



# 中华人民共和国通信行业标准

YD/T 2922-2015

---

## 数字蜂窝移动通信网通用认证架构 通用自启动架构推送功能及推送层协议

Digital cellular mobile communication network  
Generic authentication architecture-Generic bootstrapping  
architecture push function and push layer

(3GPP TS 33.223 V10.0.0, Generic Authentication Architecture (GAA); Generic  
Bootstrapping Architecture (GBA) Push function, 3GPP TS 33.224 V10.0.0,  
Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture  
(GBA) Push Layer, MOD)

2015-07-14 发布

2015-10-01 实施

---

中华人民共和国工业和信息化部 发布



## 目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 符号和缩略语	2
3.1 符号	2
3.2 缩略语	2
4 GBA要求	3
4.1 GBA概述	3
4.2 GBA架构	4
5 GBA Push架构及要求	4
5.1 介绍	4
5.2 GBA Push架构	5
5.3 GBA Push的要求	6
5.4 GPL的要求	9
6 GBA Push功能	11
6.1 GBA Push消息流程及处理	11
6.2 数据对象	14
6.3 GPI完整性保护及机密性保护	16
6.4 使用NAF SA的流程	17
7 GPL处理功能	17
7.1 处理模型	17
7.2 会话开始	18
7.3 会话终止	19
7.4 GPL安全协商	19
7.5 联合传输	19
7.6 消息格式	19
7.7 接收处理	20
7.8 外发处理	21
7.9 GPL-SA初始化	22
7.10 加密套组	23
附录A (资料性附录) 选择Disposable-Ks模型的原因	24
附录B (规范性附录) GBA-Push的 UE注册流程	25
附录C (资料性附录) GPL使用场景	26
附录D (资料性附录) 本标准与3GPP TS 33.223和3GPP TS 33.224章节对应关系	30



## 前 言

本标准按照GB/T 1.1-2009给出的规则起草。

本标准使用重新起草法修改采用3GPP TS 33.223 V10.0.0《通用认证架构; 通用自启动架构推送功能》和3GPP TS 33.224 V10.0.0《通用认证架构; 通用自启动架构推送层协议》，技术内容一致，附录D中列出了本标准与3GPP TS 33.223和3GPP TS 33.224的章条编号对照表。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本标准由中国通信标准化协会归口。

本标准起草单位：华为技术有限公司、中国信息通信研究院、上海贝尔股份有限公司、中国联合网络通信集团有限公司、中兴通讯股份有限公司、摩托罗拉（北京）移动技术有限公司。

本标准主要起草人：应江威、许怡娴、崔 洋、袁 琦、胡志远、高 枫、李 阳。



# 数字蜂窝移动通信网

## 通用认证架构

### 通用自启动架构推送功能及推送层协议

#### 1 范围

本标准定义了一种建立在通用认证架构（GAA）上的通用自启动架构推送功能（GBA Push），主要包括GBA推送架构、GBA推送功能。本标准还相应定义了一种实现GBA推送功能的推送层协议（GPL）。GPL标准内容包括消息格式、加密套组、处理模型，并假设密钥及其他安全协商（SA）参数以NAF SA的形式预配置在Push-NAF和UE中。GPL是一种可以用于单向保护的安全协议。GPL基本原理是，如果要求每个应用定义独自的安全机制，这明显会导致重复性的工作、标准化、及具体实现，而使用通用安全推送层GPL可以很好地解决这些问题。此外，GPL还可以使应用避免去了解安全层的内部工作原理。

本标准适用于应用层业务，例如手机电视（MBMS）、安全定位服务（SUPL）等，为其提供统一的安全认证服务以及通信密钥协商服务，以保证应用业务的安全性。

#### 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

ISO/IEC 10118-3:2004 信息技术-安全技术-散列函数-第三部分：专用散列函数（Information Technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions）

3GPP TR 21.905 3GPP标准词汇表（Vocabulary for 3GPP Specifications）

3GPP TS 29.109 通用认证架构；基于Diameter协议的Zh和Zn接口；阶段3（Generic Authentication Architecture (GAA); Zh and Zn Interfaces based on the Diameter protocol; Stage 3）

3GPP TS 31.101 UICC终端接口；物理和逻辑特性（UICC-terminal interface; Physical and logical characteristics）

3GPP TS 31.111 通用用户身份识别模块的应用工具包（Universal Subscriber Identity Module (USIM) Application Toolkit (USAT)）

3GPP TS 33.1023 G安全的安全架构（3G Security; Security architecture）

3GPP TS 33.2103 G安全的网络域安全中的IP网络层安全（3G Security; Network Domain Security; IP network layer security）

3GPP TS 33.220 通用认证架构；通用自启动架构（Generic Authentication Architecture (GAA); Generic bootstrapping architecture）

3GPP TS 33.222 通用认证架构；使用超文本接入网络应用功能；传输协议位于TLS传输层安全上（Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)）

ETSI TS 102 483 UICC终端接口；UICC和终端之间的互联网协议连通性（UICC-Terminal interface; Internet Protocol connectivity between UICC and terminal）



ETSI TS 102 600 UICC终端接口; USB接口特性 (UICC-Terminal interface; Characteristics of the USB interface)

FIPS PUB 180-2 (2002) 安全散列标准 (Secure Hash Standard)

FIPS PUB 197 高级加密标准 (Advanced Encryption Standard)

IETF RFC 2104 (1997) HMAC: 用于消息认证的加密散列法 (HMAC: Keyed-Hashing for Message Authentication)

IETF RFC 2246 (1999) TLS协议版本1 (The TLS Protocol Version 1)

IETF RFC 4330 用于IPv4、IPv6和OSI的简单网络时间协议SNTP的第4版本 (Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI)

NIST Special Publication 800-38A分组块加密操作模式的推荐 (Recommendation for Block Cipher Modes of Operation)

NIST Special Publication 800-38 A (2001)分组块加密操作模式的推荐-方法和技术 (Recommendation for Block Cipher Modes of Operation - Methods and Techniques)

OMA-WAP-TS-WSP-V1\_0-20020920-C 无线会话协议1.0版 (Wireless Session Protocol 1.0)

### 3 符号和缩略语

#### 3.1 符号

3GPP TR21.905和3GPP TS 33.220界定的和下列符号适用于本文件。

AUTN (\*): GBA上下文中, GBA\_ME根据AUTN值来验证认证向量来自于一个被授权的网络, 而GBA\_U根据AUTN\*值来进行对网络的认证, 具体描述参考文献3GPP TS 33.220。AUTN (\*) 用来指示AUTN和AUTN\*。

Disposable-Ks模型: 用于GBA-push的密钥生成模型。每个Ks只能用于生成一个NAF-key, 并且Ks不能重复使用。

GBA\_U aware UICC: 支持GBA\_U的UICC卡, GBA\_U代表Ks不能离开UICC卡。

GBA-Push-Info: GBA-Push-Info包含GBA-Push中用于密钥生成的相关数据, 其通过Upa参考点从NAF发送到UE。

NAF\_Id: NAF的FQDN, 与Ua安全协议标识相连接。

NAF-key: NAF-key由Ks获取, 可以用来指示Ks\_(int/ext)\_NAF或者Ks\_NAF。

NAF SA: NAF和UE之间基于NAF-key的安全协商。

Push-message: 通过Ua参考点从NAF发送给UE的消息, 该消息使用了从Upa参考点自启动过程中获取的GBA密钥。

Push-NAF: 被授权使用GBA-Push的NAF。

UE\_Trp: 用于将GPI传递给UE的传输地址。

SN\_h: 拥有有效MAC值的GPL消息中的最大序列号, 用于重放保护。

SN\_s: 用于产生发出消息序列号的计数器。

#### 3.2 缩略语

3GPP TR 21.905列出的和下列的缩略语适用于本文件。

BSF	Bootstrapping Server Function	自启动服务器功能
-----	-------------------------------	----------



B-TID	Bootstrapping Transaction Identifier	自启动传输标识
FQDN	Fully Qualified Domain Name	正式域名
GAA	Generic Authentication Architecture	通用认证架构
GBA	Generic Bootstrapping Architecture	通用自启动架构
GBA_ME	ME-based GBA	基于ME的GBA
GBA_U	GBA with UICC-based enhancements	基于UICC增强的GBA
GPI	GBA Push Info	GBA推送信息
GPL	Generic Push Layer	通用推送层
GPL_MEME中的GPL	GPL hosted in the ME	
GPL_UUICC中GPL	GPL hosted in the UICC	
GUSS	GBA User Security Settings	GBA用户安全设置
HLR	Home Location Register	归属位置寄存器
HSP	High Speed Protocol	高速协议
HSS	Home Subscriber Server	归属用户服务器
KDF	Key Derivation Function	密钥获取功能
Ks_NAF	NAF-key in GBA_ME mode	GBA_ME模式下的NAF-key
Ks_int_NAF	UICC internal NAF-key in GBA_U	GBA_U下的UICC内部NAF-key
Ks_ext_NAF	UICC external NAF-key in GBA_U	GBA_U下的UICC外部NAF-key
MAC	Message Authentication Code	消息验证码
ME	Mobile Equipment	移动设备
NAF	Network Application Function	网络应用功能
P-TID	Push Temporary Identifier	推送临时标识
SA	Security Association	安全协商
SAID	Security Association Identifier	安全协商标识
SN	Sequence Number	序列号
UE	User Equipment	用户设备
USS	User Security Setting	用户安全设置

## 4 GBA 要求

### 4.1 GBA 概述

GBA Push是在GBA安全机制基础之上提出的,而且GBA Push中的很多的定义、流程、及密钥推演都是基于GBA。GBA是一种为UE和网络侧NAF服务器之间建立共享秘密Ks而提出的,并基于3GPP AKA的通用密钥协商机制。

AKA是移动网络所用的一个非常有用的机制,而GBA重用AKA机制来建立应用层安全。GBA引入了一个新的网元BSF,该BSF与HSS之间有一个接口Zh。

UE和HSS通过BSF来运行AKA,根据运行AKA获得的结果(CK、IK),在BSF和UE之间协商产生一个共享密钥Ks,UE同时根据Ks推演Ks\_NAF。当EU和NAF需要建立连接时,应用服务器NAF能从BSF获得从Ks推演得到的会话密钥Ks-NAF和签约用户信息。通过这种方式,NAF和UE就能拥有一个共享密钥



$Ks\_NAF$ , 该共享密钥能为随后的应用提供安全保护, 特别是在应用会话开始时认证UE和NAF。UE和BSF之间的通信、NAF和BSF之间的通信、BSF和HSS之间的通信独立于应用。

## 4.2 GBA 架构

GBA网络架构如图1所示。

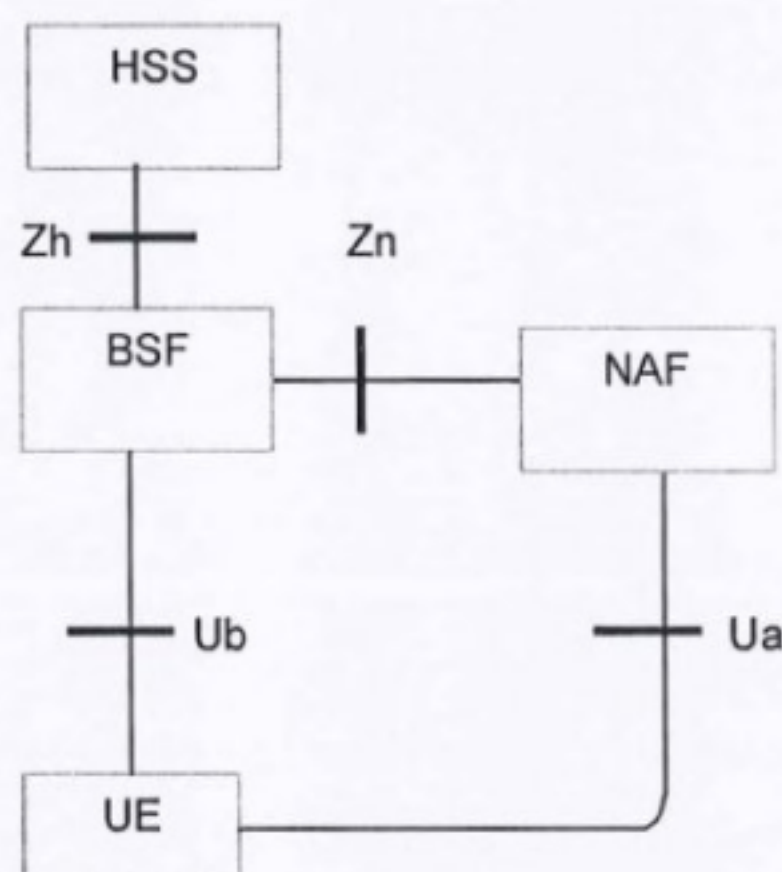


图1 GBA网络模型

图1显示了GBA网络实体模型和他们之间的参考点, 这些实体将包含在UE与网络侧之间进行的通用自启动过程中。

GBA安全过程主要由初始化、自启动安全协商、安全协商的使用三部分组成。

初始化过程是一个UE和NAF通过Ua口协商是否需要使用GBA的过程。UE要与NAF进行通信, 但是UE不知道NAF是否需要通过GBA方式产生的共享密钥, 因此双方首先需要协商是否使用GBA方式来建立应用层安全连接。

