第5章 移动通信网络完整性保护与加解密机制

保密性、完整性和可用性是信息安全的三大基石，其中，完整性原则指用户、进程或者硬件组件具有能力，能够验证所发送或传送的东西的准确性，并且进程或硬件组件不会被以任何方式改变。任何网络业务的决策都建立在精确的数据之上，如果数据被损坏，则基于该数据的任何决策都是可疑的，因此保护数据完整性至关重要。完整性保护机制旨在通过阻止威胁或探测威胁，保护可能遭到不同方式危害的数据的完整性和数据相关属性的完整性，确保信息在传输过程中没有被篡改。在完整性保护机制方面，各代移动通信网络都做了不同的探索与努力，完整性机制的发展也越来越全面、可靠。下面探讨各代移动通信网络中完整性保护机制的原理及采用的算法。

5.1 2G 完整性保护与加解密机制

2G 通信的主流标准包括 GSM、IDEN、IS-136、IS-95 以及 PDC。每种标准主要在特定地区使用。例如，IDEN 主要被美国的电信系统供应商使用，IS-136 主要在美洲使用，IS-95 在美洲和亚洲的某些国家使用，而 PDC 仅在日本使用。然而，在全球范围内，通常所说的 2G 主要指 GSM 网络。

在设计阶段，GSM 的目标是创建一种能在无线环境下工作的系统，这种系统在通信质量上应与有线电话相当。因此，GSM 主要关注的是语音传输，而非数据传输。在这方面，GSM 的主要安全需求是防止窃听和非法访问用户信息，而不是防止信息被篡改或伪造。因此，GSM 系统并未提供专门的完整性保护算法。相反，它依赖于加密来防止信息被篡改。具体来说，GSM 使用加密密钥对用户通信信息和空中接口信号进行保护。通过对传输的数据进行加密，攻击者无法获取明文信息，因此无法修改密文。这种方式间接保证了信息的完整性。

进一步来说，对于以语音通信为主的 GSM 系统，不提供专门的完整性保护也是一个权衡的结果。由于语音通信的特性，即使没有完整性保护，也只会使声音识别稍微困难一些。相比之下，如果对语音信息进行完整性保护，可能会导致语音信息断断续续，影响用户体验。因此，GSM 系统选择了只通过加密用户通信信息的方式防止信息被篡改，而没

有提供针对信令、语音和用户数据的专门完整性保护。

GSM 系统由移动站(Mobile Station,MS)、基站子系统(Base Station Subsystem,BSS)和网络子系统(Network Subsystem,NSS)三部分组成。移动站和基站间通过无线链路连接,系统其他部分间则通过有线链路连接。每个移动站,即移动电话,配有一个用户识别(SIM)卡。在 SIM 卡中存有用户在 GSM 系统中的注册信息,包括给用户特定且唯一的身份标志的国际移动用户号(IMSI)、用户电话号码、鉴别算法(A3)、加密密钥生成算法(A8)、个人识别码(PIN)、单个用户鉴别密钥(K_i)等。GSM 移动电话又包含加密算法(A5)。

5.1.1　A5 加密算法

GSM 网络中使用的加密算法并未正式公开,在标准的文本中被称为 A5 系列算法。此外,GSM 系统最多可定义 7 个 A5 算法,当 MS 与网络建立连接时,MS 应向网络端指示通信过程使用的 A5 算法的版本。如果 MS 与网络端支持至少一个相同的 A5 加密算法,则从中选择一个;否则,如果网络端希望以非加密方式进行连接,则可建立非加密方式的通信。

A5 算法是一个序列密码算法,通过密钥流与明文异或来产生密文。在通信的另一端,通过同样的方式与密文异或得到明文。加密过程是由固定网络端控制的,为了使两端同步,解密首先在 BSS 上开始,然后 BSS/MSC/VLR 发送一特殊的明文给 MS;当 MS 正确地收到这个明文后,MS 开始加密和解密。最后只有在 BSS 端正确解密从 MS 发送过来的信息和帧后,BSS 端的解密才开始。

