国家环境信息与统计能力建设项目

环境数据加密技术规定

Encryption technology criterion for environmental data

（征求意见稿）

《环境数据加密技术规定》编制组

2010 年 9 月

目　　次

环境数据加密技术规定

1 适用范围

本技术规定确定了国家环境信息与统计能力建设项目中非涉密数据的机密性、完整性、抗抵赖性和认证的要求。

本技术规定适用于国家环境信息与统计能力建设项目中非涉密数据关于机密性、完整性、抗抵赖性和认证的实现过程。

2 术语和定义

GB/T 9387.2-1995、GB/T 18794.1-2002、GB/T 18794.5-2003、GB/T 17964-2008、GB/T 18794.6-2003 、 GB/T 18794.4-2003 、 GB/T 17903.1-2008 、 GB/T 17902.1-1999 、 GB/T 18794.2-2002、GB/T 15843.1-2008、GB/T 20273-2006 中确立的术语和定义适用于本技术规定。

3 总则

3.1 环境数据保护的总体目标

根据 GB/T 17859-1999（4 等级划分准则） 、GB/T 22239-2008（7 第三级基本要求）和 GB/T 22240-2008（5 定级方法）的规定，国家环境信息与统计能力建设项目中所属信息系统应达到第三级要求，信息系统中非涉密敏感数据应达到第三级所要求的安全标准，实现过程中应使用国家安全保密部门批准使用的有关算法。

3.2 环境数据安全的基本要求

从网络环境、环境数据的安全威胁和安全需求，规定保障环境数据安全的基本要求，包括机密性、完整性、抗抵赖、认证、分级、分层安全等级保护、安全与效率兼顾等。

3.3 环境数据安全级别分类

见“环境信息安全技术规范”（在编）。

3.4 用户安全级别划分

见“环境信息安全技术规范”（在编）。

4 数据机密性

4.1 要求

a) 应采用加密或其它有效措施实现系统管理数据、鉴别信息和重要业务数据传输保密性；

b) 应采用加密或其它保护措施实现系统管理数据、鉴别信息和重要业务数据存储保密性；

c) 应对通信过程中的整个报文或会话过程进行加密。

4.2 应用范围

通过机密性机制防止数据在存储和传送过程中被泄露及非授权使用。

4.3 实现机密性的方法

信息在通信和存储过程中是用数据(项)的形式来表示，数据可能通过理解其含义，通过研究其上下文关系，通过观察数据的动态变化或通过研究数据的相关属性作推导这些方式被泄露，可以用下面的方法来实现数据的机密性：

a） 保护数据存在或数据特性的相关属性（如长度、创建日期）；

b） 阻止对数据的访问；

c） 保护数据的语义（如加密、编码规则等安全技术）。

实现机密性的过程可由隐藏和显现两个操作来完成。

（1）隐藏操作：将信息从某环境 A 移到 A 与另一个环境 C 交迭的区域 B；

（2）显现操作：隐藏操作的逆操作。

详细操作过程见 GB/T 18794.5-2003 附录 B。

数据库管理系统应确保数据库中存储和传输的用户数据的机密性，按照 GB/T 20271-2006 中 6.3.3.8 的要求，实现数据库管理系统的用户数据机密性保护功能。

4.4 机密性机制类型

按信息保护类型，机密性服务可以分成三类：

a) 保护数据语义；

b) 保护数据语义和相关属性；

c) 保护数据语义及其属性，以及可从该数据导出的任何信息。

按威胁的种类，机密性服务可以分成防止外部威胁和防止内部威胁两类：

（1）防止外部威胁

这类服务假设能够合法访问信息的访问者不会把信息泄露给未授权者。这类服务不保护泄露给已授权方的信息，并且在它们拥有以前已被保护的信息时，也不限制这些授权方的行为。例如，在 A 中的敏感文件通过加密受到保护，但拥有解密密钥的进程可以读取被保护的文件，然后又对该读取出来的不受保护的文件进行写操作。

（2）防止内部威胁

这类服务假设已授权者会从事损坏被保护信息的机密性的活动。

使用防止内部威胁服务时不允许隐蔽通道，或者将信息传输率限制在一个可接受的水平。另外还必须禁止非授权的推理，这种推理可以来自合法信息通道的意外使用。

隐蔽通道的描述见 GB/T 18794.5-2003 附录 D。

机密性机制通过访问控制、保护信息项的表示规则和表示内容来达到机密性目的。

可以通过下面两类机密性机制类型防止未授权信息的泄露：

（1）基于访问控制防止对数据的访问

通过访问控制机制对抗由涉及计算机或通信系统的非授权所造成的威胁，使只有授权实体才能访问数据，此类型实现机密性的机制见 GB/T 18794.3-2003。

（2）基于映射技术保护数据

任何实体可以访问数据，而只有授权拥有映射技术重要信息的实体才能解读数据语义。

4.5 机密性机制

(1) 通过对称加密技术提供机密性

此机制中，通过使用相同的密钥来加密（隐藏操作）和解密数据（显现操作），防止在传输或存储过程中数据语义被泄露；对称加密方法可以用分组密码和序列密码。

1）分组密码算法（块密钥算法）：

将信息作为数据分组来加密或解密，信息被分成一系列连续排列的信息分组，一次处理一个分组。

分组密码的运行模式：

电码本（ECB）模式

密码分组链接（CBC）模式

密码反馈（CFB）模式

输出反馈（OFB）模式

计数器（CTR）模式

分组链接（BC）模式

带非线性函数的输出反馈（OFBNLF）模式

运行模式的详细描述见 GB/T 17964-2008 分组密码算法的工作模式。

2）序列密码算法（流密码算法）：

加密或解密每次只处理一个字符或一个比特。每次将一个明文字符与密钥流作用进行加密，产生一个密文字符，解密时用同步产生的同样的密钥流实现；密钥流可以通过多种方法来产生：可以是一个预定的值，也可以用某种算法每次只产生一个值，这种值可能依赖于明文或密文字符，也可能依赖于以前的密钥值。

(2) 通过非对称加密技术提供机密性（公开密钥密码算法）：

此机制是将传统密码的密钥分为加密密钥 K_e 和解密密钥 K_d，用加密密钥（公开密钥）K_e 控制加密（隐藏操作），用解密密钥（私有密钥）K_d 控制解密（显现操作），而且由计算复杂性确保由加密密钥 K_e 在计算上不能推出解密密钥 K_d，K_e 公开，只对 K_d 保密。非对称加密方法可以克服传统密码在密钥分配上的困难。

使用非对称加密技术的机制其目的也是防止在传输或存储过程中数据语义被泄露。对称和非对称两种加密体系的主要差异，对称加密体系能实现加密操作的实体也能实现解密操作，能实现解密操作的实体也能实现加密操作，而非对称加密体系中大部分实体都能实现加密操作，而解密操作只有拥有私有密钥的实体才能实现。