自启动安全协商过程是UE与网络侧BSF之间通过Ub口进行安全认证和协商共享密钥 $Ks$ 的过程。如果UE上存在有效的身份信息TMPI/B-TID, 则向BSF发送携带用户身份信息TMPI/B-TID的请求, BSF根据域名识别TMPI/B-TID并查询数据库判断本地是否存在该TMPI/B-TID, 如果存在则提取对应的用户身份标识IMSI/IMPI并据此从用户归属网络服务器HSS中获取AV, 否则BSF应要求用户发送包含用户身份标识IMSI/IMPI的鉴权请求。然后UE和BSF基于AKA完成认证和密钥IK、CK的协商。最后通过串连CK、IK来生成共享密钥 $Ks$ , UE同时生成 $Ks\_NAF$ 。

NAF SA安全协商的使用过程是UE与网络应用实体NAF之间协商共享密钥 $Ks\_NAF$ 的过程。在UE通过Ua口向NAF发起基于GBA的应用请求后, NAF通过Zn口向BSF请求 $Ks\_NAF$ , BSF查找与该UE和NAF相关的 $Ks$ , 然后根据 $Ks$ 生成 $Ks\_NAF$ 并将其返回给NAF。此时, UE和NAF就已经共享了密钥 $Ks\_NAF$ 。

一旦完成上述三个GBA自启动安全协商流程, 就达到了GBA自启动安全协商的目的, 能够实现UE和NAF在Ua口上的安全通信。

## 5 GBA Push 架构及要求

### 5.1 介绍

#### 5.1.1 综述

GBA-Push是一种自启动NAF和UE间安全的机制, 不强制UE联系BSF来发起自启动过程。GBA-Push与3GPP TS 33.220中的GBA紧密相关并且基于GBA。GBA-Push针对GBA\_U和GBA\_ME两种使用场景。



### 5.1.2 GBA-Push 系统概述

基于GBA-Push系统的解决方案及其特性，本节的系统概述对该总体思路进行了大概描述。

一般的使用场景是NAF发起建立共享安全协商（SA），即在NAF和UE之间建立一个NAF SA。NAF向UE推送GBA-Push-Info（GPI）信息，用于建立NAF SA。NAF SA中使用的是NAF-key，根据3GPP TS 33.220中GBA定义的方法生成，而GPI由NAF从BSF请求获取。

在NAF SA建立后，NAF就能发送受保护的推送消息给UE。如果存在由Ua口应用所定义的返回通道，则UE也可以用SA保护返回给NAF的回复消息。如何使用NAF SA不在本规范研究范围内。NAF SA由上行、下行SA标识符来指示。

GBA-Push针对GBA\_U和GBA\_ME两种场景。如果只是用GBA-Push建立一个外部NAF-key，则应使用GBA\_Push。基于GBA\_U的GBA-Push会同时建立内部NAF-key和外部NAF-key。

GBA-Push使用Disposable-Ks（Ks用完就可以丢弃）模型。该模型中的Ks只能使用一次来获取一组NAF-keys（以及用于保护GPI信息传递的其他密钥材料）。获取NAF-key之后，清除Ks或拒绝Ks的后续使用。当当前NAF或其他NAF需要新的NAF-keys时，则需要进行新的GBA-Push操作。

注1：生成的NAF-key能用来保护NAF发送给UE的多条推送信息。不同NAFs的NAF-keys可以共存。

使用Disposable-Ks模型，则根3GPP TS 33.220或GBA-Push所建立的NAF-keys不会受到影响。基于GBA\_ME的GBA-Push不会与GBA\_U交互，但是基于GBA\_U的GBA Push会无效UICC中的Ks。

注2：3GPP TS 33.220指明当执行了新的GBA\_U Ks生成流程，则UICC中现有的Ks将被覆盖。GBA-Push之后，ME马上发起新的自启动流程，以防止延时和同步问题。

没有定义将GPI从NAF传递给UE的传递方法。

注3：可能的传递方法有SMS、MMS、SIP MESSAGE、UDP、广播。为了将GPI传递给UEs，NAF需要知道对应于已选传递方法所应使用的消息传递地址。SMS和MMS的传递地址为MSISDN，SIP MESSAGE的传递地址为IMPU，UDP使用的是UDP port - IP-address配对，而广播传递方法的传递地址是和UE相关的公共标识或NAF和UE之间协商的标志。

重发消息是一种用来获取在不可靠信道上（SMS或广播）传输可靠性的标准方法。因此，GBA-Push应允许GPI多次重传，并且可能会出现在每条推送给UE的负载里都包含GPI的情况。

由GPI定义的NAF SA是基于特殊UICC（USIM/ISIM）应用的使用。有时传递方法/地址指示UE使用哪个UICC应用，而在某些场景下传递方法/地址需要明确地发送给UE。如果MSISDN用作传输地址，则使用与MSISDN相关的USIM。因为只有UE中的与MSISDN相关的USIM处于激活状态，SMS才能到达UE。当IMPU用作目的地址，则使用相应的ISIM。对于UDP和广播，使用的USIM/ISIM应用需要在GPI中指示或者在其他频率信道协商一致。

为了保护用户隐私，GPI部分信息需要进行机密性保护，特别是使用广播传输时的NAF标识。对于NAF到UE和UE到NAF消息之间的上行能力，UE到NAF安全的单独SA标识需要由NAF分配并且要包含在GPI受机密性保护的部分中。为了帮助防止DoS攻击产生的严重影响并且阻止有些NAF滥用GBA-Push，需要对GPI进行完整性保护。由于GPI的完整性保护可以检测传输错误，因此这也可以防止UE接受不正确的GBA Push安全协商。由GPI定义的Ks密钥来获取用于机密性保护和完整性保护的密钥。

## 5.2 GBA Push 架构

### 5.2.1 技术说明及基本原理

GBA Push功能是建立在3GPP TS 33.220所提供的安全架构和功能之上的，与3GPP TS 33.220最主要的



区别是在BSF和NAF、NAF和UE之间定义了新的参考点，如图2所示，该图由3GPP TS 33.220的图4.1修改得到。

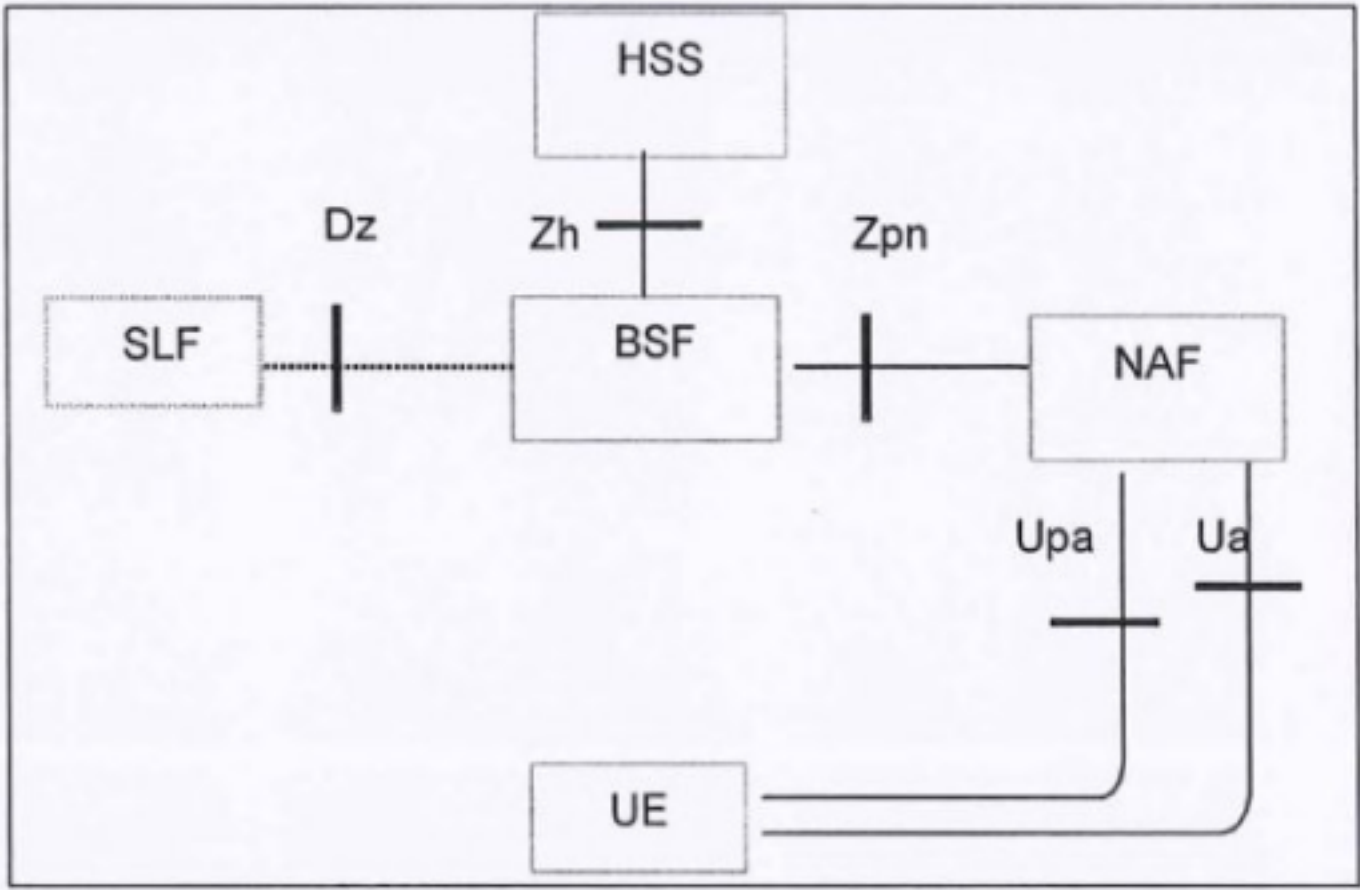


图2 通过NAF完成推送自启动过程的网络模型

图2所示的GBA Push架构基于以下基本原理：

——不能影响 Ua 参考点的安全保护，例如：不管用于安全保护的 GBA 密钥是 UE 发起的还是 NAF 推动过程发起的，都不应改变 Ua 参考点协议。

——从 BSF 的角度看，NAF 还是密钥获取的发起实体，然而这是在 NAF 没有 B-TID 的场景下（而 UE 可能拥有一个有效的 GBA 会话）。基于 3GPP TS 33.220 定义的 Zn 参考点协议，引入了新参考点 Zpn。

——在 NAF 和 UE 之间引入了新的参考点 Upa，通过 Upa 的所有消息都是由网络侧发起的。Upa 定义了 GBA 推送消息。

——NAF 通过 Zpn 参考点接收来自 BSF 的 GBA 推送消息，并将其通过 Upa 参考点发送给 UE。

### 5.2.2 GBA-Push 密钥生成模型

Disposable-Ks模型是应用于GBA-Push中的密钥模型。在该模型中，Ks只能被使用一次来生成一套单独的NAF-keys以及其他一些用来保护GPI传输的密钥材料，见6.3。获取NAF-key后，清除Ks或者不能再使用Ks，这意味着将没有建立好的可用Ks。

如果只是想用GBA-Push建立一个外部NAF-key，那么应总是使用GBA\_ME，这个功能不要求UICC支持GBA\_U。基于GBA\_U的GBA-Push不仅会协商一个内部NAF-key，还会协商一个外部NAF-key，3GPP TS 33.220定义了NAF-keys的获取方式。

在基于GBA\_ME的GBA-Push自启动安全协商中，根据3GPP TS 33.220生成的Ks不会受到影响。

在基于GBA\_U的GBA-Push自启动安全协商中，根据3GPP TS 33.220生成的GBA\_U Ks将会无效。如果在基于GBA\_U的GBA-Push安全协商后有应用要求使用GBA\_U NAF-keys，则需要进行一次通用GBA来建立新的GBA\_U Ks。但是，该应用可以继续使用从该无效的Ks中获取的NAF-keys，例如，原来已经使用NAF-keys的应用不会受到GBA-Push自启动安全协商的影响。

GBA-Push只支持生成在UE和NAF之间共享的NAF SAs，NAF SA包括NAF-key、密钥生命周期、以及其他一些信息，它们在6.2.3小节中定义。

## 5.3 GBA Push 的要求

### 5.3.1 GBA Push 的基本要求

以下这些基本要求应用于GBA Push：



- 网络实体 Push-NAF 应能够安全地触发建立其与 UE 之间的 NAF SA。
  - 在触发 NAF SA 的生成时, Push-NAF 应能够使用存在消息延时的传输通道。
  - Push-NAF 应能够在发给 BSF 的请求消息中使用指示 UE 的公共标识。
  - 如果 GBA Push 使用了公共标识, 则公共标识应对应一个唯一的私有标识。
  - 当只需要基于 ME 的 NAF-keys 时, 例如, Ks 在 ME 中建立, 则应使用基于 ME 的 GBA Push。
- 当 UE 拥有支持 GBA 的 UICC 卡 (GBA\_U), 并且需要基于 UICC 和 ME 的 NAF-keys 或者需要只需要基于 UICC 的 NAF-keys, 例如, Ks 在 UICC 中建立, 则应使用基于 UICC 的 GBA Push。
- 从 Push-NAF 推送给 UE 消息的接收, 触发了 UE 中 NAF SA 的生成。
  - UE 不应需要联系网络实体才能正确生成 NAF SA。
  - 不管 GBA 自启动安全协商是通过 Ub 还是 Upa 参考点, UE 和 NAF 应能够独立地在 Ua 参考点上使用 NAF-keys。