A5 算法有两个输入参数:一个是初始密钥 K_c,这是由 A8 算法生成的 64 位密钥;另一个是序列号 Fn,长度为 22b,表示当前的时间分隔多址(TDMA)帧号。在每个 TDMA 时隙中,A5 算法生成一个 228b 的密钥流,如图 5-1 所示。这个密钥流被分成两个 114b 的部分:在 MS 中,第一个 114b 的部分用于解密接收到的数据,第二个 114b 的部分用于加密要发送的数据;在 BSS 中,具体是在基站收发器中,第一个 114b 的部分用于加密要发送的数据,第二个 114b 的部分用于解密接收到的数据。这样,当数据从 BSS 发送到 MS 时,BSS 会使用密钥流的第一部分进行加密,然后 MS 会使用同样的密钥流部分进行解密。反过来,当数据从 MS 发送到 BSS 时,MS 会使用密钥流的第二部分进行加密,然后 BSS 会使用同样的密钥流部分进行解密。这确保了通信的安全性,并且由于每个 TDMA 帧都有一个独特的密钥流,所以即使攻击者能够截获一部分通信,他们也无法解密其他的通信。

A5 系列算法包含 A5/1、A5/2 和 A5/3 算法,以及近年新出现的 A5/4 算法。A5/1、A5/2 和 A5/3 算法的密钥长度为 64b,A5/4 是 A5/3 算法的另一种模式,密钥长度为 128b。除了这几种版本外,A5 算法还有另外一种实现方式,称为 A5/0,即采用非加密的方式。

A5/1 算法产生于 1987 年,为流密码算法。A5/1 算法曾经是使用最广泛的 GSM 加密算法,在设备中的支持程度也是几种算法中最高的。由于受巴统限制,该算法作为出口管制技术无法集成到在中国境内使用的设备中。该算法在 1994 年被初步泄露,在 1999

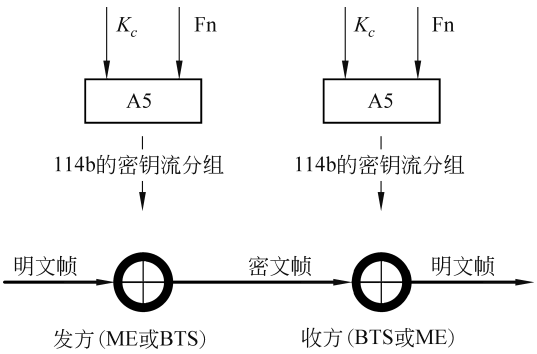


图 5-1　GSM 系统中的加密算法 A5

年通过逆向工程的方式被公布出来，从 2000 年开始即被逐步破解。2000 年，Alex Biryukov 等通过构造庞大的具备初步彩虹表概念的数据库，通过查表取代计算的方式，以空间换时间，对 A5/1 实施已知明文攻击，但该方法需要首先通过大约 2^{48} 次运算，处理约 300GB 的数据[4]。2007 年，德国 Bochum 大学搭建了具有 120 个 FPGA 结点的阵列加速器，对包括 A5/1 算法在内的多种算法进行破解[5]，由于采用 FPGA 成本较低，使 A5/1 算法破解在商业上成为可能。2009 年黑帽大会上，Karsten Nohl 等利用三个月时间制作了 2TB 的彩虹表，并宣布利用 P2P 分布式网络下的 NVIDIA GPU 显卡阵列即可破解 A5/1 算法[6]。2016 年，新加坡科技研究局用约 55 天创建了一个 984GB 的彩虹表，通过使用三块 NVIDIA GPU 显卡构成的计算装置在 9s 内完成对 A5/1 算法的破解[7]。