(3) 通过上下文位置提供机密性

如果数据可以通过大量不同的上下文确定，可以通过禁止对数据访问来阻止这种行为。上下文被改变之前如果不可能检测到所有可能的上下文，则可阻止通过大量不同的上下文找到数据，达到机密性目的。

这种机制下包括：提供大量传输信息的物理的或虚拟的通道（如“展频”使用大量无线电频率中的一个）、提供大量存储数据的地方（如在磁盘上的地址）、通过隐藏在主信道中的隐秘辅助通信通道传输信息（隐写术）。

此形式的机密性假设非授权接受者不能得到识别当前正确上下文所需的信息，因此这一信息本身必须受到机密性服务的保护。

（4）通过访问控制提供机密性

此机制可以通过物理介质保护和路由选择控制获得机密性。物理介质保护下的数据只能通过一些特殊的机制才能检测到，机密性是通过确保只有授权的实体才能利用这些机制而实现；路由选择控制中只使用可信和安全的设施来路由数据，这一机制的机密性是通过防止由被传输的数据项所表示的内容未授权泄露来得到。

(5)通过数据填充提供机密性

此机制下增加了数据项的大小，填充后数据项的大小与原数据项的大小之间不存在关联；其目的是防止泄露以数据项大小所表示的信息；通常是在数据项的开始或末端填充随机数据。填充的方法或规则要使得填充数据能被授权的实体识别，而对于未授权的实体不能识别，为此，数据填充可以和密码技术结合使用。

(6)通过虚假事件提供机密性

该机制的目的是防止通过事件的发生率作推理。此机制产生只有授权方才能识别的虚假事件，可以用来对抗隐蔽通道攻击。

(7)通过保护 PDU 头提供机密性

此机制的目的是在通信中防止基于 PDU（协议数据单元）头的推理。

(8)通过时间变化字段提供机密性

此机制是把时间变化字段和被保护的数据结合起来，让攻击者不能判断数据表示的变化

是由数据的改变还是由时间变化字段中的变化而引起的，和加密机制结合使用可以防止基于数据项的动态变化进行的推理。此机制能与填充和分段机制结合使用，用来隐藏被保护数据大小的变化。

5 数据完整性

5.1 要求

a) 应能够检测到系统管理数据、鉴别信息和重要业务数据在传输过程中完整性受到破坏，并在检测到完整性错误时采取必要的恢复措施；

b) 应能够检测到系统管理数据、鉴别信息和重要业务数据在存储过程中完整性受到破坏，并在检测到完整性错误时采取必要的恢复措施；

5.2 应用范围

防止或检测非法的或未授权的数据修改，包括未授权的数据创建和删除，保护数据及其相关属性的完整性。

5.3 实现完整性的方法

完整性服务的目的是提供对数据及其相关属性的完整性保护，避免遭受下列不同方式的危害：

a) 阻止或检测未授权的数据修改

b) 阻止或检测未授权的数据创建

c) 阻止或检测未授权的数据删除

d) 阻止或检测未授权的数据插入

e) 阻止或检测未授权的数据重放

实现完整性的过程可由屏蔽、证实和去屏蔽三个操作来完成。

（1）屏蔽：由数据生成完整性保护的数据；

（2）证实：对受完整性保护数据进行检查，以便探测完整性是否失败；

（3）去屏蔽：由受完整性保护数据重新生成数据。

这些操作可以不使用密码技术，如果使用密码技术，则不需要对数据进行变换。例如，屏蔽操作可以通过给数据项添加封印或者数字签名来完成，在证实成功之后，去屏蔽操作可以通过去除封印/数字签名来完成。

数字签名是用于证明当事人的身份和数据真实性的一种信息。数字签名的目的是提供一种手段，使得一个实体把他的身份与某个信息捆绑在一起。一个信息的数字签名实际上是一个数，它依赖于签名者的密钥和被签名的消息。数字签名主要由签名算法（signing algorithm）和验证算法（verification algorithm）两部分组成。

数字签名有两类机制：

（1）带附录的数字签名

若验证进程需要消息作为输入部分，这种机制称为“带附录的数字签名”。在计算附录时使用了散列函数。

标准 GB/T 17902 规定了任意长度消息的带附录数字签名机制，这些机制适用于提供实体认证、数据原发认证、抗抵赖和数据完整性的方案。GB/T 17902.1 中描述了带附录的数字签名的基本原则和要求，GB/T 17902.2 中规定了任意长度消息的带附录的基于身份的数字签名机制的签名和验证过程的总的结构和基本过程，GB/T 17902.3 中对带附录的基于证书的数字签名机制作了一般描述并且提供了使用任意长度的基于证书机制的带附录的各种常规数字签名机制。

（2）带消息恢复的数字签名

若验证进程给出消息及其特定冗余（有时也称作消息影子），这种机制称为“带消息恢复

的签名机制”。

GB/T 15851-1995 中规定了有限长消息的数字签名方案，此方案的验证进程只需要尽量少的资源，不涉及到使用散列函数，可以避免对这种一般算法的已知攻击。

散列函数是验证数据完整性的重要工具。散列函数又称为哈希函数，Hash 函数或杂凑函数，它把任意长度的消息变换为固定长度比特串的散列值，该散列值被称为消息摘要(message digest)。消息摘要就象数字指纹，即用一小段数据来识别长的消息，而哈希函数就好象压缩函数一样把长消息变短了，其目的是为了产生消息的“数字指纹”，而不要求解压恢复原消息。消息在插入、篡改、重排之后，改变后的消息的“数字指纹”将发生改变。所以，利用散列函数可以提供数据完整性服务。GB/T 18238 中提供了适用于各种安全技术的各种散列函数，它们可以用于提供认证、完整性和抗抵赖服务。

消息认证码也是一种可以用于数据完整性检验的机制，检验数据是否被非授权地改变，同时也可以用于消息认证，保证消息来源的合法性，在采用分组密码的消息认证码机制中，数据完整性和消息认证的强度依赖于密钥的长度及其保密性、分组密码的算法强度以及分组长度、消息认证码的长度和具体的消息认证码算法。GB/T 15852 中提供了一些可以使用的消息认证码算法。

对数据库管理系统中处理的用户数据（如进程间的通信），应提供保证数据完整性的功能，按照 GB/T 20271-2006 中 6.3.3.7 的要求，实现实体完整性、参照完整性和用户定义完整性。