注: 当使用 GBA-Push 机制来创建 UE 和 NAF 之间的 NAF SA 时, 不局限 NAF 将获取的安全协商只应用于网络侧发起的协议; 同样, 当使用 UE 发起的 GBA 时, 不局限 NAF 将获取的安全协商只应用于 UE 发起的协议 (Ua 参考点)。

- 为了避免预配置密钥, 保护 GPI 信息机密性和完整性的密钥生成机制应基于 GBA 原则。
- NAF 不能够获取或生成用于保护 GPI 信息的密钥。

### 5.3.2 HSS 及 HLR 的要求

HSS 和 HLR 的要求应符合 3GPP TS 33.220。

### 5.3.3 BSF 的要求

除了 3GPP TS 33.220 的 4.2.1 小节中的 BSF 要求, 还存在以下要求:

- BSF 应能够查找到与公共标识所对应的私有标识。
- BSF 应能够根据私有标识查找到对应的 Ks。
- BSF 应能够根据新的 Ks 生成 GPI。
- BSF 应对 GPI 进行完整性保护。
- BSF 应对 GPI 中特定域进行机密性保护, 需要机密性保护的域在 6.2.1 小节中介绍。

### 5.3.4 UE 的要求

除了 3GPP TS 33.220 的 4.2.4 小节中的 UE 要求, 还存在以下要求:

- UE 应能够存储并处理 NAF SAs。
- 本标准中的 ME 也应实现 3GPP TS 33.220 中所定义的 GBA\_U 和 GBA\_ME。
- UE 可能执行一种认证授权机制来认证接收到的 GBA Push 消息。

注: GBA Push 消息认证授权机制可以基于 Push-NAFs 的 FQDN 名所对应的白或黑名单列表。

### 5.3.5 参考点 Upa 的要求

参考点 Upa 的要求:

- UE 应能够基于 AKA 来验证 GPI 来自一个被授权 BSF。

注1: 通过 Push-NAF 可以拥有正确的 Ks\_(ext/int)\_NAF, 表明 BSF 已经认证了 NAF, 这样, Push-NAF 就间接地得到了认证。

- UE 应能够确定进行自启动安全协商的 UICC (USIM/ISIM) 应用。
- NAF 和 UE 应能够建立一个共享的 NAF SA。



——NAF 应能够发送 NAF SA 标识信息。

——BSF 应能够将密钥材料的生命周期指示给 UE。BSF 通过 Upa 发送的密钥生命周期应指示密钥的过期时间。

注2: Upa参考点的要求是基于3GPP TS 33.220中所述的Ub参考点的要求。

### 5.3.6 参考点 Zh 的要求

参考点Zh的要求与3GPP TS 33.220中一样。

### 5.3.7 参考点 Zpn 及 Zpn'的要求

参考点Zpn的要求:

——应提供互认证、机密性保护、完整性保护。

——如果 BSF 和 NAF 处于同一个运营商网络,则应用 3GPP TS 33.210 中的 NDS/IP 来保护基于 DIAMETER 协议的参考点 Zpn。

——如果 BSF 和 NAF 处于不同的运营商网络,则应根据 IETF RFC 2246 的 TLS 来保护 Zn-Proxy 和 BSF 之间的基于 DIAMETER 协议的参考点 Zpn'。

注1: 3GPP TS 33.220的附录E指定了需要使用TLS特性。

——基于 Zpn/Zpn'参考点的网络服务应用 IETF RFC 2246 中的 TLS 来保护。

注2: : 3GPP TS 33.220的附录E指定了需要使用TLS特性。

注3: 该要求根据3GPP TS 33.220中的要求修改得到,以便适用于GBA Push。

注4: 由于UE可能无法验证pNAF PQDN,在网络侧的Zpn-proxy中严格检查pNAF FQDN-name显得十分重要。过于简单地检查pNAF FQDN-name,例如,只验证FQDN的一部分,可能会导致pNAFs滥用pNAF FQDN-name的上升。

——BSF 应能够将 NAF 请求的密钥材料发送给 NAF。

——根据 BSF 的策略以及 NAF 通过 Zpn 的请求消息中指示的应用,NAF 应能够从 BSF 获取一组特定应用的 USSs。

——NAF 应能向 BSF 表明其需要申请一个应用还是一些应用的 USSs。

注5: 如果某些应用只需要一个特定应用USS的子集,例如,IMPI,则NAF从来自BSF的完整USS中选择该子集。

——BSF 一个能够以单个 NAF 或单个应用的粒度进行配置。

——私有用户标识,例如,IMPI,可能发送给 NAF。

——特殊的 USS 可能发送给 NAF。

——如果 NAF 向 BSF 请求的 USSs 不在用户的 GUSS 中,但只要满足 BSF 本地策略的条件,则这不应形成错误。BSF 应只向 NAF 发送其所请求的并找到的 USSs。

——可以这样配置本地策略:BSF 可能请求一个或多个特定应用的 USS,这些 USS 出现在特殊请求 NAF 的特殊用户的 GUSS 中;而如果条件不满足,则 BSF 拒绝来自 NAF 的请求。为了满足本地策略,不要求 NAF 通过 Zpn 参考点请求 USSs 信息,这些 USSs 是 BSF 要求出现在 GUSS 中的,更确切地说,BSF 本地检查 USS 的存在性已经足够。也可以这样配置 BSF:对于请求 NAF,没有要求的 USS。

注6: 更多关于本地策略使用的信息可以查看3GPP TS 33.220的附录J。

——NAF 应能够请求 NAF SA 的生命周期。BSF 通过 Zpn 发送的密钥生命周期应指示密钥的过期时间。

注7: 根据NAF的本地策略,这不妨碍NAF在密钥过期时间之前更新NAF SA。



注8: 如果传递给NAF的一个或多个USSs在HSS中的用户GUSS中发生了更新, 则在下一次NAF从BSF请求USS时, BSF将更新的USSs通过Zpn参考点发送给NAF (如果BSF已经通过Zh参考点获取了HSS中更新了的用户GUSS)。

### 5.3.8 Zn-Proxy 的要求

当Push NAF不在归属网络, 则访问NAF应使用NAF网络的Zn-proxy来与用户BSF (归属BSF) 交互。Zn-proxy的要求在3GPP TS 33.220中描述。

### 5.3.9 参考点 Ua 的要求

参考点Ua的要求见3GPP TS 33.220, 另外增加以下要求:

——应可以在上行方向使用 SA 标识, 这些标识与建立 NAF SA 所使用的推送消息没有关系。

### 5.3.10 NAF SA 标识的要求

要求如下:

——下行 NAF SA 标识应在 UE 中唯一并且唯一指示一个对应于特定 NAF\_ID 的 NAF SA。

——上行 NAF SA 标识应在 NAF 中唯一并且唯一指示一个对应于特定 UE 和 Ua 安全协议标识的 NAF SA。

### 5.3.11 参考点 Dz 的要求

该接口位于BSF和SLF之间, 用于获取HSS的地址, Dz参考点应和3GPP TS 33.220中的一样。该接口不要求处于只存在一个单一HSS的环境。

## 5.4 GPL 的要求

### 5.4.1 GPL 会话的概念

可以预见未来将会出现基于Push-NAF的服务, 这些服务依赖于单设备会话的概念, 基于同一个安全协商来推送多条消息将带来很多益处。一个很好的例子是病毒签名服务器, 病毒签名可能通过多条推送消息来传递 (基本推送传递机制的消息尺寸限制原因), 如果为每条消息都建立一个新的安全协商, 这将会十分低效。

这要求GPL除了提供完整性保护之外 (以及可能的机密性保护), 还需要提供抗重放保护。图3所示描述了使用场景, 其中只使用一个安全协商来将三条推送消息从Push-NAF传递给UE。注意步骤 (1) 和步骤 (2) 不属于本标准范围, 步骤 (1) 和步骤 (2) 见3GPP TS 33.222的规定。

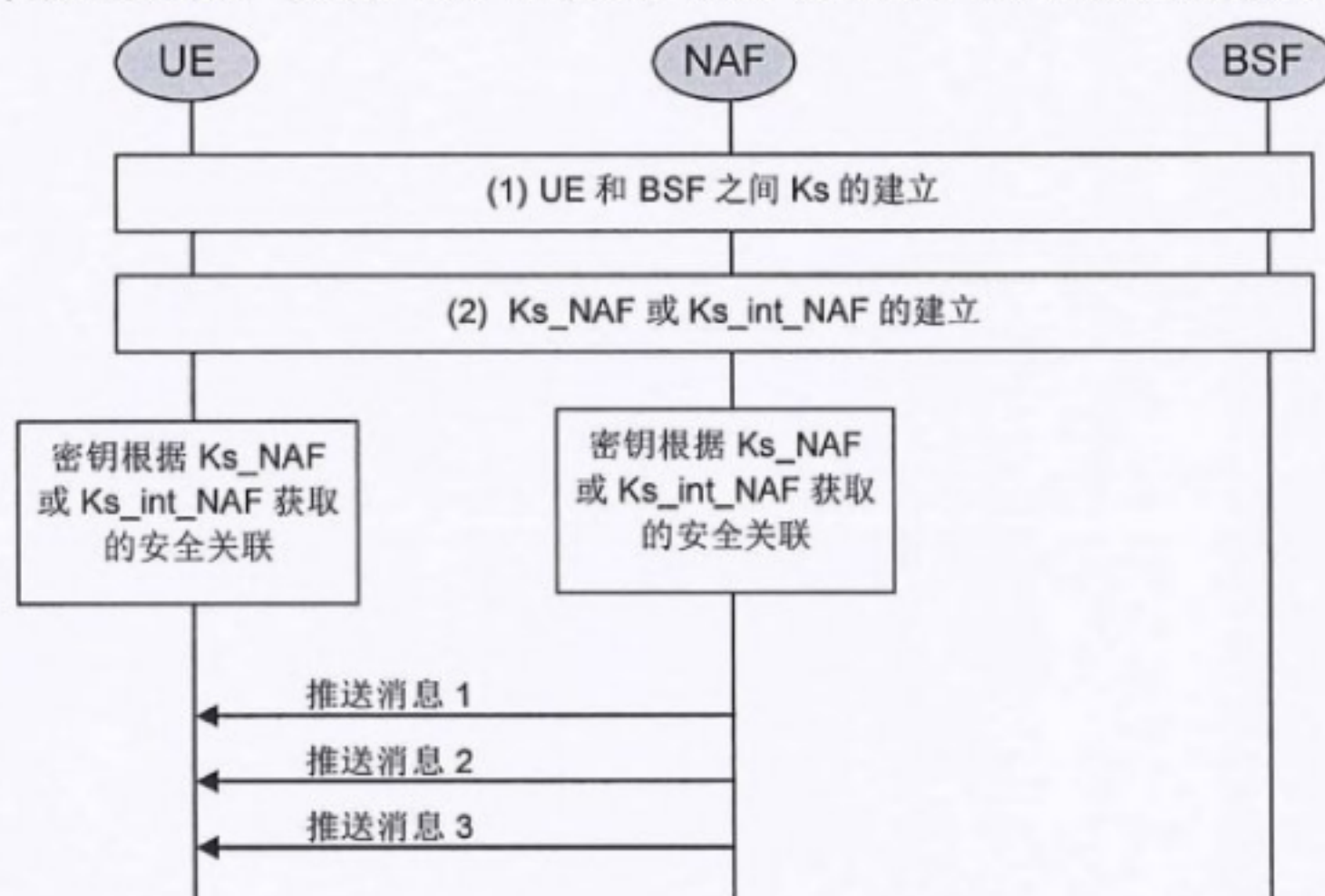


图3 安全会话示例

如果GPL用来提供一个完整的会话概念, 包括通过超时设定/确认和重传机制、会话的重建、消息的



重排序等来实现消息传递的可靠性。那么, GPL将会变得不必要的复杂, 而且GPL消息的尺寸对于某些应用(例如传输的是短消息业务)来说会显得过大。因此, GPL应只需提供足够的会话状态即可, 以确保多条GPL消息的安全性没有受到威胁。GPL应为GPL会话提供安全业务的机密性保护、完整性保护、以及抗重放保护。

如果应用要求一个更复杂的会话概念, 即包括除了安全业务外的其他业务, 例如, WSP应用于GPL之上, 但这不属于本标准研究范围。

尽管有时一个从Push-NAF到UE的安全下行通道已经足够(对于只接收广播业务的UEs), 但也可能需要建立上行通道。一个例子是基于OMA位置的业务, 服务器向终端请求位置信息, 终端返回其所在的位置信息。这样的请求/回复交互可能需要每十分钟重复一次, 这种情况下对上行通道进行安全保护就显得十分明智。上行通道的安全建立可以基于和下行通道一样的NAF SA。

为了将GPL消息发送给UICC, Push-NAF选择一个通向UICC并且被有GPL能力的ME(短消息业务类别2)所支持的传输通道。拥有GPL能力的ME接收GPL\_U消息所用的协议依赖于ME和UICC之间的接口类型(ISO或者HSP)。