A5/2 算法产生于 1989 年，算法存在缺陷，一经公布即被发现算法弱点[8]，使得采用 A5/2 算法加密的数据可以被实时破解，因此 A5/2 算法被弃用。

A5/3 算法，其核心算法是 KASUMI 分组加密算法，加密方式采用流加密的形式。A5/3 算法的安全性不断受到挑战，2001 年，Kühn Ulrich 在欧洲密码会议上提出了针对 MISTY1 六轮计算的理论攻击方法[9]，2005 年，以色列研究员提出了一种理论攻击的方法[10]，可以在构造了 2^{55} 数量级的选择明文情况下，经过 2^{76} 次运算破解算法。2010 年，Dunkelman、Keller 和 Shamir 的论文论证了一种对 KASUMI 可实施的攻击[11]，其可以在一台装有英特尔酷睿双核处理器的计算机上在 2h 内破解 KASUMI 算法。由于攻击实施之前需要事先构造出百万个已知明文，并获取这些明文经过运营商网络加密之后的密文，在现实中很难做到，这种攻击并未对电信系统中的 A5/3 算法造成实质的威胁，但仍揭示了算法的不安全性。因此，负责管理 GSM 后续进展的 3GPP(第三代通信网络合作伙伴计划)在 2010 年定义了新的 A5/4 算法。A5/4 算法与 A5/3 相同，但密钥长度由 64b 扩展到了 128b。

5.1.2　A3/A8 加密算法

在 GSM 中，A3 和 A8 是两种用于身份认证和密钥生成的算法。它们通常在 SIM 卡中实现并在 GSM 网络认证中心中执行，用于保护用户的通信隐私。

A3 算法用于身份认证，防止未经授权的设备接入网络。2G 鉴权具体过程如图 5-2

所示。A3 的输入是 GSM 网络发给移动电话的一个 128b 的随机数呼叫和单个用户鉴别密钥 K_i，输出为一个 32b 的标示响应。这个标示响应发回到网络，网络使用同样的随机数和它存储的 K_i 来计算出预期的应答，并与设备的应答进行比较。自己生成的结果进行比较，如果二者相符，用户便得到了合法鉴权。

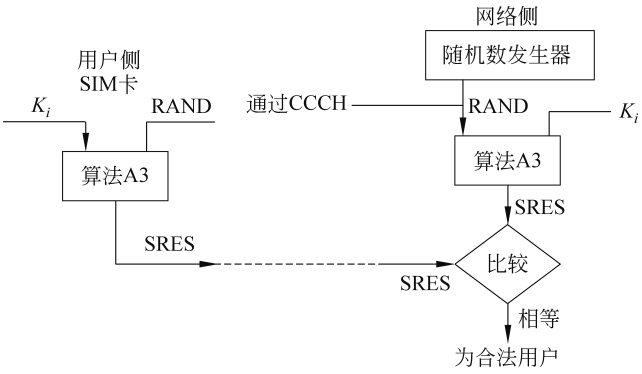


图 5-2　2G 鉴权流程

A8 算法用于生成会话密钥 K_c。一旦设备的身份得到验证，网络就会使用 A8 算法生成 K_c，其输入是 128b 随机数和 K_i，输出为一个 64b 的加密密钥 K_c，这个 K_c 被用于后续加密和解密设备与网络之间的通信。

A3 和 A8 算法不是具体的算法，而是两个算法接口，这意味着其具体的实现可能会因运营商或地区而异。GSM 中的 A3 和 A8 算法都是基于密钥的单向碰撞函数，它们通常合在一起，统称为 COMP128 算法。COMP128 算法目前被用在鉴权过程中，存储在 SIM 卡和 AUC 中。

5.1.3　A5/1 加密算法

A5 算法被用于 GSM 网络蜂窝通信，A5 加密算法在电话听筒和 BSS 之间搅乱用户语音和数据传输来提供保密性。GSM 系统中的移动电话和基站之间的信令和数据通过加密算法 A5 使用从 A8 中得到的 K_c 进行加密和解密。针对当前在使用中的三种 A5 算法 A5/1、A5/2 和 A5/0。A5/1 被认为是这三种中性能最好的加密算法。因此，下面将详细解释 A5/1 加密算法的具体流程。

A5/1 算法使用三个线性反馈移位寄存器 X(19 位)、Y(22 位)和 Z(23 位)，如图 5-3 所示。根据 A5/1 算法规定，三个寄存器中选择三个控制信号，分别为 X 的第 8 位 x_8，Y 的第 10 位 y_{10} 和 Z 的第 10 位 z_{10}。