5.4 数据完整性机制类型

可以根据要防止的危害类型、所支持的保护类型和是否包含恢复机制对完整性服务分类：

根据要防范的类型可以分为：

a) 阻止或检测未授权的数据修改

b) 阻止或检测未授权的数据创建

c) 阻止或检测未授权的数据删除

d) 阻止或检测未授权的数据插入

e) 阻止或检测未授权的数据重放

根据所支持的保护类型可分为：

（1）完整性损害的预防；

（2）完整性损害的检测。

根据是否包含恢复机制可分为：

（1）在具有恢复机制时，如果证实操作指示数据发生了改变，则去屏蔽操作能够恢复原始数据；

（2）在不具有恢复机制时，即使证实操作指出数据已发生改变，去屏蔽操作也无法恢复原始数据。

完整性机制可分为两大类：预防机制和检测机制。

预防机制通过阻止所有未经授权的数据修改或所有使用未经授权的方法来修改数据，以确保数据的完整性。前者发生在当用户企图改写其未经授权修改的数据时，后者发生在已授权对数据做特定修改的用户试图使用其它手段来修改数据时。

检测机制只是检测完整性是否遭到侵害，并不阻止对完整性的破坏，如果完整性被破坏，则报告数据的完整性已不再可信。

（1）阻止访问介质的机制

这类机制包括：物理隔离、无噪音的信道，路由选择控制，访问控制。

（2）检测对数据或数据项序列未授权修改的机制，包括未授权的数据创建、删除和恢复。

这类机制包括：封印、数字签名、数据复制、数字指纹、消息序列号

5.5 数据完整性机制

a) 通过对称密码技术提供完整性

此类机制对应于封印，消息发送者将消息编码，生成一个与之相关的值或一个密码校验值，并将其与消息封装在一起，接受者收到消息后利用共享密钥重新计算密码校验值并与收到的密码校验值做比较，如果一致则认为该消息未被修改。

b) 通过非对称密码技术提供完整性

此类机制对应数字签名，消息发送者通过私有密钥生成数字签名，接受者用公开密钥验证（证实），如果验证失败表示数据已经被更改。

c) 通过对冗余数据加密提供完整性

数据在加密前可以使用数字指纹、纠错编码、检错编码、散列函数这些方式作扩充，使数据产生某种冗余，通过加密对冗余数据的完整性作支持。

d) 通过数据上下文提供完整性

此类机制提供数据删除的完整性检测。通过在某变化程度内的特定时间和/或位置提供数据实现屏蔽，而通过预期数据是否在给定的时间和/或位置出现来验证，如果未出现则认为完整性受到破坏。

为防止数据被替换，数据必须以不可假冒的方式被识别，使用该机制时必须与其它完整性机制相结合。

e) 通过复制数据提供完整性

这类完整性机制基于在多个不同的存储区域或不同的时间对数据做备份，可以通过对每个备份副本作比对，如果之间有差异，则认为完整性收到破坏；只要可能的攻击者不能同时破坏有限某个数量的副本，则当检测到有攻击时，可以从真正的副本重新恢复数据。

f) 通过检测和确认提供完整性

这些机制使用一个完整性检测和一个修改完整性保护机制，通过反复执行同一个操作，直到收到一个肯定的确认实现屏蔽，去屏蔽的获得是反复的受到完整性检测机制的验证实过程的控制，证实成功后给执行屏蔽操作的实体一个肯定的确认，当证实失败时能给屏蔽操作一个否定的确认。

这类机制假定屏蔽和证实/去屏蔽操作在相同的时间段结束，通常不适合数据存储和恢复。

g) 通过预防提供完整性

阻止对数据存储或传输介质的物理访问及通过访问控制可以提供完整性。

访问控制的描述见 GB/T 18794.3-2003。

5.6 数据完整性恢复

可以通过简单的标准通信错误恢复机制来支持完整性恢复机制，一般在检测到完整性破坏发生和重新发送所有数据之前要重新同步到检测位置。

6 数据抗抵赖性

6.1 要求

a) 应具有在请求的情况下为数据原发者或接收者提供数据原发证据的功能；

b) 应具有在请求的情况下为数据原发者或接受者提供数据接收证据的功能。

6.2 应用范围

生成、收集、维护已声明的事件或动作的证据,并使该证据可得并且确认该证据,以此来解决关于某事件或动作发生或未发生而引起的争议。

6.3 实现抗抵赖性的方法

抵赖类型可以分为两种，一种是对起源的抵赖，存在是否某方产生了关于某个特定数据项的纠纷和关于产生时间的纠纷；另一种是对传递的抵赖，存在是否某个特定数据项传递给

了某方的纠纷或关于传递发生时间的纠纷。可以通过使用数字签名、加密、公证和数据完整性等机制提供抗抵赖服务。在通信中，起源的抗抵赖服务和大多数实现抗抵赖服务的机制也适用于数据库存储。

数字信封是公钥密码体制在实际中的一个应用，是用加密技术来保证只有规定的特定收信人才能阅读通信的内容。在数字信封中，信息发送方采用对称密钥来加密信息内容，然后将此对称密钥用接收方的公开密钥来加密（这部分称数字信封）之后，将它和加密后的信息一起发送给接收方，接收方先用相应的私有密钥打开数字信封，得到对称密钥，然后使用对称密钥解开加密信息。这种技术的安全性相当高。数字信封主要包括数字信封打包和数字信封拆解，数字信封打包是使用对方的公钥将加密密钥进行加密的过程，只有对方的私钥才能将加密后的数据(通信密钥)还原；数字信封拆解是使用私钥将加密过的数据解密的过程。数字信封的功能类似于普通信封，普通信封在法律的约束下保证只有收信人才能阅读信的内容；数字信封则采用密码技术保证了只有规定的接收人才能阅读信息的内容。数字信封中采用了对称密码体制和公钥密码体制。信息发送者首先利用随机产生的对称密码加密信息，再利用接收方的公钥加密对称密码，被公钥加密后的对称密码被称之为数字信封。在传递信息时，信息接收方若要解密信息，必须先用自己的私钥解密数字信封，得到对称密码，才能利用对称密码解密所得到的信息。数字信封可以保障数据传输的真实性和完整性。

可信第三方（TTP）

根据使用的抗抵赖机制和策略，抗抵赖服务通常需要可信第三方的参与，参与的可信第三方可以是一个或多个，一个可信第三方可充当一种或多种角色，如公证人、时间戳、监控、密钥证书、签名生成、签名验证、安全信封生成、安全信封验证、权标生成或交付等角色，可以要求可信第三方记录和/或收集证据，也可以要求可信第三方证明证据的有效性。

为保证密钥的真实性，使用非对称密码技术时需要一个离线的可信第三方，可信第三方也可以是 TTP 链中的一部分，只要它满足抗抵赖策略的要求，同意履行抗抵赖服务的职责；使用对称密码技术时，要求一个在线的可信第三方参与，以生成和验证安全信封。抗抵赖策略可要求由可信第三方生成部分或全部证据。