3GPP TS 33.222所定义的GBA Push能够使用一个GPI来建立Ks\_int\_NAF和Ks\_int\_NAF。但是GPL没有设计用来完全利用这一特性, 不可能由一个GPI来为GPL\_U和GPL\_ME生成GPL SAs。

#### 5.4.2 GPL 要求

通用安全推送层需要满足以下要求:

- 要求 1: 应对 Push-NAF 到 UE 的通用应用层消息进行封装。
- 要求 2: 应允许基于同一个安全关联来发送多条消息。
- 要求 3: 应可以为消息提供完整性保护和机密性保护, 完整性保护是必须的, 而机密性保护是可选的。
- 要求 4: 应可以检测到同一个会话中的重放消息。
- 要求 5: 如果上行消息出现在应用协议中, 则应可以为这些消息提供同等级的安全保护, 安全保护所需密钥的获取基于 Ks\_NAF 或 Ks\_int\_NAF。
- 要求 6: Push-NAF 应通过选择传输通道的类型来选择 GPL 消息的目标, UICC 或 ME。为了将 GPL 消息发送给 UICC, Push-NAF 选择一个通向 UICC 并且被有 GPL 能力的 ME(短消息业务类别 2)所支持的传输通道。
- 要求 7: 拥有 GPL 能力的 ME 接收 GPL\_U 消息所用的协议依赖于 ME 和 UICC 之间的接口类型:
  - 当 ME 和 UICC 之间的是 ISO 接口时, ME 应支持"ENVELOPE SMS-PP data download" 和"Bearer Independent Protocol in client mode"(用户模式类别 e), 它们在 3GPP TS 31.111 中定义。
  - 当 ME 和 UICC 之间是 TS 102600 所定义的 HSP 接口时, ME 应支持 HSP 上的和 ETSI TS 102 483 定义的"ENVELOPE SMS-PP data download"。

注: 没有必要在ME和UICC之间定义新的接口。

为了使用本标准中的GPL, ME和/或UICC应配置GPL协议实体以实现协议处理功能。另外, 为了使用GPL, 有必要根据3GPP TS 33.222来实现GPI的处理。

GPL协议实体位于ME(GPL\_ME)或UICC(GPL\_U)中。当GPL协议实体位于ME中时, Ks\_NAF应用作ME和Push-NAF之间的共享主密钥; 而当GPL协议实体位于UICC中时, Ks\_int\_NAF应用作ME和



Push-NAF之间的共享主密钥。

Push-NAF应知道ME支持GPL\_ME的能力以及/或者USIM/ISIMs支持GPL\_U的能力（依赖于目标是哪个GPL协议实体）。否则，Push-NAF不知道其能否将GPL消息发给UE或者UE能读懂哪种类型的GPL消息。因此，应在GBA-Push UE注册过程中将ME的GPL能力指示给Push-NAF，GBA-Push UE注册过程在3GPP TS 33.222附录B中定义。USIM/ISIM的GPL\_U能力应存储在HSS的GUSS信息中。

## 6 GBA Push 功能

### 6.1 GBA Push 消息流程及处理

#### 6.1.1 GBA Push 消息流程

图4所示为在NAF需要给UE发送数据而没有有效的NAF-key（如没有可用的Ks(\_int/ext)\_NAF）场景下的消息流程。NAF需要发起建立NAF SA的原因可能是UE没有能力直接与BSF发生GBA自启动安全协商，或者UE不能够直接与BSF发生GBA自启动安全协商。

注：UE没有能力直接与BSF发生GBA自启动安全协商的一个例子是UE处于广播场景下。

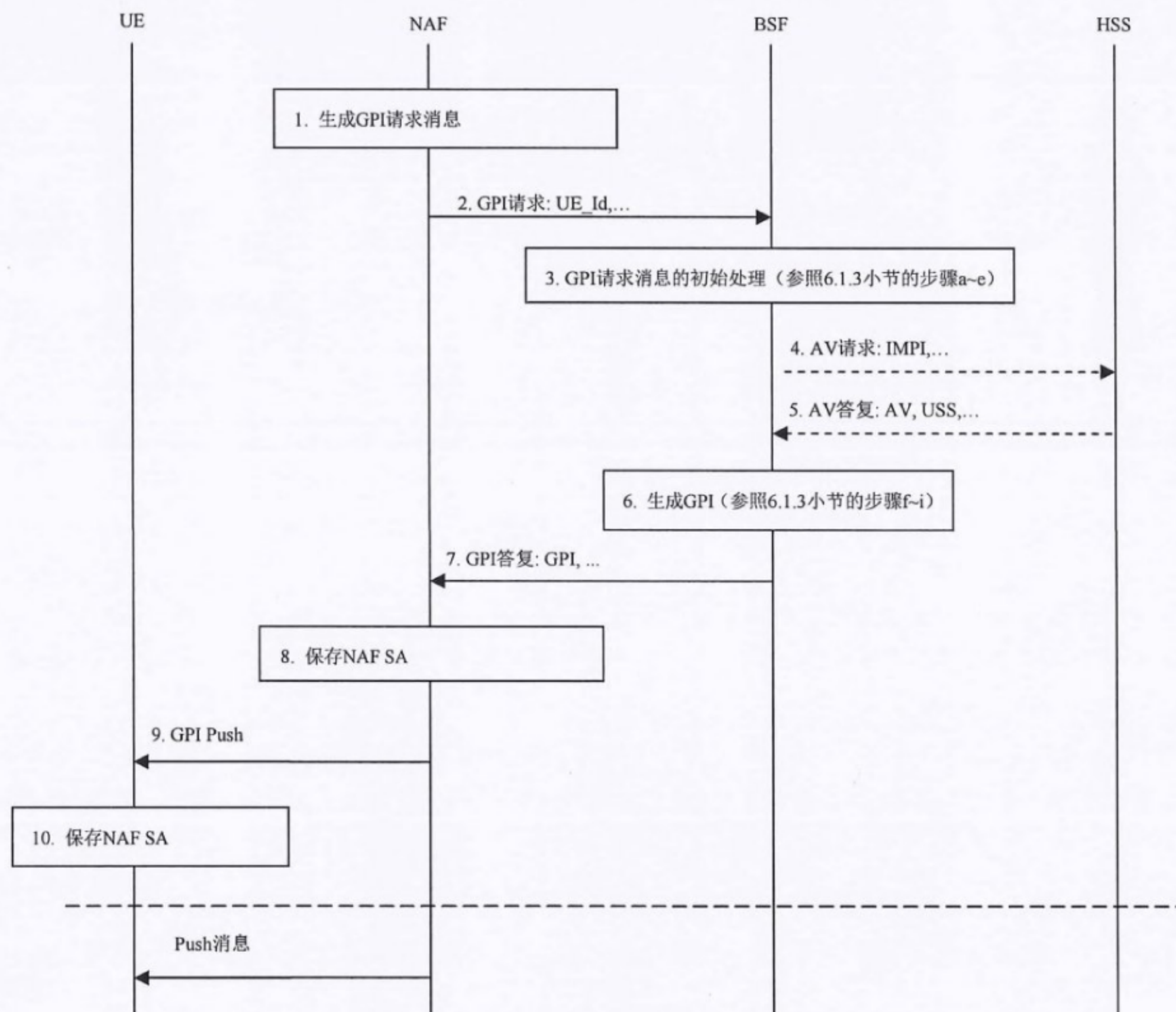


图4 通过NAF进行的GBA Push自启动消息流程

如果用户签约信息由HLR管理而不是HSS，则GUSS功能和SLF功能不可用，除非功能流程不变并将



文字描述和消息流程中的HSS替换为HLR。

在进行GBA Push之前，UE需要为想要进行的业务注册到Push NAF，以共享Push-NAF在进行推送业务时所必需的注册信息以及需要在UE和Push-NAF间协商的信息。在GBA/GBA Push的初始化流程/注册流程中，在UE没有保存NAF要求使用的鉴权的信息情况下，UE需要在请求消息中携带其支持的鉴权类型信息以使得NAF判断自身是否支持UE使用的鉴权类型，支持则通知UE进行相应的GAA类型鉴权，否则向UE返回拒绝消息并结束流程。这样便可以在UE和NAF之间协商双方是否支持使用GBA/GBA Push以及是使用鉴权类型2G GBA还是3G GBA。然后UE和BSF便可以进行鉴权和密钥协商。最后UE向NAF发送应用请求消息，其中携带用户标识UE\_ID、推送传递方法、传输地址UE\_Trp、鉴权类型等信息，NAF收到请求消息后，判断UE使用的鉴权类型是否符合自身要求，若是，与UE执行后续连接建立过程；否则，向UE返回连接拒绝消息，结束本流程。

在完成GBA-Push的UE注册流程后，就可以执行以下GBA Push流程。

处理和消息流程：

- a) NAF与申请推送业务的UE建立共享的NAF SA。Push-NAF知道签约用户的身份标识，执行6.1.2小节中的处理并生成GPI请求。
  - b) NAF向BSF发生GPI请求。
  - c) 根据从NAF接收到的请求，BSF执行6.1.3小节中步骤a~e的处理过程。
  - d) BSF从HSS获取新的AV以及签约用户的GUSS。GUSS包含用户安全相关信息，例如，UICC中GBA信息及USS信元。
  - e) HSS将AV和GUSS发送给BSF。
  - f) BSF接收到来自HSS的AV响应后，执行6.1.3小节中步骤f~i的处理过程。
  - g) BSF发送GPI响应给NAF。
  - h) NAF存储接收到的信息以及其他NAF SA中的用户信息，参见6.2.3小节。
  - i) NAF利用选择的传输机制和给定的传输地址，通过Upa接口将GPI发送给UE。
  - j) 当UE接收到包含GPI的消息，根据6.1.4小节处理GPI并存储相应的NAF SAs。
- 此时，UE和NAF就已经准备好使用建立好的NAF SA了。

### 6.1.2 发起 GPI 请求前的 NAF 处理

NAF读取与用户和应用相关的可用数据，该应用将由建立的NAF SA保护。然后NAF决定该请求中使用的Ua接口安全协议标识，NAF还需决定NAF SA的生命周期。最后NAF生成包含表1中参数的GPI请求消息。

表1 NAF GPI请求中的参数

参数名	描述	注释
UE_Id	UE标识	可以是私有标识也可以是公共标识
UE_Id_Type	指示UE标识是私有标识还是公共标识	BSF需要知道该信息以正确地向HSS/HLR发起从公共标识到私有标识的决议
App_Lbl	UICC应用标识	如果UICC应用可以根据上下文或达成的协议知道，则该变量可以为空
NAF_Id	NAF FQDN和Ua接口安全协议标识的串连	在3GPP TS 33.220中定义
P-TID	NAF SA标识	在UE回复NAF时使用。该标识在GPI消息中并得到机密性保护，参见6.2.1和6.2.2小节
U/M	指示使用 GBA_ME 或者GBA_U的标识	



表1 (续)

参数名	描述	注释
Key_LT	NAF-Key生命周期	
Priv_Id	指示请求用户私有标识	对于UICC应用 (USIM/ISIM), 用户私钥标识为IMSI/IMPI
GSID_List	USS请求信息的GSIDs	

### 6.1.3 BSF 对 NAF GPI 请求的处理

当BSF接收到来自NAF的GPI请求时, 执行以下的处理:

a) BSF检查NAF被授权使用GPI请求中的NAF\_ID, 如果结果为未授权, 则生成一个错误消息并且结束处理流程。BSF检查GPI请求中的Key\_LT低于系统允许的最大值, 如果Key\_LT大于最大值, 则生成一个错误消息并且结束处理流程。

b) 如果UE\_ID是一个公共标识, 则BSF根据3GPP TS 29.109来查找到相应的私有标识(IMPI或IMSI)。

c) 如果需要, BSF使用SLF获取UE所对应的HSS地址。

d) BSF向HSS请求AV以及用户GUSS。

注1: 如果网络侧使用HLR, 则不使用SLF。

注2: 如果网络侧使用HLR, 可以根据3GPP TS 33.220使用外部数据库获取GUSS。

e) BSF检查NAF请求的GBA\_ME还是GBA\_U。如果是GBA\_U, BSF根据GUSS来检查这个NAF请求符合UICC支持GBA这个特性。如果不符合, 则生成错误消息并且结束处理流程。

根据3GPP TS 33.220所述, BSF可能用USS来实现策略控制和密钥选择指示。如果请求的是GBA\_U, BSF查询其数据库以确认私有UE\_ID是否已注册、是否已存在有效的Ks。如果存在有效Ks, 则BSF需要无效该Ks。