下面由一个具体例子给出 A5/1 算法流程。假设当前寄存器状态如下。

(1) 根据算法规定，找到 $x_8=1$，$y_{10}=0$，$z_{10}=1$。

(2) 定义多数投票函数 maj(x，y，z)：如果 x、y 和 z 的多数为 0，那么函数返回 0；否则，函数返回 1。因此，$m=\text{maj}(x_8, y_{10}, z_{10})=\text{maj}(1,0,1)=1$。

(3) 根据 m 与(x_8，y_{10}，z_{10})的值对寄存器进行移位。

因为 $x_8=m=1$，所以 X 右移一位，第一位 $x_0=x_{13}\oplus x_{16}\oplus x_{17}\oplus x_{18}=0\oplus 1\oplus 0\oplus 1=$

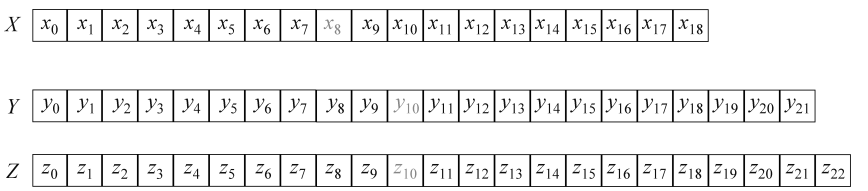
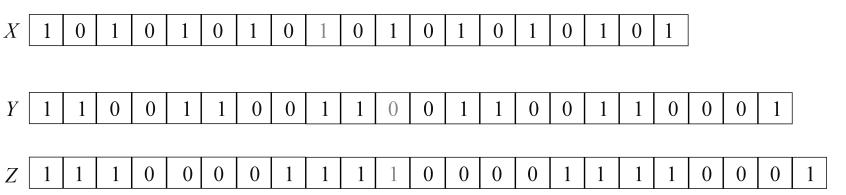
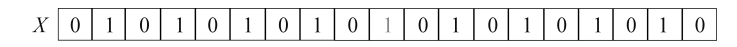


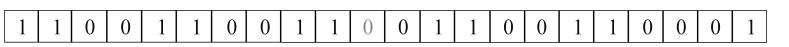
图 5-3　线性反馈移位寄存器



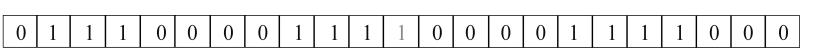
0，移位后 X 寄存器状态如下。



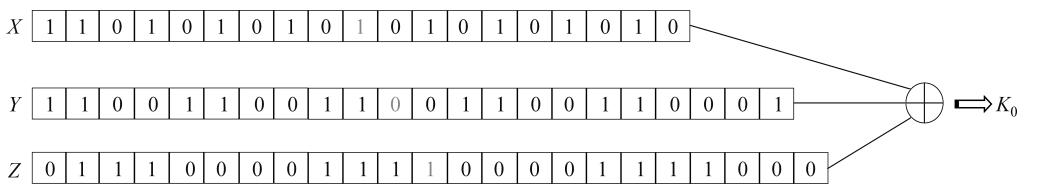
因为 $y_{10}=0$，$m=1$，所以 Y 寄存器不需要进行移位，Y 寄存器状态依旧如下。



因为 $z_{10}=m=1$，所以 Z 需要右移一位，第一位 $z_0=z_7\oplus z_{20}\oplus z_{21}\oplus z_{22}=1\oplus 0\oplus 0\oplus 1=0$，移位后 Z 寄存器状态如下。



(4) 最后得到三个寄存器的状态如下，然后将 X、Y、Z 寄存器的最后一位进行异或操作得到一位密钥，即 $K_0=x_{18}\oplus y_{21}\oplus z_{22}=0\oplus 1\oplus 0=1$。



