10] Sun Xuehong, Sahni S K, Zhao Yiqiang Q. Packet Classification Consuming Small Amount of Memory. IEEE/ACM Transacton on Networking, 2005, 13(5): 1135-1145

[11] Xu Zhen, Sun Jun, Zhang Jun. A Novel Hash-based Packet Classification Algorithm. ICICS, 2005: 1054-1059

[12] Taylor D E, Turner J S. ClassBench: A Packet Classification Benchmark. IEEE/ACM Transactions on Networking, 2007, 15(3): 499-511

[13] Jelassi O, Paul O. Markers - based Space Decomposition Algorithm: A new algorithm for multi-fields packet classification. IEEE, 2006: 43-47

[14] Lu Haibin, Sahni S. O(logW) Multidimensional Packet Classification. IEEE/ACM Transactions on Networking, 2007, 15(2): 432-442

[15] Cable Television Laboratories, Inc. Data-Over-Cable Service Interface Specifications DOCSIS 1.1 Radio Frequency Interface Specification CM-SP-RFIv1.1-C01-050907[S]. 2005

[16] Nu K, Wu J P, Yu Z C, et al. A Non-Collision Hash Trie-ree Based Fast IP Classification Algorithm. J. Computer Sci. & Technol., 2002, 17(2): 219-226

[17] 刘惠义，董志勇，秦益，等. 基于无冲突哈希Trie树的IP分类算法的研究. 计算机与现代化，2004(5)


(上接第121页)


[5] Bellovin S, Merritt M. Limitations of the Kerberos Authentication System. Computer Communications Review, October 1990

[6] Meyer C, Matyas S. Cryptography: A New Dimension in Computer Data Security. New York: Wiley, 1982

[7] Kohl J. The Use of Encryption in Kerberos for Network Authentication// Proceedings, Crypto'89. New York: Springer-Verlag, 1989

[8] Kohl J, Neuman B, Ts'o T. The Evolution of the Kerberos Authentication Service. IEEE Computer Society Press, 1994

[9] Kohl J, Neuman C. The Kerberos Network Authentication Service (V5). Digital Equipment Corp, 1993

[10] Stallings W. 密码编码学与网络安全——原理与实践. 第3版. 刘玉珍，等译. 电子工业出版社，2004

[11] Moron. Guide to Kerberos. 1996. http://www.isi.edu/gost/brian/security/kerberos.html

[12] Butle F, Cervesato I, Jaggard A D, et al. Formal Analysis of Kerberos 5. Theoretical Computer Science, 2006, 367: 57-87

[13] Cervesato I, Jaggard A D, Tsay Joe-Kai, et al. Breaking and Fixing Public-Key Kerberos. Information & Computation, 2007