如果网络侧使用的是HLR而不是HSS, 则BSF只从HLR请求AV。

注3: 如果网路侧使用HLR, 可以根据3GPP TS 33.220使用外部数据库获取GUSS。

f) BSF根据提供的NAF\_ID生成NAF-keys。

g) BSF生成GPI, GPI的参数信息在6.2.1小节中定义。GPI的生成包括计算GPI MAC值以及对GPI的部分信息进行机密性保护, GPI的保护在6.3节中描述。

h) BSF向NAF发送回复消息并删除使用的Ks。GPI回复消息在表2中定义。

表2 GPI回复消息中的参数信息

参数名	描述	注释
GPI	GPI	GPI 信息在 6.2.1 小节中定义
Ks_NAF / Ks_ext_NAF	外部 NAF-key	Ks_NAF 是基于 GBA-Push 在 GBA_ME 中生成; Ks_ext_NAF 是基于 GBA-Push 在 GBA_U 中生成
Ks_int_NAF	UICC 外部 NAF-key	Ks_int_NAF 是基于 GBA-Push 在 GBA_U 中生成
Key_LT	NAF-Key 生命周期	
UE_Priv_Id	UE_Id 所对应的私有用户标识 (IMSI/IMPI)	只有 NAF 请求了私有标识, 且 GPI 请求中使用了公共标识, 且 NAF 被 BSF 授权接收私有标识, BSF 才返回用户私有标识给 NAF
USS	USS 信息	如果可用

### 6.1.4 UE 对 GPI 的处理

当UE接收到GPI信息时, 执行以下处理步骤:

a) UE接收GPI信息, GPI参数在6.2.1和6.3.5小节中定义。



b) 如果GPI中包含了App\_Lbl, 那么如果App\_Lbl:

- 1) 指示 USIM 或 ISIM 应用已经激活, 则 UE 继续执行步骤 d。
- 2) 指示 USIM 应用不同于当前激活的 USIM 应用, 则 ME 拒绝该请求, 因为同一时刻只允许存在一个 USIM 处于激活状态。
- 3) 指示 ISIM 应用不同于当前激活的 ISIM 应用, 则 ME 不应终止当前的 ISIM 应用, 而应根据 3GPP TS 31.101 激活该 ISIM 应用, 因为 UE 允许同时激活几个 ISIM 应用。

c) 如果GPI中的App\_Lbl没有定义, 则UE可以根据使用的GPI传输通道(SMS、MMS、SIP Message等)或者其他上下文信息来决定该UICC应用。

d) UE检查其之前是否接收到过相同的GPI。

- 1) 如果 GPI 对应于一个已存在的 NAF SA, 则丢弃该 GPI 并终止 GPI 处理流程。
- 2) 如果 GPI 对应于一个不完整的 NAF SA, 则该激活 GPI 所对应的 Ks 并从第 g 步继续执行处理(第 h 步描述了为什么会出现不完整的 NAF SA)。

注1: 为了有效地执行重传输, UE在检查GPI不对应一个已存在的NAF SA后而只唤起一个UICC应用, 这样UE可以从受益。通过对比接收到的三元组(RAND, AUTN(\*), App\_Lbl)和已存在NAF SAs对于的三元组, UE完成该检查。

e) UE读取GPI版本号并选择对应GPI的完整性保护算法和加密算法。如果UE不支持GPI版本号, 则丢弃该GPI并终止GPI处理流程。

f) 如果UICC应用已激活或可以被激活, 则UE通过向UICC发起认证命令来获取Ks。认证命令类型由GPI中指示的U/M-mode决定使用GBA\_ME或GBA\_U。如果认证命令回复失败, 则结束GPI处理流程。

如果U/M指示使用GBA\_U, 则Ks可以在UICC中有效生成, 并且直到下一次使用认证命令来生成新的GBA\_U Ks才会删除该Ks。在GBA-Push流程所关联的NAF SA生成中, ME应限制在UICC中使用Ks来生成NAF-keys。

g) ME获取GPI保护密钥以及其他用来检查GPI完整性和解密GPI加密部分的参数信息, 该过程在6.3节中定义。

h) ME检查GPI消息的完整性。如果检查失败, 执行一下流程:

- 1) 对于 GBA\_ME, 存储获取的 Ks 并标记为不完整, 结束 GPI 处理流程。
- 2) 对于 GBA\_U, 通过认证命令保存 Ks。Ks 标识, 一般为 B-TID (见 3GPP TS 33.220), 设置为 RAND@'undefined', 结束 GPI 处理流程。

i) ME使用GPI版本号中定义的算法和GPI机密性密钥, 解密GPI中的加密部分。

j) UE使用GPI中的NAF\_ID来获取NAF-Key(s)及 Ks(\_int/ext)\_NAF。密钥获取见3GPP TS 33.220。

k) 存储由NAF-key(s)和关联参数组成的NAF SA。

注2: 当使用GBA\_U时, 会生成两个NAF-keys, Ks\_ext\_NAF存储在ME中而Ks\_int\_NAF存储在UICC中。两个密钥都是NAF SA的一部分。

## 6.2 数据对象

### 6.2.1 GBA Push 信息 (GPI)

GPI信息的定义在表3中给出。GPI不包含任何用户标识或传输地址, 因为UE中的GBA处理不需要这些实体。它们只和GPI信息的传输有关。



表3 GPI信息

参数名	描述	注释
Ver	GPI 版本号	版本号的引入允许改变 GPI 帧格式和保护算法
RAND	UMTS AKA 中的 RAND 值	在 3GPP TS 33.102 中定义
AUTN(*)	AUTN 或 AUTN*	在 3GPP TS 33.220 中定义
App_Lbl	使用的 UICC 应用的标识	如果 UICC 应用可以根据上下文或两者间的协商获知,则可以将该变量设为空。应用表在 3GPP TS 31.101 中定义
U/M	使用 GBA_ME 或 GBA_U 的指示	
NAF_Id	NAF FQDN 和 Ua 接口安全协议标识的串连	在 3GPP TS 33.220 中定义, 机密性保护
Key_LT	请求的 NAF-Key 的生命周期	机密性保护
P-TID	NAF SA 标识	在 UE 回复 NAF 时使用, 该标识在 GPI 中受到机密性保护。见 6.2.2 小节
MAC	基于 GPI 的消息认证码	对整个 GPI 消息的完整性保护

### 6.2.2 NAF SA 标识

NAF SA拥有NAF-key(s)并且拥有指示上行和下行参数的唯一标识, 这是为了支持上行和下行保护方法之间的不相关性。

P-TID由NAF分配并在NAF内唯一。

NAF SA标识:

RAND@'naf': UE中的NAF SA标识 (NAF使用)。

P-TID值: NAF中的NAF SA标识 (UE使用)。

注: 'naf'指示字符串naf。

### 6.2.3 NAF SA

相对于UE中的NAF SA, NAF需要维护一些额外的NAF SA信息。需要存储NAF向BSF请求GPI所用的UE标识, 以用来判定回复消息来自哪个UE以及联系同一个UE的SA序列。同时, NAF还要存储传输GPI的目的地址。如果NAF使用重传机制来达到更好的传输可靠性, 还需要存储GPI部分的加密版本, 该GPI部分受到机密性保护。此外, 还需要存储GPI MAC信息。NAF SA的定义见表4。

表4 NAF SA定义

参数名	NAF	UE	描述	注释
UE_Id	m	o	请求中使用的用户标识	
UE_Priv_Id	o	-	对应于UE_Id的用户私有标识(IMSI/IMPI)	
UE_Trp	m	-	传输GPI所用的传输地址	NAF推送GPI给UE时使用的传输地址
RAND	m	m	UMTS AKA中的RAND值	从GPI中获取
AUTN(*)	m	m	AUTN或AUTN*	从GPI中获取
App_Lbl	m	m	UICC应用标识	从GPI或其他暗含协商或信息中获取
NAF_Id	m	m	NAF FQDN和Ua口安全协议标识的串连	
Enc_GPI	m	-	GPI加密部分加上MAC值	
Mac_GPI	m	-	BSF生成基于GPI的MAC值	
UL_SA_Id	m	m	上行NAF SA标识	
DL_SA_Id	m	m	下行NAF SA 标识	
Ks_NAF / Ks_ext_NAF	m	m	外部NAF-key	基于GBA-Push在GBA_ME中生成的Ks_NAF 基于GBA-Push在GBA_U中生成的Ks_ext_NAF
Ks_int_NAF	o	o	UICC内部NAF-key	基于GBA-Push在GBA_U中生成Ks_int_NAF
Key_LT	m	m	接收到的 NAF-Key的生命周期	



### 6.3 GPI 完整性保护及机密性保护

#### 6.3.1 综合考虑

BSF和UE之间的GPI信息需要进行完整性保护和机密性保护。用于保护的密钥材料不允许离开BSF和UE，这意味着除了UE和BSF外，NAF和其他实体不能修改GPI（由于完整性保护）或读取其中的机密性保护部分。

注：NAF\_Id与用户标识/传输地址一起进行明文传输，则在广播网络或没有加密保护的接入网，会增加隐私问题的风险。

#### 6.3.2 密钥材料的生成

用于保护GPI完整性和机密性的密钥材料来源于Ks。GPI版本1中的密钥获取方法使用3GPP TS 33.220附录B3中定义的KDF，只定义了以下NAF\_ID（变量P3）的修改，NAF\_ID使用UTF-8编码方式。所有密钥长度都为128bit，KDF输出的最低128bit有效位作为密钥，定义了一下密钥：

GPI\_INT\_Key: NAF\_ID应为'GPI\_integrity'。

GPI\_ENC\_Key: NAF\_ID 应为'GPI\_confidentiality'。

GPI\_IV:NAF\_ID应为'GPI\_IV'。

注：这样生成IV是合理的，因为这些密钥只会用来保护一条消息。

#### 6.3.3 GPI 完整性保护

GPI完整性保护是强制的。在如6.3.4小节定义的对GPI进行机密性保护之后，再对GPI进行完整性保护并计算MAC值。

GPI版本1中的完整性保护使用基于128bit密钥的HMAC-SHA256-32算法，如FIPS PUB 180-2 (2002)、IETF RFC 2104 (1997)、ISO/IEC 10118-3:2004中所定义。对于GPI MAC的计算如6.2.1节中定义。在计算MAC时，MAC域应置为全零。

#### 6.3.4 GPI 机密性保护

GPI机密性保护是强制的。需要对表3中的GPI信元进行机密性保护。

GPI版本1中的机密性保护使用基于CTR-AES128算法，如NIST Special Publication 800-38A、FIPS PUB 197中所定义。使用的加密密钥是GPI\_ENC\_Key，计数器的初始值T1为GPI\_IV。使用标准的计数增加功能，其 $m=16$ ，参照NIST Special Publication 800-38A中的附录B，例如，T中最低16bit有效位作为计数器，而最高112bit有效位是静态的并且与GPI\_IV的最高112bit有效位相同。

#### 6.3.5 GPI 消息帧格式和编码

GPI消息结构如图5所示，每个域以网络字节顺序（如，高位优先法）进行编码并将最高位设置为“0”。GPI消息的各个域如下：

——Ver (4 bit): GPI 消息版本编码为 4bit 二进制码，本标准中的任何消息都使用版本 1，例如，消息的第一个半字节为 0x1。

——Reserved (3 bit): 为本标准将来新版本所保留的比特位。在传输消息前，需要将这些保留比特位设置为“0”，而接受者应忽略这些比特位。

——Length App\_Lbl: 一个字节，以字节数式指示 App\_Lbl 的长度。

——App\_Lbl (variable length): 以 UTF-8 形式编码的字符串。

——Length NAF\_Id: 一个字节，以字形数式指示 NAF\_Id 的长度。



- NAF\_Id (variable length):对 NAF FQDN 和 5 字节的 Ua 口安全协议标识串连进行 UTF-8 编码。
- Key\_LT: 4 字节, 密钥过期时间与 NTP 时间戳的前四个字节的格式一样 IETF RFC 4330, 描述了从 1900 年 1 月 1 号 0 点开始以来的秒数, 到 2036 年 2 月 7 日该时间值将会溢出。在 IETF RFC 4330 中描述了一种将过期时间扩展到 2104 年的方法, 这种方法应被支持。
- Length PTID: 一个字节, 以字形数式指示 PTID 的长度。
- PTID (variable length): 以 UTF-8 编码的字符串。
- MAC: 4 字节。