(5) 假设需要 64 位密钥，则按照上述步骤进行 64 次循环操作即可得到 64 位密钥。

5.2　3G 完整性保护与加解密机制

随着 GSM 网络系统的广泛使用，人们开始意识到，仅依赖加密的方式保护数据的完整性是不够的。这是因为加密流程是可选的，未经加密的信息无法受到任何保护。此外，虽然对于语音信息来说，完整性的需求可能不高，但是如果信令消息被篡改或者伪造，则可能产生非常严重的问题，例如，将语音通信重定向到非法用户。信令消息中的控制消息是维护无线通信系统正常工作的重要信息，其格式与内容均是被严格定义的，因此对控制

消息的改动将会导致严重的系统问题，或者使用户受到精心构造的欺诈消息的欺骗，如伪基站的问题等。在这种情况下，3GPP 作为 3G 系统的标准化制定组织，已经规范了 3G 系统前期应用的安全接入标准，并保证了与 GSM 安全接入机制的最大兼容性。这些安全接入标准规定了保护信息机密性和完整性的算法，以及对用于用户认证和密钥协商的算法。

我国 3G 网络主要采用 TD-SCDMA 和 WCDMA 两种标准，其完整性保护机制均采用了 KASUMI 中的 f8 和 f9 算法。f8 算法被用于加密保护，被称为机密性算法，可以保护传输的信息不会泄露或被窃听；f9 算法用于保护信息的完整性，称为完整性算法，可以保护传输的信息不会受到任何形式的破坏。如果对传输信息进行任何修改、增加、删除或其他形式的破坏，都可以被检测出来。在标准中，标识算法的比特共有 4 位。除了规定的标准化算法 f8 和 f9 之外，考虑一些特殊的因素，标准也允许使用其他算法。

5.2.1　f8 算法

f8 算法是一种流密码机制，用于加密和保护传输信息，以防止信息被非法窃听或泄露。这种机密性算法主要在无线链路控制(Radio Link Control，RLC)层和介质访问控制(Media Access Control，MAC)层中执行数据的加密操作。f8 使用保密性密钥 CK 来加密和解密数据块，这些数据块的长度可以在 1～20 000 位之间变动。该算法基于 KASUMI 分组密码的输出反馈(Output Feedback，OFB)模式作为密钥流生成器进行操作。

基于分组密码操作的标准流密码模式有计数器模式和 OFB 模式，而 f8 算法并不仅仅沿用了标准的分组密码操作流密码模式，如计数器模式和 OFB 模式，它可被看作两种标准模式的结合，并且利用了反馈数据的预白化(预先混淆)。在 f8 流加密模式中，输出反馈、计数器和预白化三个元素按如下方式共同作用：首先，新生成的密钥流块会被计数器值和预白化数据块进行逐位异或修正，然后这个结果再送回到生成器函数作为其输入。在解密过程中，只需将相同的密钥流与密文流进行异或运算，就可以得到原始的明文数据流。

图 5-4 说明了加密算法 f8 通过使用密钥流来对明文加密的过程，其中，明文和密钥流按异或运算得到密文。明文也可通过密钥流与密文进行异或运算来恢复。

f8 算法接收一系列输入参数，包括加密密钥 CK、计数器 COUNT-C、承载标识 BEARER、传输方向 DIRECTION 和要求的密钥流长度 LENGTH。基于这些输入参数，f8 算法产生一个输出密钥流块。这个密钥流块用于对待传输的明文进行加密，从而产生相应的密文。

主要参数简要说明如下。

- COUNT-C：加密序列号，长度为 32b。每个无线承载在上行和下行链路上都有一个独立的 COUNT-C 值，初始值由相应的 HFN 确定。
- CK：完整性密钥，长度为 128b。CK 在身份验证协商期间，在 HLR/AUC 中产生并发送到 VLR/SGSN。作为五元组的一部分存储在 VLR/SGSN 中，然后由 VLR/SGSN 发送到 RNC。