离线可信第三方是指一个可信第三方支持抗抵赖但没有主动地介入在服务的每个使用中；如果可信第三方主动介入证据的生成或验证则称为在线可信第三方。

抗抵赖还可能要求：

（1）时间戳服务机制：由可信时间戳机构提供的可信时间戳；

（2）公证服务机制：由公证机构或公证人提供证据，以验证有关实体的性质和被存储或传输的数据性质，并且将现有权标的生命期延长到期满和被撤销以后，确保所执行的事件或动作；

（3）证据记录服务机制：保存操作的记录，在以后发生争议时恢复数据，提供关于实体的性质和存储或传输的数据性质的证据。

可信第三方可以不同程度地参与在抗抵赖过程中。当交换证据时，双方须知道或同意适用于证据的抗抵赖策略。

抗抵赖的四个独立阶段：

（1）证据生成过程：证据是用于解决争议的信息。

证据生成过程中涉及 3 个实体：

1）需要得到证据的证据请求者；

2）完成某动作或某事件所涉及的证据主体(被证据证实卷入在一个事件或动作中的实体称为证据实体)；

3）生成证据的证据生成者。

在证据生成过程中，证据由证据生成者代表证据主体、可信第三方生成，或者应证据请

求者的请求而生成。如果证据主体和证据请求者都不能直接提供证据，则由证据生成者生成证据并将证据送给证据请求者。生成的证据还可以提供给其它实体。

（2）证据传送、存储和检索过程：此过程中，证据在各个需要使用的实体之间传输或在数据存储区之间传输，这一阶段的活动并非在抗抵赖服务的所有情况下都进行。本阶段的活动可由可信第三方执行， 作为交付机构时，可信第三方处于联线状态，完成提交抗抵赖和传输抗抵赖，作为证据记录保管机构时，可信第三方记录数据，可供证据用户或仲裁者以后进行检索。

（3）证据验证过程：核对被传递的或被可信地保存着的证据。

证据验证过程中涉及两个实体：

1)希望验证证据但不能直接验证证据的证据用户；

2)应证据用户的要求，可以验证证据的证据验证者。

在证据验证过程中，证据用户希望验证证据的正确性，如果证据用户不能直接验证证据的正确性，则由证据验证者验证。作为证据的验证机构，可信第三方是受证据用户信任的在线机构，用于验证抗抵赖权标提供的各种抗抵赖信息，并非在所有情况下都必须要可信第三方的参与，抗抵赖权标的验证方法取决于所使用的技术。

安全信封只能由持有用于生成安全信封的秘密密钥的可信第三方进行验证；

持有签名者的公开密钥的任何实体都可以验证数字签名。向验证者提供公开密钥证书的方式依赖于生成数字签名的签名方案的类型，基于证书的签名使用签名者的公开密钥进行验证，该公开密钥可以从认证机构（CA）颁发的公开密钥证书中取得；对基于身份的签名，持有签名实体的标识数据和可信机构（TA）的公开系统参数的任何实体都可以进行验证。签名者的基于身份的私有密钥由 TA 提供。对数字签名来说，必须对一个公开密钥证书链或身份标识符链顺序进行验证才能得到必要的保证。

（4）解决纠纷过程：判决者解决纠纷时从纠纷方和/或可信第三方（仲裁者）收集证据，并且核实证据，解决纠纷。

6.4 抗抵赖机制类型

按抗抵赖机制所采用的技术，可以分为采用对称技术的机制和采用非对称技术的机制，GB/T 17903.2-2008 和 GB/T 17903.3-2008 中分别描述了基于对称和非对称技术的抗抵赖服务机制以及一些与通信有关的特定机制，这些机制可以用于提供原发抗抵赖、交付抗抵赖、提交抗抵赖和传输抗抵赖。

通用抗抵赖模型可以提供六种基本的抗抵赖服务：创建抗抵赖、发送抗抵赖、接收抗抵赖、认知抗抵赖、提交抗抵赖和传输抗抵赖，通过这些基本服务可以组合成其他需要的抗抵赖服务。特定的抗抵赖机制类型可以根据其抗抵赖服务的不同需要，由实现上面六种基本服务的体制结合而成，如：原发抗抵赖机制、交付抗抵赖机制、提交抗抵赖机制及传输抗抵赖机制。

6.5 抗抵赖机制

根据证据生成所使用的方法不同，证据可以分成两类：安全信封和数字签名，安全信封通过对称密码技术生成，数字签名通过非对称密码技术生成。使用数字签名、加密、公证和数据完整性这些机制来提供抗抵赖服务可以增加时间戳等其它服务的支持，这些机制和服务的适当组合能满足特定应用中抗抵赖服务的安全需要。

（1）使用 TTP 安全权标（安全信封）的抗抵赖

此方式下抗抵赖证据由一个安全权标构成，用只有 TTP 知道的秘密密钥封印。TTP 担任证据生成者和证据验证者的角色，证据生成请求者将数据或数据的数字指纹以及生成安全权标的请求传输给 TTP，该请求必须是受到完整性保护（例如使用封印）或是受到机密性保护（例如使用加密），在证据生成请求者的要求下 TTP 产生安全权标，并且以后能应证据使用者或判

决者的要求对其进行验证。

（2）使用安全权标和防篡改模块的抗抵赖

此方式下抗抵赖证据由一个安全权标构成，用一个秘密密钥封印，该秘密密钥存放在防篡改模块中，防篡改模块由证据生成者、证据验证者和判决者所拥有，证据生成者模块可以用秘密密钥来创建一个封印的权标，而证据验证者和判决者拥有的模块只能验证权标。

防篡改模块限制那些可用秘密密钥执行的操作，并防止密钥的值被暴露在模块外面，所有的相关实体必须相信秘密密钥已被正确安装在防篡改密码模块中，同一个秘密密钥只能被一个实体用于证据生成，而其它实体只能将其用于证据验证。

如果出现纠纷，证据使用者向判决者出示封印的权标，并证明该权标必然是使用证据生成者模块创建的，因为拥有同样密钥的其它模块不具有生成安全权标的能力。

（3）使用数字签名的抗抵赖

此方式下抗抵赖证据由一个数字化签名的数据结构构成。签名密钥用于签名生成，验证密钥用于签名验证。数字签名可由证据主体生成者或者由签名生成角色中的 TTP 生成，由证据主体生成的数字签名称作直接数字签名，由代表证据主体的 TTP 生成的数字签名机制称作中介数字签名。

根据安全策略，可能需要时间信息。这可以包含在有实体提供的和/或由作为时间戳机构的 TTP 提供的数字签名中。当时间信息不是由 TTP 提供时，其它实体没有必要相信它。如果判决者需要时间戳和/或上下文信息来解决纠纷时，此信息必须从可信任源（例如 TTP）处获取。