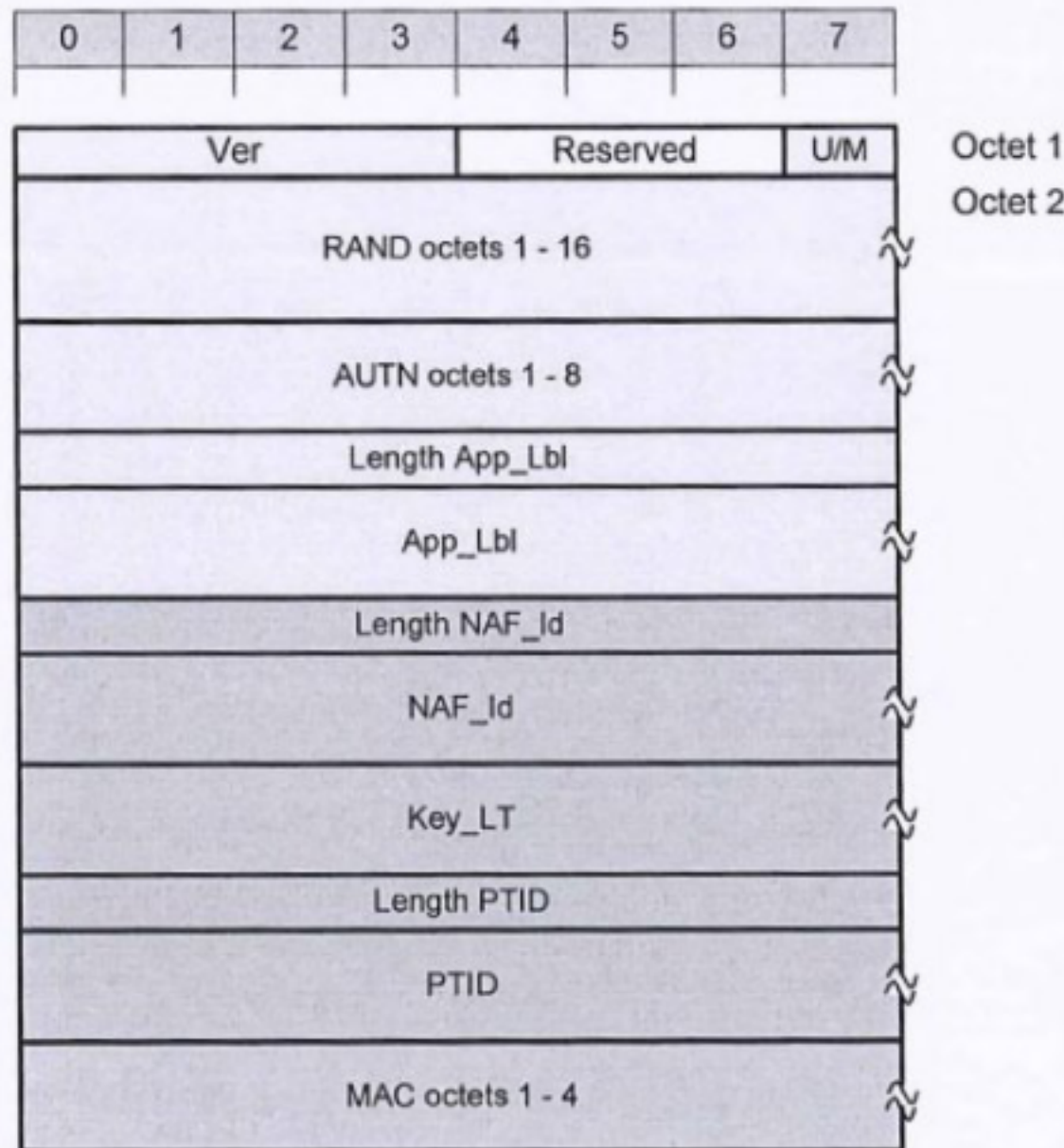


图5 GPI消息结构

#### 6.4 使用 NAFSA 的流程

终端可以利用协商好的NAF SA来建立Ua接口的通信。

如果终端想基于3GPP TS 33.222建立的NAF SA来创建3GPP TS 33.222所定义的Ua口连接, 则该终端需要遵循3GPP TS 33.220的4.5.3小节和3GPP TS 33.222的5.3及5.4所定义的原则, 但还需要做如下修改:

- 不使用 3GPP TS 33.220 的 4.5.3 小节所描述的 B-TID 用来指示 SA (NAF-Key), UE 使用 P-TID。P-TID 包含在 NAF 发给 UE 的 GPI 消息中, 并唯一指示 SA 及用户标识。
- 不使用 3GPP TS 33.222 的 5.3 节所描述的 B-TID 来作为用户名, UE 使用 P-TID。NAF 据此获取用户标识并从 NAF SA 中获取密钥。
- 不使用3GPP TS 33.222的5.4节所描述的B-TID作为PSK标识, UE使用P-TID。NAF据此获取用户标识并从NAF SA中获取密钥。

### 7 GPL 处理功能

#### 7.1 处理模型

在GPL\_ME模式下, GPL协议实体位于传输机构(可以是SMS、IP、IP/UDP)和应用之间, 或者位于应用内。

在GPL\_U模式下, GPL\_U协议实体位于目标USIM或ISIM之内。

当接收到用GPL保护的消息, 接收者将消息转发给GPL协议实体。接收者如何知道消息是GPL消息取



决于各个传输机构的定义，例如，可以通过一个标记在消息上的特殊应用标识，这样就需要定义一个GPL应用标识。

在GPL\_ME中，GPL消息处理完成之后，消息再次被传送到传输机构。此时，已经移除了GPL应用标识和GPL相关信息，只保留了正常的业务数据消息，并利用传输层的正常调度机制将消息路由到目标应用。GPL\_ME的处理模型在图6所示中描述。

在GPL\_U模式下，受保护的GPL消息被传送给目标USIM或ISIM并由它们进行处理。应用数据将保留在USIM或ISIM中，并由所定义的应用对数据做进一步处理。

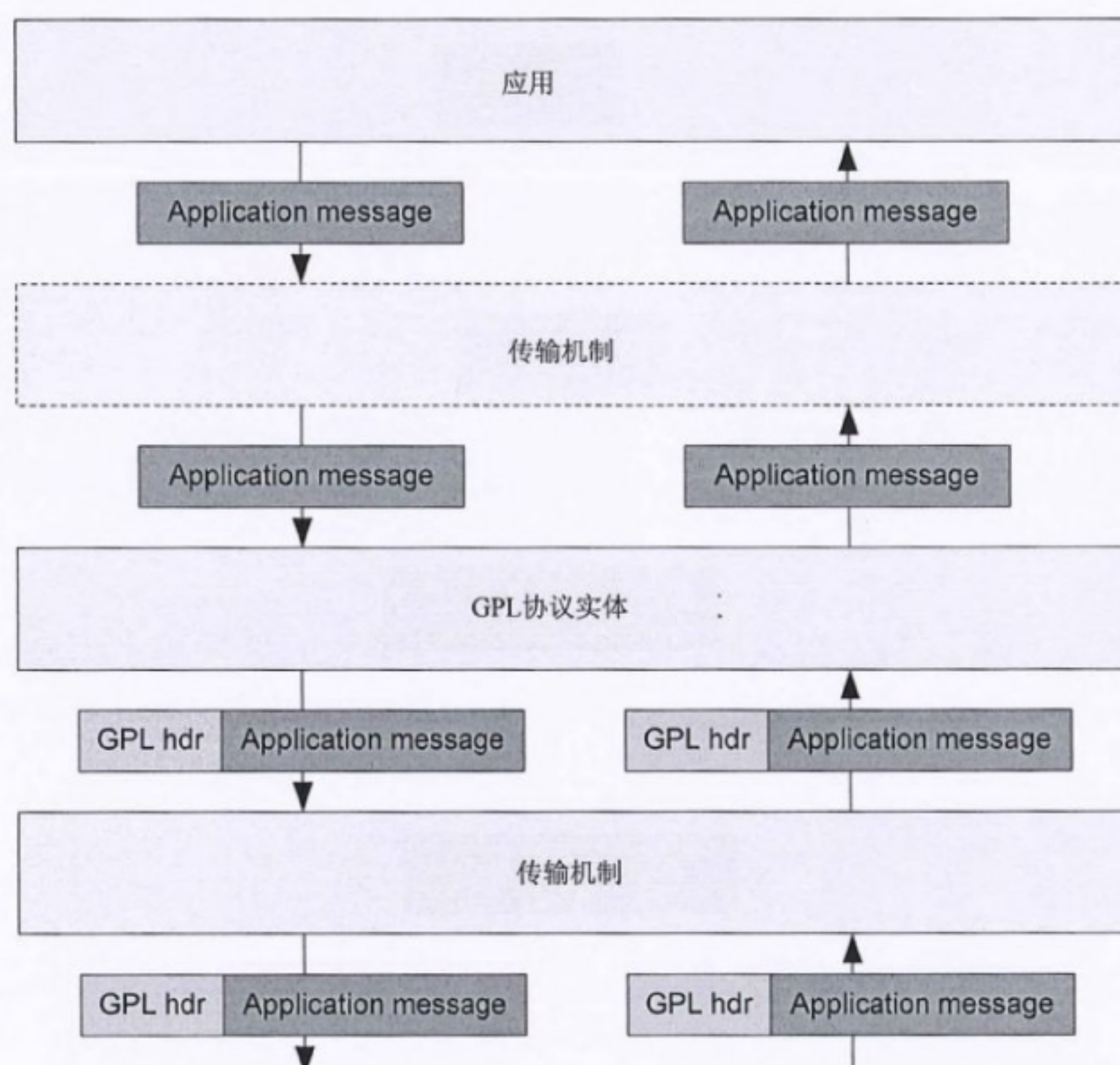


图6 GPL\_ME未知应用的处理模型，右边是接收处理，左边是外发处理

GPL\_ME协议实体逻辑地位于传输结构和应用之间的场景如图6所示。这种场景下，应用不需要知道GPL，任何应用都可以不加修改地使用GPL。另一个场景就是应用感知到GPL，原因可能是应用需要通知GPL协议实体使用哪个安全协商，例如，应用直接调用GPL协议实体，GPL协议实体可能将GPL封装消息传送给传输机构或者返回给应用。

## 7.2 会话开始

当相应的GPL安全关联（GPL-SA）完成初始化，就可以认为GPL协议实体中的GPL会话已经开始，见7.9。对于Push-NAF，这意味着只要NAF接收到来自NAF的GPI并且配置了NAF SA及GPL-SA，则可以认为会话已经开始。对于UE，当其接收到GPI消息并且配置了GPL-SA时，则可以认为会话已经开始。

除了GPI，GPL协议实体需要获取该会话的GPL策略信息，例如，使用哪个加密和完整性保护算法。策略信息可能由应用本身或其他某些管理实体决定。

Push-NAF应选择下行消息使用的策略并且将策略包含在GPL消息中。Push-NAF应为下行消息选择加密套组，UE应为上行GPL消息（如果存在上行消息）选择加密套组。推荐UE为上行消息选择的加密套组



与Push-NAF为下行消息选择的加密套组一样。

### 7.3 会话终止

会话不会明确地终止，例如，没有专门用于关闭会话的GPL消息。UE和NAF都保存了NAF-key的生命周期，一旦生命周期截止，则GPL会话结束并且应删除NAF SA、对应的下行GPL-SA、对应的上行GPL-SA（如果存在）。

### 7.4 GPL 安全协商

GPL安全协商（GPL-SA）是处理进入或外发GPL消息所需要的数据。在存在双向通信链路的场景下，每个实体都需要维护两个GPL-SAs，一个用于接收数据业务，一个用于外发数据业务。

GPL-SA从NAF SA获取，它们共享生命周期和ID标识（更多NAF SAs信息可参考3GPP TS 33.222），如果相应的NAF SA被删除则应删除GPL-SA，反过来也一样。

GPL-SA应与标识SAID相关联，以允许独立地索引Push-NAF和UE中的GPL-SA。上行/下行GPL-SA通常由SAID、Push-NAF的NAF\_ID、及方向来标识。

GPL-SA应至少包括以下条目：

——SAID：用于区分同一个 Push-NAF 或 UE 中的 GPL-SAs，上行/下行 SAID 等同于相应的上行/下行 NAF SA 标识。

——Master key：256bit 的主密钥用来获取完整性保护密钥和加密密钥，主密钥来自相应的 NAF-key。

——SN<sub>h</sub>：拥有有效 MAC 值的 GPL 消息中的最大序列号，用于重放保护。该状态变量只用于进入 GPL-SA。

——SN<sub>s</sub>：用于为外发消息生成序列号的计算器。每发一条消息计时器加“1”，该状态变量只用于外发 GPL-SA。

——Cipher suite：用于保护消息的加密套，加密套由完整性保护算法、加密算法、及密钥生成算法组成。

——GPL-SA life time：GPL-SA 的过期时间，GPL-SA 的生命周期和帧格式应等同于 NAF-key 的生命周期。

### 7.5 联合传输

GPI消息可以通过GPL消息发送，也可以单独发送。当GPI消息包含在GPL消息中发送时，称之为联合传输。

注：3GPP TS 33.222中的GBA-Push允许重传多次GPI消息，包括每次推送一个数据包给UE的场景。为了有效执行重传，3GPP TS 33.222定义了一种UE只有在检查出该GPI不对应于一个已存在NAF SA情况下，才唤醒USIM/ISIM的机制。

### 7.6 消息格式

#### 7.6.1 数据单元传输帧格式

GPL消息结构如图7所示，GPL消息在GPL负载中封装并保护应用消息。

每个域按照网络字节顺序编码（例如，高位优先法），最高有效位为“0”比特位。消息域如下：

——Ver (4 bit)：编码为整数的 GPL 协议版本。遵循本规范的所有消息的版本都应为“1”，例如，消息的第一个半字节应为 0x1。

——GPI Indication (1 bit)：指示是否使用了联合传输，即 GPI 消息是否出现在 GPL 消息中。当 GPI 指示为“0”，则不会出现 GPI 长度域和 GPI 消息域。当 GPI 指示为“1”，则出现 GPI 长度域和 GPI 消息域。