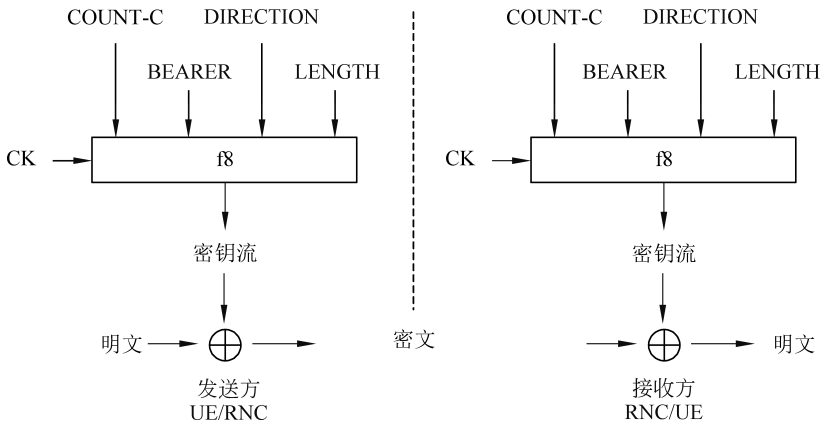


图 5-4　在无线接入链路上传输的用户和信令数据的加密

- BEARER：无线承载标识符，长度为 5b。每一个 10ms 物理层帧上复用的无线承载都有一个与之相关的 BEARER 参数。采用无线承载标识符是为了避免对不同的密钥流使用相同的输入参数集。
- DERECTION：传输方向。上行时（信息从 UE 到 RNC）DIRECTION 的数值是 0，下行时值为 1。由于上行和下行的信道有可能使用相同的密钥，DIRECTION 位的目的就是避免上下行传输时使用相同的密钥流。
- LENGTH：LENGTH 是一个 1～20 000 的整数，共 16b 代表密钥流长度。
- 密钥流：密钥流块的长度等于输入参数 LENGTH 的值。
- 明文：明文块的长度等于输入参数 LENGTH 的值。这里一次输入的明文长度为 256b。
- 密文：密文块的长度等于输入参数 LENGTH 的值。

这种加密和解密的操作使得 f8 算法能够在保持数据安全的同时，满足无线通信系统对数据处理效率和实时性的要求。然而，类似于所有的加密算法，f8 的安全性也取决于密钥 CK 的管理和使用方式。特别是，如果 CK、计数器值或预白化数据块被重复使用，可能会导致加密的安全性降低。因此，在使用 f8 算法的过程中，必须确保这些关键数值的唯一性和安全性。

5.2.2　f9 算法

f9 算法是一种完整性保护算法，用于确保传输信息的完整性，防止信息被篡改或损坏。这种完整性保护主要在 RRC 层实施。f9 算法的工作原理与 f8 算法类似，它利用 KASUMI 算法生成完整性消息认证码 MAC-I，以对 UE 和 RNC 之间的无线链路上的信令数据进行完整性保护，并对信令数据来源进行认证。在使用 f9 算法时，首先对信令数据 MESSAGE 进行处理，计算出完整性消息认证码 MAC-I，然后将此认证码附加到 MESSAGE 的尾部，一起在无线链路上发送到接收端。接收端在收到数据后，也将采用 f9 算法对收到的 MESSAGE 进行相同的计算，得出一个消息认证码 XMAC-I。然后，接收端将自己计算出的 XMAC-I 和接收到的 MAC-I 进行比较，以此来验证数据的完整性。