为检验证据，证据验证者和判决者必须能够获得验证密钥，如果不能保证判决者将通过其它方法知悉证据生成者的公开密钥，则证据还必须为此密钥而包含一个安全证书。当用作验证签名的证书已被撤销后，单独用数字签名是不足以解决纠纷的，为解决此类纠纷，还必须另外向判决者提供有关该证书撤销的证据（例如证书撤销列表-CRL），表明证书在生成数字签名时是任然有效的，然而，当私有密钥的拥有者自愿使用不正确时间时，或攻击者毁坏了用于生成签名的私有密钥时，不允许将本方案用于解决纠纷。为解决此类纠纷，必须另外使用一个可信时间基准，或使用来自其时间戳角色中的 TTP 的后随签名。

证据验证者可使用目录服务来获得验证过程所需的信息（例如安全证书）。证据验证者必须获得证据生成者的公开密钥。这个密钥可以包含于存储在目录中的安全证书里。可能需要不只一个证书。为确保一个证书是合法的，还有必要请求可用的撤销证书列表。这对出现在证书路径中的每个证书机构来说都是必须的（见 GB/T 16264.8--1996）。证据使用者可以寻求充当数字签名角色中的 TTP 的帮助来证实数字签名。在这个角色中，TTP 验证原始消息（或者消息的数字指纹，如果使用了的话）于数字签名之间的关系。在这种情况下，TTP 的作用是降低证据使用者签名验证过程的复杂性，并为了优先响应以后的验证请求而维护先前验证请求的结果，为达到此目的，TTP 可能需要与目录进行一些交互，要求充当签名验证角色中的 TTP 至少持有一个证书机构的公开密钥。TTP 也要考虑不同证书机构之间存在的信任关系。

（4）使用时间戳的抗抵赖

时间戳可以用来证实消息是在签名密钥泄露之前签署的，因此该消息不是伪造的。当需要可信时间基准，而产生数字签名或安全权标的实体提供的始终又不可信时，有必要依赖可信第三方来提供时间戳。在时间戳角色中，可信第三方将提供数字签名或安全权标以证实收到请求的时间。证据生成者、抗抵赖服务请求者、证据使用者或证据验证者都可以请求时间戳。

时间戳将时间、日期以及一个封印或数字签名添加到数据上以证明数据的真实性。时间戳不需要对提出时间戳请求的实体进行鉴别。证据验证者必须确定时间戳是否像安全策略所命令的那样在可接受范围内。

时间戳可与签名生成或权标生成结合。如果生成数字签名的实体包含可靠并且可信的时钟，可不需要后随签名。

（5）使用内线可信第三方的抗抵赖

在所有交互中作为中介进行动作的在线可信第三方称为内线可信第三方。

内线可信第三方设施能够被特定的事件或动作显式地请求，或者可被隐式地提供。而内线可信第三方，在抗抵赖服务被请求的所有交互中扮演中介角色，并可向证据使用者（如判决者）提供证据。在所有情况下，内线 TTP 将中继数据并监视该事件或动作。

内线 TTP 向请求抗抵赖服务的证据使用者提供证据，TTP 保管可以作为证据的数据或数据的数字指纹。

（6）使用公证的抗抵赖

公证为多个实体内通信数据的属性提供保证，例如完整性、原发、时间和目的地。公证者受所涉及的实体委托，以一种可测试方法持有提供担保所需的必要信息，以及为将来解决纠纷而保管记录。数字签名、加密和完整性机制可在适当时候用于支持公证所提供的服务。

在证据生成角色中，公证者将记录保证数据属性的证据，另外，可使用记录号来标识证据。

在证据验证角色中，公证者将确定证据的合法性。

7　认证

7.1　要求

a)　在通信双方建立连接之前，应用系统应利用密码技术进行会话初始化验证；

b)　在通信过程中应保证数据来源的可靠性。

7.2　应用范围

认证是证实某事是否真实或是否有效的一个过程，用来确保数据发送者和接收者的真实性以及数据的完整性，阻止对数据的主动攻击。

7.3　实现认证的方法

认证或称为鉴别，认证的基本目标是证明实体的身份和消息的来源，提供主角身份的保证，防止其它实体占用被认证实体的身份，防止欺骗和伪装等攻击。

认证对抗的主要安全威胁是“冒充”和“重放”攻击。冒充是指一个实体作为另一个不同的实体出现，或者说一个实体以特定的方式伪装成与验证者相关的另一个实体，冒充通常与其它攻击方式（如修改）一起使用，为对抗与冒充相关的威胁，认证必须结合完整性服务一起使用；重放是指重复信息的全部或部分内容，重放经常与其它攻击（如数据修改）组合使用，认证服务用于对抗重放，因为它能够确定被交换信息的来源。认证信息可以分为交换认证信息（交换 AI）、申请认证信息（申请 AI）和验证认证信息（验证 AI）这三类。

认证过程可以包括下面这些阶段：

(1) 安装阶段：

定义申请认证信息和验证认证信息。

(2) 更改认证信息阶段：

主角或管理员修改申请认证信息和验证认证信息（如修改口令）。

(3) 分发阶段：

将验证认证信息分发给 验证交换认证信息 中要使用的实体（如申请者或验证者）。例如：在离线方式中，实体可以获取认证信证书、证书撤销列表和机构撤销列表。分发阶段可以在传送阶段之前、期间或之后。

(4) 获取阶段：

申请者或验证者可获取为生成用于认证实例的具体交换认证信息所需的信息。通过与可

信第三方的交互或通过认证实体间的报文交换（信息交换），不同的规程可以获取不同的交换验证信息。例如：当使用在线密钥分发中心时，申请者或验证者可从密钥分发中心获取一些信息，如认证证书，使得能与其它实体进行认证。

(5) 传送阶段：

在申请者和验证者之间传送交换认证信息。

(6) 验证阶段：

对比检查交换认证信息和验证认证信息。在此阶段，不能验证交换认证信息本身的实体可以与将执行交换认证信息验证的可信第三方联系，第三方将完成对交换认证信息的验证，并送回一个肯定或否定的响应。

(7) 关闭阶段：

建立一种状态，使得先前能被认证的主角临时不能被认证。

(8) 重开启阶段：

关闭阶段所建立的状态被终止。

(9) 卸载阶段：

一个主角从众多主角中取消。

认证的这些阶段可以重叠交错，在特定情况下某些阶段可以省去，并且各阶段的顺序也可能不同。

数据库管理系统中应对登录到系统中的用户进行身份真实性鉴别，通过对用户所提供的“鉴别信息”的验证，证明该用户确有所声称的某种身份，这些“鉴别信息”必须是保密的，不易伪造的，按照 GB/T 20271-2006 中 6.3.3.1 的要求，实现数据库管理系统的身份鉴别功能。