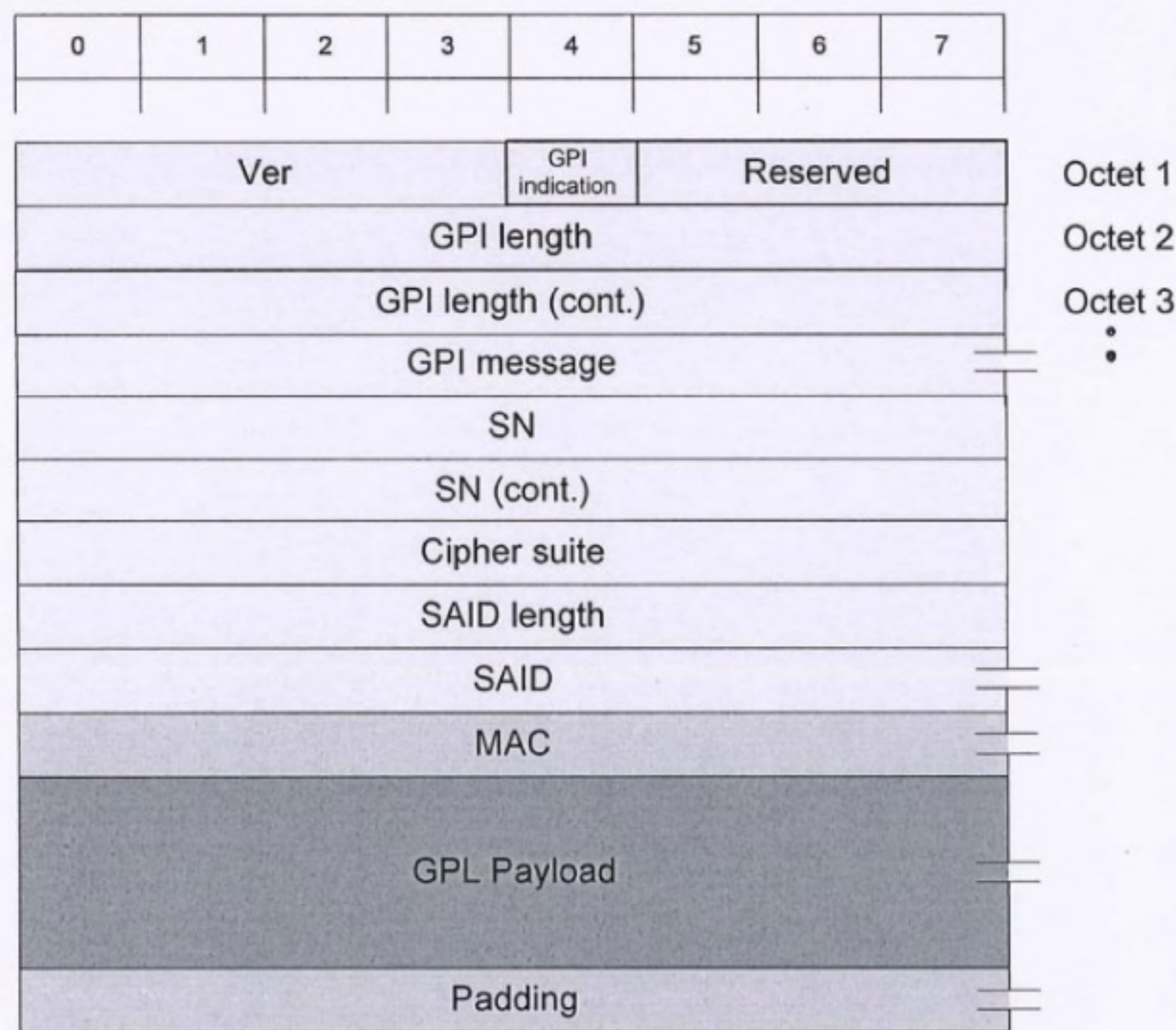


图7 GPL消息帧格式

——Reserved (3 bit): 为本标准将来新版本所保留的比特位。在传输消息前，需要将这些保留比特位设置为“0”，而接受者应忽略这些比特位。

——GPI length (16 bit): 以字节数形式指示 GPI 消息长度，该域只有在 GPI 指示为“1”时才出现。

——GPI message (可变长度): GPI 消息，该域只有在 GPI 指示为“1”时才出现。

——SN (16 bit): 用于同步加密及提供抗重放保护的序列号。

——Cipher suite (8 bit): 用于保护消息的加密套组。加密套组由完整性保护算法、加密算法、及密钥获取算法组成。

——SAID length (8 bit): 以字节数形式指示的 SAID 长度。

——SAID (可变长度): 用于保护消息的 GPL 安全协商的标识。

——MAC (可变长度): 为 GPL 消息提供完整性保护的消息认证码，该域的长度由完整性保护算法的输出大小决定，但应是 8bit 的整数倍。

——GPL Payload (可变长度): 受保护的真正应用消息。消息长度为 8bit 的整数倍，否则，应用需要填充为 8bit 的整数倍。这样的填充由应用决定，不在本规范讨论范围之内。该域需要进行加密保护。

——Padding (可变长度): 加密转换要求填充，填充物具体如何生成、验证、及删除由各个加密转换定义。如果加密转换不要求填充，则该域不出现。该域需要进行加密保护。

域Ver、GPI length、GPI message、Cipher suite、SAID length、及SAID在GPL-SA生命周期内是固定的，并且各个消息都一样。而GPI length 与GPI message域可能出现在某些消息中也可能不在某些消息中，这取决于GPI Indication域。

7.7 接收处理

在处理接收到的GPL消息前，GPL协议实体启动7.9节中所述的GPL-SA。在联合传输情况下，该步骤应在GPI消息处理（步骤b）完成后再执行。



当GPL消息到达接收者的GPL协议实体,应执行以下处理步骤:

- a) 验证 Ver 域为“1”。如果不是,则应丢弃 GPL 消息并结束处理流程。
- b) 如果 GPI 指示没有指示联合传输,则跳到步骤 d。否则,按照以下方法处理 GPI 消息:
  - 在 GPL 消息发给 ME 的场景下,GPL\_ME 根据 3GPP TS 33.222 所定义的方法处理 GPI 消息。
  - 在 GPL 消息发给 UICC 的场景下,GPL\_U 协议实体检查 GPI 是否对应于一个已存在的 NAF SA。如果不是,GPL\_U 协议实体根据 3GPP TS 33.222,从 GPI 消息获取 Ks\_ext/int\_NAF,创建 NAF SA,并且存储与 Ks\_int\_NAF 关联的 NAF SA。密钥 Ks\_ext\_NAF 不发送到 UICC 之外。

注:3GPP TS 33.222中的GBA-Push允许重传多次GPI消息,包括每次推送一个数据包给UE的场景。为了有效执行重传,3GPP TS 33.222定义了一种UE只有在检查出该GPI不对应于一个已存在NAF SA情况下,才唤醒USIM/ISIM的机制。

c) 获取对应于 GPL 消息头中 SAID 的 GPL-SA。如果没有找到与 SAID 对应的 GPL-SA,则丢弃该消息并终止处理流程。

d) 如果 GPL-SA 中没有设置加密套组变量,则验证 GPL 消息中的加密套组被支持,并将加密套组变量设置为 GPL 消息中的加密套组。如果 GPL-SA 中设置了该加密套组变量,则验证该变量与 GPL 消息中的加密套组域相同。如果 GPL 消息中的加密套组不被支持或者该域值与变量值不同,则应丢弃该 GPL 消息并终止处理流程。

e) 验证之前没有收到过 SN 域中的序列号。一种方法是验证 SN 域中的序列号大于当前接收到的最大序列号 SN\_h。如果不是这种情况,则丢弃该消息并终止处理流程。当 SN\_h 等于 0xffff 时,应丢弃所有携带该 SAID 的消息并终止处理流程。不强制执行该抗重放机制(抗重放机制抵触消息的重排序),但是接收者的 GPL 协议实体应验证 SN 域中的序列号没有在之前的有效消息中收到过。

f) 用加密套组指示的完整性保护算法计算 MAC 值。MAC 的计算是对整条 GPL 消息,并且需要在计算 MAC 时将 MAC 域设置为“0”。计算得到 MAC 后,对比 MAC 域中携带的 MAC 值,如果两者不同,则应丢弃所有携带该 SAID 的消息并终止处理流程。

g) 更新抗重放保护状态。如果使用了第 e 步中的机制,则状态变量 SN\_h 设置为 SN 域中的值。

h) 用 GPL-SA 定义的解密算法解密应用消息,解密转换用于 GPL 消息中的 GPL 负载和填充域。

i) 在 GPL\_ME 场景下,将 GPL 消息的应用消息(也就是移除 GPL 消息头和可能的填充域后的 GPL 消息)返回给对应的传输机构。在 GPL\_U 场景下,GPL 消息负载仍然保留在该应用中。

如果在整个处理流程完成之前,GPL协议实体终止了处理流程,则可能会从GPL协议实体返回一个错误指示。

## 7.8 外发处理

在处理任何外发GPL消息前,GPL协议实体启动7.9节中所述的GPL-SA。当应用消息到达发送者的GPL协议实体中后,应执行以下的处理流程:

- a) 如果变量 SN\_s 等于 0xffff,则 GPL 协议实体终止处理流程并返回一个处理指示。
- b) 在联合传输场景下,将 GPI 信息包含在 GPI 消息域中,并相应地设置 GPI 指示和 GPI 长度域。
- c) 根据 GPL 协议实体的唤起者所指示的 SAID,获取对应的 GPL-SA。如果没有找到对应的 GPL-SA,则终止该处理流程。在联合传输场景下,GPL-SA 需要根据相应 GPI 的 NAF SA 获取。
- d) 将 Ver 域设置为“1”,根据 GPL-SA 来设置加密套组域、SAID 域、及 SAID 长度域。将 SN 域设置为状态变量 SN\_s 值。



e) 用 GPL-SA 定义的加密算法加密应用消息, 如果加密算法需要, 应在加密之前对消息进行填充。

f) 将 GPL 消息头中的 MAC 域值设置为“0”, 使用 GPL-SA 定义的完整性保护算法对整条 GPL 消息进行 MAC 计算, 然后将得到的 MAC 值复制到 GPL 消息头的 MAC 域中。

g) 状态变量 SN<sub>s</sub> 加“1”。

h) 将受保护的 GPL 消息传送到处理链中的下一步。

如果在整个处理流程完成之前, GPL 协议实体终止了处理流程, 则可能会从 GPL 协议实体返回一个错误指示。

## 7.9 GPL-SA 初始化

### 7.9.1 概述

GPL SA 从相应的 NAF SA 获取。NAF SA (用于 GPL) 应与一个下行 GPL-SA 相关联, 可能和一个上行 GPL-SA 相关联。只允许由一个 NAF SA 在每个方向 (上下行) 初始化一个 GPL-SA, 如果不这样做则会导致重用对应同一序列号的同一密钥。NAF SA 在 3GPP TS 33.222 中定义, 从 NAF SA 中复制过来的 GPL-SA 域将保持原有的定义。

### 7.9.2 由 NAF SA 初始化下行 GPL-SA

在发送 GPL 消息给 UE 前, Push-NAF 应由相应的 NAF SA 来初始化下行 GPL-SA。Push-NAF 应:

- 将 GPL-SA SAID 设置为 NAF SA 的 DL\_SA\_Id。
- 根据 NAF SA 将 GPL-SA 的主密钥设置为 Ks\_NAF 或 Ks\_int\_NAF。
- 将 GPL-SA 的 SN<sub>s</sub> 设置为“1”。
- 将 GPL-SA 的生命周期设置为 NAF SA 的生命周期。
- 根据应用策略设置加密套组及密钥指示 ID。

UE 应在 NAF SA 建立之后 (例如, 处理完 GPI 信息之后), 从该 NAF SA 初始化一个下行 GPL-SA, UE 执行:

- 将 GPL-SA SAID 设置为 NAF SA 的 DL\_SA\_Id。
- 根据 NAF SA 将 GPL-SA 的主密钥设置为 Ks\_NAF 或 Ks\_int\_NAF。
- 将 GPL-SA 的 SN<sub>h</sub> 设置为“0”。
- 将 GPL-SA 的生命周期设置为 NAF SA 的生命周期。

### 7.9.3 由 NAF SA 初始化上行 GPL-SA

如果应用需要上行 GPL-SA, 在处理第一条来自 UE 的 GPL 消息前, Push-NAF 应由相应的 NAF SA 来初始化上行 GPL-SA。Push-NAF 应:

- 将 GPL-SA SAID 设置为 NAF SA 的 UL\_SA\_Id。
- 根据 NAF SA 将 GPL-SA 的主密钥设置为 Ks\_NAF 或 Ks\_int\_NAF。
- 将 GPL-SA 的 SN<sub>h</sub> 设置为“0”。
- 将 GPL-SA 的生命周期设置为 NAF SA 的生命周期。

UE 应在 NAF SA 建立之后 (例如, 处理完 GPI 信息之后) 并在向 Push-NAF 发送第一条上行 GPL 消息之前, 从该 NAF SA 初始化一个下行 GPL-SA, UE 执行:

- 将 GPL-SA SAID 设置为 NAF SA 的 UL\_SA\_Id。
- 根据 NAF SA 将 GPL-SA 的主密钥设置为 Ks\_NAF 或 Ks\_int\_NAF。



- 将 GPL-SA 的 SN<sub>s</sub> 设置为“1”。
- 将 GPL-SA 的生命周期设置为 NAF SA 的生命周期。
- 根据应用策略设置加密套组及密钥指示 ID。