f9 算法如图 5-5 所示。

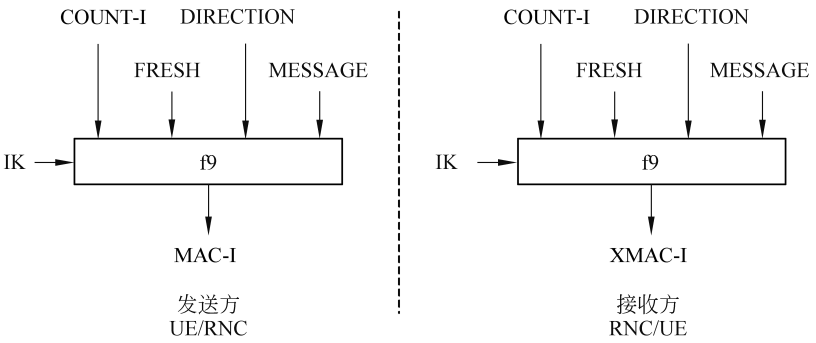


图 5-5　信令消息的完整性保护方法

由图可知，f9 算法接收一系列输入参数，包括完整性密钥 IK、完整性序列号 COUNT-I、随机数 FRESH、传输方向 DIRECTION 和信令消息 MESSAGE。发送方利用 f9 算法计算出消息认证码 MAC-I，随消息一起发送出去，接收方计算 XMAC-I 并与收到的 MAC-I 相比较以验证消息的完整性。

主要参数简要说明如下。

- COUNT-I：完整性序列号，长度为 32b。对于无线承载的信令，每个上行和下行链路中都有一个 COUNT-I 值。每当有一条受到完整性保护的信令消息时，此值就增加 1。用户会存储上一次连接中最常使用的超帧号，并将其值增加 1，以此来确保网络侧没有重新使用任何 COUNT-I 值。
- FRESH：临时随机值，长度为 32b。每个用户都有一个 FRESH 值。在建立连接时，RNC 会生成一个临时随机值 FRESH，并将其附加到 security mode command 消息中，然后发送给用户。在接下来的连接过程中，FRESH 值会被网络和用户使用。这个随机参数值确保了用户没有使用任何旧的 MAC-I，即使同一个完整性密钥可能会被用于多次连续的连接。
- MESSAGE：无线承载身份会被添加到信令消息之前。值得注意的是，尽管无线承载身份不会随消息一起发送，但为了避免对不同的 MAC 实例使用相同的输入参数集，它仍然是必要的。
- DIRECTION：方向标识符，长度为 1b。DIRECTION 的作用是防止计算 MAC-I 的完整性算法使用相同的输入参数集。从用户设备(UE)发送到无线网络控制器(RNC)的消息中，DIRECTION 的值为 0，从 RNC 发送到 UE 的消息中，DIRECTION 的值为 1。
- IK：完整性密钥，长度为 128b。对于电路交换连接，会在电路交换服务域和用户之间建立一个 IK。对于 UMTS 的用户来说，IK 是在 UMTS AKA 过程中建立的，作为完整性算法 f4 的输出，在 USIM 和 HLR/AUC 中是有效的。
- MAC-I：完整性保护的消息认证码，由算法模块生成。在 HE/AUC 中生成后，与 USIM 中生成的 XMAC-I 进行比较，完成用户对网络的认证。USIM 中的 XMAC-I 也是使用 f9 算法模块生成的。

5.2.3 与 CDMA 2000 的比较

另外一种 3G 标准 CDMA 2000 是美、韩支持的 3G 方案，由 IS-95 演进而来。它的认证协议也采用了“请求-响应”的形式。整个认证方式与 WCDMA 类似，但其认证是以基站为核心的，只有其增强认证和加密模式才可提供用户和网络的相互认证。其异同点如表 5-1 所示。

表 5-1 WCDMA 与 CDMA 2000 的安全方法比较

标　　准	WCDMA	CDMA 2000
演进策略	由 GSM 演进	由 IS-95 演进
用户身份保密方式	IMSI 号和 TMSI 号	IMSI-I 号和 IMSI-T 号
认证措施	请求-响应形式	请求-响应形式
共享秘密信息	*K*	SSD
加密算法	KASUMI 算法	美国公用加密算法
密钥长度	CK 和 IK 均为 128b	A-key 64b
是否防范伪基站攻击	支持	增强认证模式下支持
是否支持 MEXE	支持	目前尚不支持

5.3 4G 完整性保护与加解密机制

在 4G 网络技术的初期，尽管 4G 已经完全面向数据通信的网络，但是完整性保护仍然只针对信令，用户面并未进行完整性保护。这是因为在数据业务的早期发展阶段，系统设计要求的重点在于最大化系统空口的吞吐效率和最小化时延。假设平均数据报文长度为 100B，而完整性保护需要增加的 MAC-I 长度为 32b(4B)，这意味着数据包需要增加约 4%的额外开销。