7.4 认证机制类型

认证是安全服务的基础，许多安全服务都是建立在认证的基础之上。

按认证的范围分，认证可以分成数据源认证和实体认证。

a) 数据源认证（也称为消息认证）：其作用是验证消息是否来自特定的实体，并且事后不能改变。这一服务是由接受者在接受时进行验证，目的是确认消息发送者的身份及数据的完整性。

b) 实体认证：其作用是用来证实某个实体就是他所声称的实体，待鉴别的实体通过表明他确实知道某个秘密来证实其身份，使通信双方能够验证对方的身份。

实体认证更多地涉及验证消息发送者所声称的身份，认证方法依赖于一个或多个原则相关的一系列假设，这些原则一般包括：某些已知的东西（如口令）、某些已经拥有的东西（如物理密钥或卡）、某些永远不变的东西（如指纹等生物特性）、接收第三方（可信第三方）已经确立的认证和上下文（如主角的地址）。所有原则都有各自的缺点，可以通过多个原则的组合使用来克服这些缺点。

认证可以是单向的也可以是双向的，单向认证只提供一个主角身份的保证，如果要同时提供对通信双方主角身份的保证，则为双向认证或相互认证。实体认证可以是单向或双向的，数据源认证就其本质来说总是单向的。

认证的方法可以是对称/非对称的，对称类（如口令、使用对称密钥技术加密的盘问）中认证双方共享公共认证信息，非对称类（如非对称密钥技术、能够在不泄露其中任何一部分信息的情况下而验证信息的拥有技术）中不是所有认证信息都被认证双方共享。

按是否有可信第三方参与，认证机制可以分为有可信第三方参与的认证和无可信第三方参与的认证。

（1）有可信第三方参与的认证中，验证认证信息能够通过与可信第三方的交互操作获取，其完整性必须得到保证。

认证过程可以包括一个可信第三方或多个可信第三方（一个可信第三方链），多个可信第三方的引入使得在大量实体间的认证中，每一个实体只需要维护有限个实体（非所有实体）的信息。

有可信第三方参与的认证又分为内线认证、在线认证和离线认证。

1）内线认证

内线认证情况下，可信第三方直接介入申请者和验证者之间的认证交换。主角由在后续内线认证交换中担保其身份的中间者认证。内线认证要求验证者信任中间者能够正确认证主角，并且要求通过认证向验证者保证中间者的身份。

2）在线认证

在线认证情况下，一个或多个可信第三方被包含在某个认证交换的每一个实例中。与内线认证不同的是，在线可信第三方不直接位于申请者和验证者认证交换的路径上。在线可信第三方能被申请者要求产生交换认证信息以便能帮助交换认证信息进行交换验证。在线可信第三方能够生成在线认证证书。在线认证要求在验证者和能够证明主角申请认证信息有效性的可信第三方之间，存在一个在生成交换认证信息过程中所包含的可信第三方链。在最简单的情形下，只需要一个可信第三方直接与申请者或验证者交互。然而，这能够被扩展到直接或者间接与申请者或验证者通信的可信第三方链。

3）离线认证

离线认证是由需要使用撤销证书的签发列表、撤销证书的证书列表、证书超时、或用于验证认证信息撤销的其它非直接方法所表示的。在离线认证情况下，一个或多个可信第三方支持认证而无需参与其中的每一个认证实例。离线可信第三方提前生成和分发被验证者以后用于验证认证交换有效性的离线认证证书，认证交换不需要机构的介入而独立进行。由于可信第三方不能在认证发生时直接与申请者或验证者交互，所需的交互次数少、效率高。

（2）在无可信第三方参与的认证中，最简单的情形，申请者和验证者在生成和验证交换认证信息时都得不到其它任何实体的支持，所以用于主角的验证认证信息必须已经安装在验证者中。

如果在大规模通信环境中使用这类方法，大多数实体的通信对象个数要严格控制。可能出现这种最坏的情况：验证者被要求知道安全域中所有主角的验证认证信息，如果不限制通信对象个数，信息需求量因随参与实体个数的平方而快速增长，将变得非常巨大。

国标 GB/T 15843 的第一部分中给出了实体认证机制的一般模型、要求和约束，在其它部分中分别描述了采用对称加密算法、采用数字签名技术、采用密码校验函数和使用零知识技术的机制，并给出了使用这些机制的要求。

7.5 认证机制

依据对攻击的抵抗性分类，认证机制可以被定义为下面几个等级：

等级 0：不被保护的；

等级 1：抗泄露的保护；

等级 2：抗泄露和对不同的验证者重放的保护；

等级 3：抗泄露和对同一验证者重放的保护；

等级 4：抗泄露和对不同或同一验证者重放的保护；

此处抗泄露的保护是指保护申请认证信息不被泄露。

以下（1）至（5）中对于认证交换的描述都是假设认证交换是由申请者发起的。（1）至（5）描述的交换适用于单向认证。在某些情况下有必要确认认证是否成功，可能因此需要一次附加的数据传送，此次传送在下面的（1）至（5）中没有被描述。

(1) 不被保护的机制(等级 0)

此机制中，申请认证信息与可区分标识符一起，只是简单的作为申请者到验证者的交换

认证信息被发送（如发送一个口令），难以对抗认证信息泄露和重放攻击。此机制属于对称认证，适用于数据源认证和实体认证。

生成设施直接根据输入生成交换认证信息，如图 1 所示，[]中的数字指纹为可选项；验证设施验证所接收到的申请认证信息（如口令）是否与收到的可区分标识符相关的验证认证信息相符合。

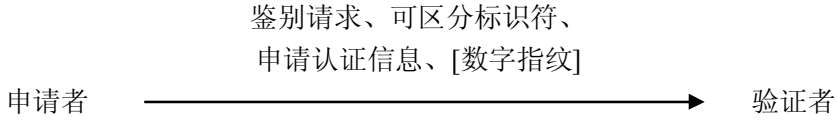

图 1 等级 0 机制示意图

(2) 抗泄露的保护(等级 1)

此类机制使用了一个转换函数，通过转换函数使用申请认证信息，申请认证信息和可区分标识符（可选）被转换，并且与可区分标识符一起被传送，而实际的申请认证信息并没有通过通信信道传送。示例有：

1）发送经过单向函数转换了的口令（例如：密码校验值或散列函数）；

2）发送由秘密密钥加密后的数字指纹；

3）发送由保密性密钥加密的口令；

4）发送由私有密钥签发的数字指纹。

此类机制适用于数据源认证和实体认证，提供了抗申请认证信息泄露的保护，但它们难以对付重放攻击。

生成设施使用申请认证信息，申请认证信息和可区分标识符（可选）及数字指纹（可选）作为密码学变换的输入来生成交换认证信息，如图 2 所示，图 2 中的[可区分标识符]和[数字指纹]为可选项，F 为转换函数。

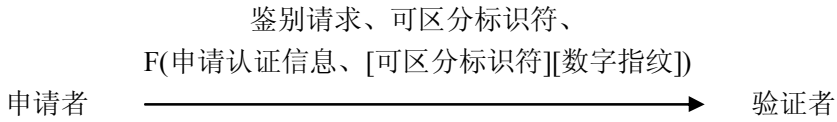

图 2 等级 1 机制示意图

转换函数包括：

1）单向函数：此时，验证设施重复施用单向函数于验证认证信息而不是申请认证信息，并且将其与所接收到的交换认证信息想匹配。

2）对称加密：此时，验证设施使用验证认证信息来解密所收到的交换认证信息，然后通过检查交换认证信息中包括的可区分特征（如申请者的可区分标识符、数字指纹的正确性、口令或一个不变值）来验证解密的正确性。

3）数字签名：此时，验证设施用所接收到的数据重新计算数字指纹，并使用验证认证信息来验证所收到的数字签名对于本数字指纹而言是有效的。

对于数据源认证，交换认证信息中的数字指纹应该和从数据申请认证中重新生成的数字指纹相符合。

为提供保密性，转换函数要求没有逆反函数，如果存在逆反函数，此逆反函数必须从计算角度来看对于以机密性形式存放着申请认证信息（和数字指纹）当事人而言是不可追踪的。转换函数可以使用单向函数、对称加密和数字签名。

(3) 抗泄露和对不同的验证者重放的保护(等级 2)

此类机制提供了对不同的验证者抵抗申请认证信息泄露和重放的保护，但不能抵抗对同一验证者的重放攻击。与等级 1 中的机制相比较，只是转换函数的输入增加了验证者独特的数据项，以此提供了额外的保护。

(4) 抗泄露和对同一验证者重放的保护(等级 3)

此类机制提供了抗泄露和对同一验证者重放的保护。

此类机制中的唯一编号机制通过使用转换函数与唯一性的信息相结合来提供对同一验证者重放的保护。申请认证信息和唯一编号被转换，并与可区分标识符一起被传送。

唯一编号的来源：

1）随机或伪随机数：这类数在申请认证信息的生存周期内不会被故意重复。

2）时戳：唯一编号可以是一个从可信来源的时戳，它在申请认证信息的生存期内是唯一的。旧的时戳将会被拒绝。

3）计数器：只要同一个申请认证信息还在，计数器的值就会不断变化。

4）密码编链(块链)：唯一编号可以是从较早的在申请者和验证者之间以块编链方式交换的数据的内容派生而来。

转换函数的三个范例：

1）单向函数：唯一编号、申请认证信息和可选的可区分标识符通过单向函数转换。唯一编号同时也被传送，验证者可以反过来执行同样的转换；

2）非对称算法：当申请认证信息是私有密钥时，唯一编号要被私有密钥签发；

3）对称算法：　当申请认证信息是秘密密钥时，唯一编号要被秘密密钥加密。

生成设施生成唯一编号，然后使用唯一编号、申请认证信息、可区分标识符（可选）和数字指纹作为输入进行加密，生成交换认证信息；验证设施解密并且用验证认证信息来验证交换认证信息的正确性，且检查收到的唯一编号以前是否收到过，如果以前收到过，则说明这是一个重放。

生成交换认证信息如图 3 所示， []中的均为可选项，F 为转换函数。

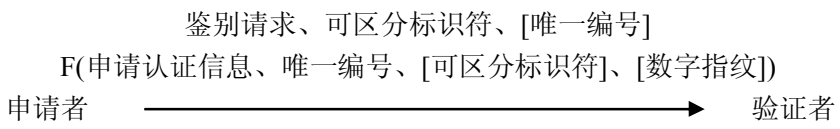

图 3 等级 3 机制示意图

唯一编号机制适用于数据源认证和实体认证，对于数据源认证，交换认证信息中的数字指纹应该和从所获得的数据中重新生成的数字指纹相匹配。

(5) 抗泄露和对不同或同一验证者重放的保护(等级 4)

1）唯一编号机制：

此类机制在等级 3 机制的基础上增加了验证者唯一的数据项作为交换过程中转换函数的输入，提供了额外的保护。

2）盘问机制：

盘问机制的目的是对抗重放攻击，该机制可以用于数据源认证和实体认证。在回应认证请求时，验证者以具有唯一值的数据项的形式向申请者发布一个盘问，申请者使用某些特定函数对盘问信息和申请认证信息作转换，然后将转换结果返回给验证者。

盘问机制包含三次信息传送：发送认证请求、发布盘问和发送响应。申请认证信息、盘问信息和可区分标识符（可选）通过转换函数被处转换成某一个值，响应应包括这个值。一般，可区分标识符应该与认证请求或最后的响应一起发送。

可能用于盘问机制的转换函数形式：

i 单向函数：盘问和申请认证信息通过单向函数作转换；

ii 非对称算法：当申请认证信息是私有密钥时，盘问要被私有密钥签发；

iii 对称算法：当申请认证信息是秘密密钥时，盘问要被秘密密钥加密；

iiii 非密码：如使用盘问响应队列表及利用生物特性。

作为盘问机制的特例，盘问的生成要依赖于从认证请求中所获得的身份，这种机制被称为专用盘问机制，此时，可区分标识符必须强制包含在认证请求中。

生成设施生成认证请求（在特定盘问情况下，此申请必须与可区分标识符一起），接收到认证请求后，验证设施生成唯一的盘问，生成设施然后用它作为转换的输入信息生成交换认证信息，如图 4 所示，[]中的为可选项，F 为转换函数。

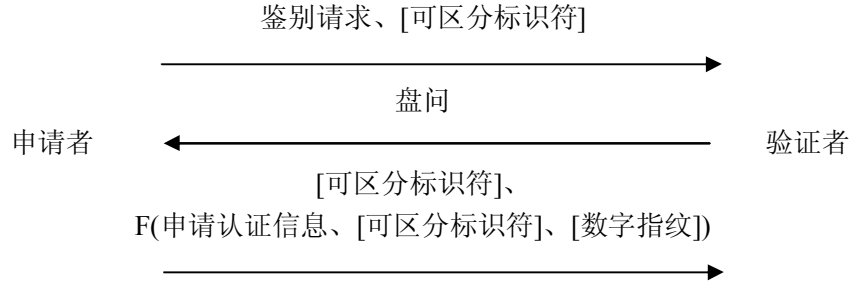

图 4 盘问机制示意图

在使用单向函数时，验证设施使用验证认证信息代替申请认证信息来重复转换，并且使用它来检验所收到的交换认证信息，为了重复这个函数，可区分标识符和此项服务需要用到的数据对验证者来说必须是可用的。在其他的转换过程中，验证设施或者重复转换或者使用反函数，并且使用验证认证信息来检查其内容。

3）专用加密盘问机制：

专用加密盘问机制也包含三次信息传送：发送认证请求信息和可区分标识符、发布经转换函数转换的盘问和验证认证信息（可能还包括可区分标识符）以及发送由盘问信息组成的响应。