对于大块或大流量的数据，机密性可以提供一定程度的完整性保护，因为需要篡改的核心数据是难以定位的。特别是对于数据承载类业务中的语音(VoIP/VoLTE)类业务、流媒体类业务等，根据前述 3G 语音消息完整性分析，这些业务的完整性保护并不必要。

另外，无线通信网络中传输的数据主要是 IP 数据包，而大部分数据包通过 TCP 承载，TCP 本身提供了完整性检查和校验能力，因此在无线通信系统中再次提供完整性保护可能会导致重复保护。

在 4G 网络中，有三种加密算法，被称为 EEA(EPS Encryption Algorithm)系列，分别是 SNOW 3G 算法、AES 算法和 ZUC 算法。这三个加密算法被封装以实现完整性保护/校验和加密/解密流程，这些算法既在 AS 层数据转发中使用，也在 NAS 层信令流程中使用。对应的名称如下：Snow 3G 被封装为 4G 完整性保护/校验算法 128-EIA1 和加密/解密算法 128-EEA1；AES 被封装为 4G 完整性保护/校验算法 128-EIA2 和加密/解

密算法 128-EEA2；ZUC 被封装为 4G 完整性保护/校验算法 128-EIA3 和加密/解密算法 128-EEA3。

5.3.1　4G 中完整性保护/校验算法

4G 完整性保护流程如图 5-6 所示。EIA 算法接收一系列输入参数，包括密钥 KEY、计数器 COUNT、承载标识 BEARER、传输方向 DIRECTION 和消息本身 MESSAGE。主要参数简要说明如下。

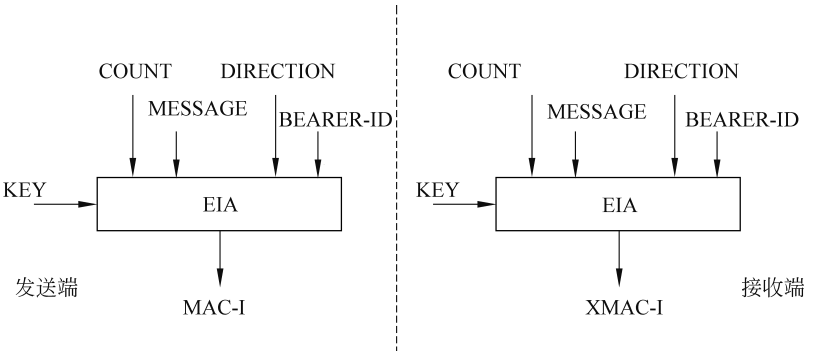


图 5-6　4G 完整性保护流程

- KEY：128b 密钥。
- COUNT：32b 计数器，每次完整性保护/校验一次就加 1。
- BEARER：5b 无线承载标识符。
- DIRECTION：1b 传输方向，标识上行还是下行。
- MESSAGE：完整性保护/校验的消息体。

基于输入参数，发送方使用完整性算法 EIA 计算 32b 消息身份验证码 MAC-I，MAC-I 随后在发送时附加到消息中。接收方在接收到的消息上计算 XMAC-I，方法与发送方在发送的消息上计算 MAC-I 的方式相同。通过将其与接收到的 MAC-I 进行比较，验证消息的数据完整性。

需要注意的是，128-EIA0 意味着空完整性保护，不提供安全保护，其算法的实现方式应为生成全零的 32b MAC-I 和 XMAC-I。当 EIA0 被激活时，重放保护将不被激活。EIA0 仅应用在处于受限服务模式的 UE 进行紧急呼叫时。在不需要通过认证的紧急会话支持的部署中，应在 AMF 中禁用 EIA0。

5.3.2　4G 中加密/解密算法

4G 加解密流程如图 5-7 所示。EEA 算法接收一系列输入参数，包括密钥 KEY、计数器 COUNT、承载标识 BEARER、传输方向 DIRECTION 和 LENGTH，主要参数简要说明如下。

- KEY：128b 密钥。
- COUNT：32b 计数器，每次完整性保护/校验一次就加 1。