此机制下转换函数可用对称和非对称加密算法。采用非对称算法时，如果申请认证信息是私有密钥，盘问要使用相应的公有密钥加密；采用对称算法时，如果申请认证信息是秘密密钥，盘问要用秘密密钥加密。

专用加密盘问机制只适用于实体认证，不适用于数据源认证。

生成设施生成认证请求，在接收到认证请求和可区分标识符后，验证设施生成一个不可预测的盘问，此盘问经由转换设施生成交换认证信息，如图 5 所示，[]中的为可选项，F 为转换函数。

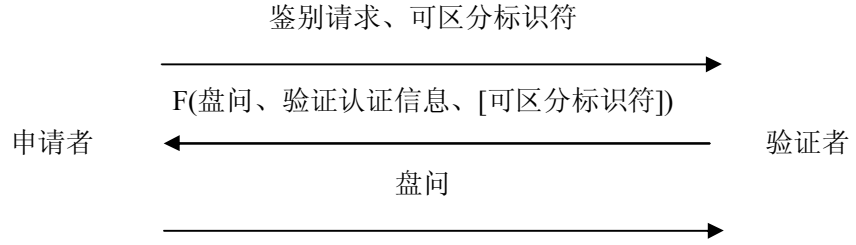

图 5 专用盘问机制示意图

生成设施然后使用申请认证信息代替验证认证信息来进行相反的转换，以获得作为交换认证信息返回的盘问，验证设施最后检查此盘问是否与较早生成的盘问一样。要注意的是此方案只与加密转换相关。

4）计算响应机制：

此类机制同样也包含三次信息发送：第一次将认证请求与备选的可选值和身份信息一起发送；第二次发布盘问，此盘问指明验证者选择了哪个值；第三次发送一个由唯一编号、盘问，或用于计算响应的所选择的值，和申请认证信息组成经过适当函数转换后的响应。

这类机制的一个示例是零知识技术，验证者从一系列的“问题”中选择一个，申请者必须在没有泄露如何解答的情况下解答这个问题。

为了提供更高层次的身份确证，交换可能需要重复。这样可以防止此类冒充攻击，象攻击者可以计算出某些验证者选择值的正确响应，但不是全部。如果只有一次交换，验证者可

能凑巧选择了一个攻击者知道正确响应的值。增加交换的次数可以减少此类型攻击成功的可能性。

生成设施首先生成一个唯一编号和备选值，将它们放入交换认证信息，如图 6 所示，[] 中的为可选项，F 为转换函数。

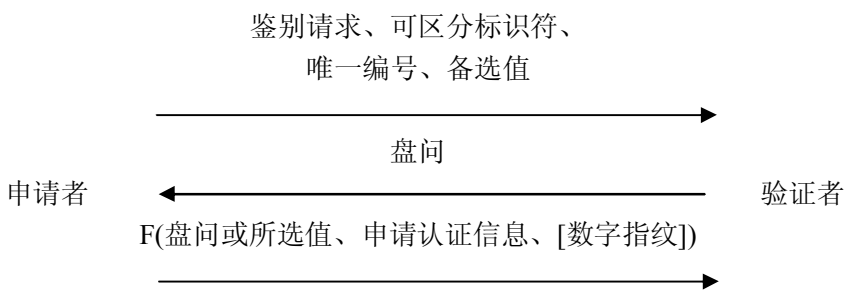

图 6 计算机响应机制示意图

验证设施从备选值中选择多个值并且生成盘问后组成第二个交换认证信息，然后生成设施使用申请认证信息来执行对盘问或者所选值的转换，验证设施最后使用验证认证信息执行一个相反的转换并检查所收到的值。

上面的（1）至（5）中都是假设申请者使用认证请求发起交换，但有些机制中包含有验证者使用认证邀请来发起交换的情况，此时传送的次数会不同。

对于包括单向交换（等级 1、2、3 和等级 4 中唯一编号）的机制，同样形式的交换可以用在双向认证中的任何一方。

对于等级 4 中的盘问机制，同类型的机制可以被用在两个方向。第一个盘问可以和认证请求一起被传送，并且转换后的第一个盘问可以和第二个盘问一起传送，这要求与单向认证同样的交换次数。

对于等级 4 中的专用加密盘问机制，第一个盘问可以和认证请求一起被传送，并且转换后的第二个盘问可以和第一个盘问一起传送。

等级 4 中的盘问机制和专用加密盘问机制可以结合起来使用。两个盘问被放在经转换后的数据中，在对称加密情况下，两端的申请认证信息和验证认证信息相同，转换只执行一次；在非对称加密时，每端都要执行两次转换。

对于等级 4 中的计算响应机制，单向认证需要三次或更多次的传送，双向认证需要四次或更多次的传送。

8 参考文献

[1] GB/T 9387.2-1995 信息处理系统 开放系统互连 基本参考模型 第 2 部分：安全体系结构
[2] GB/T 15843.1-2008 信息技术.安全技术.实体鉴别.所有部分
[3] GB/T 15851-1995 信息技术.安全技术 带消息恢复的数字签名方案
[4] GB/T 15852.1-2008 信息技术.安全技术.消息鉴别码.所有部分
[5] GB/T 16264.8-1996 信息技术.开放系统互连.目录.第 8 部分:鉴别框架
[6] GB/T 17859-1999 计算机信息系统安全保护等级划分准则
[7] GB/T 17902 信息技术.安全技术.带附录的数字签名.所有部分
[8] GB/T 17903.1-2008 信息技术.安全技术.抗抵赖.所有部分
[9] GB/T 17964-2008 信息安全技术.分组密码算法的工作模式
[10] GB/T 18238 信息技术.安全技术.散列函数.所有部分

[11] GB/T 18794.1-2002 信息技术.开放系统互连.开放系统安全框架.第 1 部分：概述
[12] GB/T 18794.2-2002 信息技术.开放系统互连.开放系统安全框架.第 2 部分：鉴别框架
[13] GB/T 18794.3-2003 信息技术.开放系统互连.开放系统安全框架.第 3 部分：访问控制框架
[14] GB/T 18794.4-2003 信息技术.开放系统互连.开放系统安全框架.第 4 部分：抗抵赖框架
[15] GB/T 18794.5-2003 信息技术.开放系统互连.开放系统安全框架.第 5 部分：机密性框架
[16] GB/T 18794.6-2003 信息技术.开放系统互连.开放系统安全框架.第 6 部分：完整性框架
[17] GB/T 20271-2006 信息安全技术.信息系统通用安全技术要求
[18] GB/T 20273-2006 信息安全技术.数据库管理系统安全技术要求
[19] GB/T 22239-2008 信息安全技术.信息系统安全等级保护基本要求
[20] GB/T 22240-2008 信息安全技术.信息系统安全等级保护定级指南