## 7.10 加密套组

GPL定义了以下加密套组：

### a) 加密套组1：

- ID: 0x01。
- 加密算法：在NIST Special Publication 800-38A、FIPS PUB 197中定义的CTR-AES128。根据NIST Special Publication 800-38A中的附录B，使用m=16的标准增量功能，即T中最低16bit有效位作为计数器，而112bit最高有效位静止。128bit的初始计数器T1组成为：96bit的最低有效位RAND值，串连16bit的序列号SN，串连16bit的填充位“0”，即：T1 = RAND || SN || 0x0000。
- 完整性保护算法：在FIPS PUB 180-2 (2002)、IETF RFC 2104 (1997)、ISO/IEC 10118-3:2004中定义的HMAC-SHA256-32，MAC域长度为32bit。
- 密钥获取功能：根据3GPP TS 33.220附录B定义，KDF的输入为256bit的主密钥及字符串S，S定义为：

- FC=0x40
- P0=密钥目的的字符串
- L0=密钥目的的字符串长度，以16bit整形表示
- P1=方向指示
- L1=方向指示的长度（即，0x00 0x01）
- P2=加密套组 ID
- L2=加密套组 ID 的长度（即，0x00 0x01）

FC数字空间的使用遵循3GPP TS 33.220，为本规范分配的FC值范围为0x40 – 0x48。

对于加密密钥，密钥目的的字符串应为“gba-push-enc”；而对于完整性保护密钥，密钥目的的字符串应为“gba-push-int”。KDF输出的128bit最低有效位作为密钥。

在双向GPL使用场景下，需要两对GPL-SAs，每个方向一对，根据GPL-SA对设定方向指示。对于用GPL-SA保护从Push-NAF到UE的消息，方向指示应为0x00；对于用GPL-SA保护从UE到Push-NAF的消息（如果应用需要这样的SA），则方向指示应为0x01。

### b) 加密套组2：

- ID: 0x02。
- 加密套组2应和加密套组1一样，除了应将完整性保护算法定义为：
- 完整性保护算法：在FIPS PUB 180-2 (2002)、IETF RFC 2104 (1997)、ISO/IEC 10118-3:2004中定义的HMAC-SHA256-64，MAC域长度为64bit。



## 附录 A

## (资料性附录)

## 选择 Disposable-Ks 模型的原因

GBA-Push 使用的是 Disposable-Ks 模型,只使用一次 Ks 来生成 NAF-keys。这意味着在获取 NAF-keys 之后,应删除该 Ks 或禁止再使用 Ks。此外,基于 GBA\_U 的 GBA\_Push 使用 Single Ks 模型,也就是某一时刻只能存在单个 GBA\_U Ks。而使用 Single Ks 模型的原因是为了重用 Rel-6 UICC 的 GBA\_U 的功能。

对于基于 GBA\_ME 的 GBA-Push,本标准假设 ME 能够执行必要的操作,而不需要删除 GBA\_ME 自启动过程生成的 Ks。

采用 Disposable-Ks 模型的原因是为了避免同步的问题,因为 GBA-Push 可能位于一个不可靠的信道上,这会导致不成功的传输或者不确定的时延。对于只存在单个 GBA-Push Ks,这将致使 UE 和 BSF 不同步。另一种出现 UE 和 BSF 不同步的情况是,BSF 执行了正常的自启动过程,而 NAF 几乎同时发起了一次 GBA-Push,这使得 NAF 在 UE 执行 bootstrapping 之前向 BSF 请求 GPI 以及 NAF 在正常的 bootstrapping 之后向 UE 发送 GBA-Push 消息。而 Disposable-Ks 解决了大部分的这些不同步问题。

还存在一种即使采用了 Disposable-Ks 模型也会出现不同步问题的情况,即由于 UE 无法验证 GBA\_Push 消息,致使 BSF 删除了 Ks 而 UE 却保存了 Ks。如果 UE 尝试使用这样的 Ks,就会导致错误的情况。但可以比较容易地解决这种错误情况,因为 NAF 会从 BSF 获取一条错误消息告知该 Ks(由 B-TID 指示)不可用。然后 NAF 返回这条错误消息,而终端将执行一次新的 bootstrap。

该已选择的密钥处理模型的备选方案都是基于允许存在一个或多个由 GBA-Push 生成的 Ks,这样便在 UE 或 UICC 上维护了一组安全上下文。维护一个或多个由 GBA-Push 生成的 Ks 可能使得不同步问题消除或至少变得更小。缺点当然是,由于基于 GBA\_U 的 GBA-Push 是必要的,采用那些模型将会要求 UICC 上实现新的功能,这被认为使得 GBA-Push 的引入和采用变得更加困难。当考虑到已选择的密钥处理模型的微小功能缺点,其他模型所引入的额外开销和复杂度不被认为是引入新 UICCs 的充分理由。



## 附录 B

## (规范性附录)

## GBA-Push 的 UE 注册流程

为了能够使用基于 GBA Push 的业务, 用户和业务提供商需要共享信息, 这通过注册流程来完成。注册流程可能是明确的并包含了用户; 或者也可能是自动的, 依赖于用户运营商提供的用户信息。如果注册流程是运营商发起的, 则运营商将使用所有需要的注册信息。

注: 如果用户使用公共标识注册, 特别是如果 NAF 是第三方服务提供商, 则可能不会是这种情况。一种缓解这种问题的方法是让用户在用标准 GBA 建立的已认证的或安全的连接上执行注册流程, 然后 BSF 就可以将所有需要的信息提供给 NAF。但请注意该功能没有被标准化, 而且所有必要的信息可能不能在现有标准化的接口上使用。

Push NAF 在注册时应当记录用户标识 (UE\_Id)、推送传递方法、及关联传输地址 (UE\_Trp)。用户标识可能是一个公共标识或者私有标识。

公共 IMS 用户标识 (IMPU) 只有在其映射到一个特定的私有标识 (IMPI) 时才被使用。这应当在注册流程中进行检查, 因为如果条件不满足, 则业务将会失败。

当 UE 标识是 MSISDN 时, 则该公共标识将映射到一个特定的 IMSI 但号码可携, 无论如何都需要关联运营商的激活信息来确定用户属于哪个 BSF (也就是用户与哪个运营商签约)。知道了运营商将使得 NAF 能够获取运营商网络中 BSF 的 FQDN。

如果注册的 UE 配备了一张拥有多个 UICC 应用的 UICC 且这些应用能执行 AKA, 而且如果使用的 UICC 应用不是唯一地由 UE 传输方法和/或 UE\_Id 决定的, 则注册过程应当决定使用哪个 UICC 应用以及 NAF 如何联系相应的 BSF (需要知道 BSF 的 FQDN)。如果需要明确的信号来确定使用的 USIM/ISIM, 则应同意使用 App\_Lbl 并做记录。

注册过程中, 想要使用 GPL 功能的 Push NAF 应当记录 ME 是否支持 GPL。如果 Push NAF 想要使用 GPL\_U, 那么 Push NAF 也应当记录 ME 支持的到达 UICC 的传输通道



## 附录 C

### (资料性附录)

### GPL 使用场景

本附录描述了 GBA Push 和 GPL 相结合后所存在的使用场景。因此有些信息描述了 GBA Push 的特性，因为 GBA Push 是用来为 GPL 建立安全关联的。

#### C.1 通用推送层-没有返回通道的终端

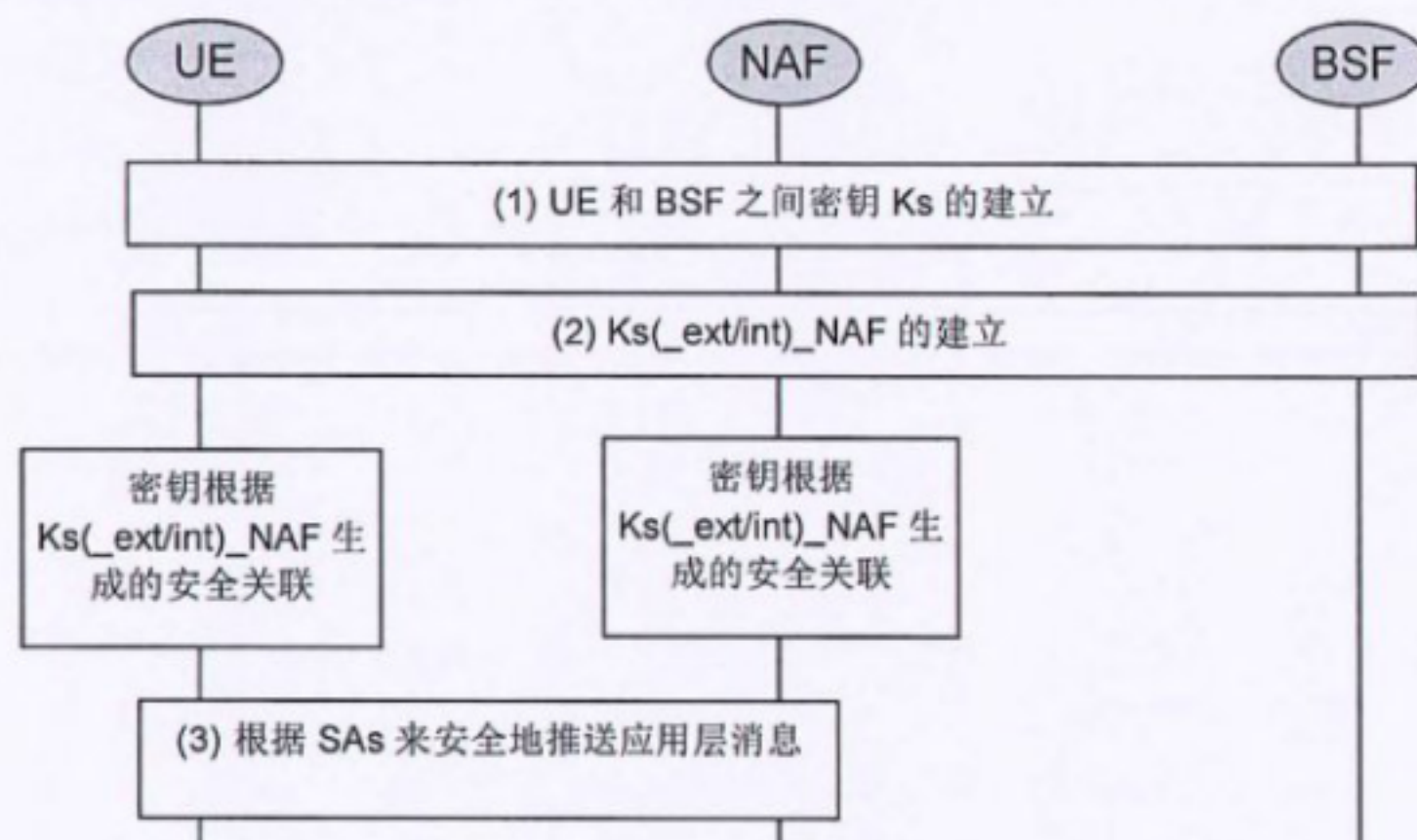
本节描述了一种使用场景，即一个应用如何使用 GBA Push 和 GPL。

应用的目标是能够将消息安全地从应用服务器（在 Push NAF 中实现）推送给 UE。例如，Push-NAF 推送给 UE 的消息包含最新的病毒签名。没有返回通道的终端可能对这种场景感兴趣，例如，纯粹接收广播消息的终端。

该功能分为两个阶段：Push-NAF 和 UE 之间建立安全关联，以及消息的保护。安全关联包含根据  $Ks\_ext/int\_NAF$  获取的密钥。这两个阶段如图 C.1 所示中描述，其中第一阶段包括步骤（1）和（2），第二阶段包括步骤（3）。阶段 1 使用 GBA Push 建立了 Push-NAF 和 UE 之间的安全关联；阶段 2 使用 GPL。

安全关联的建立归结于  $Ks$  的建立，然后是  $Ks\_ext/int\_NAF$  的建立。

对于保护推送的实际消息，存在两种选择：或者消息推送是一次性事件，或者引入会话的概念。会话指的是在 Push-NAF 和 UE 之间建立安全的单向通信信道。



图C.1 将消息安全地从应用服务器推送给UE所包括的两个阶段

#### C.2 特殊使用场景

##### C.2.1 网络侧发起的 NAF-Key 更新和密钥分发

与该使用场景相关的有：将密钥分发给接收广播消息并需要密钥来解密消息内容的终端；为现存的安全关联更新密钥。以正常的间隔将密钥推送给终端，从而避免系统中存在长期有效的密钥。由于推送消息能够通过广播系统发送出去，因此不能认为反向通道适用于所有场景。

那些本身没有内置密钥更新机制的业务（例如，MBMS）可能想要推送新的安全关联给用户。更新密钥的原因可能是用连续的用户业务运行（例如，密钥在某些重要时间段到期了）来均匀地分散网络侧



负载，或者在时间到期时网络里存在某些维护信息（例如，在 Push-NAF 到 BSF 的连接上，该连接导致了带宽的减少）。BSF 和 Push-NAF 知道 Ks 的最大生命周期，如果 Push-NAF 想要获取新鲜密钥（由于密钥到期或者 Push-NAF 策略原因），则 NAF 可以主动推送新的安全关联给 UE。

特性：

- 某些 UE 要求在旧密钥到期之前的某个预定时间建立新的安全关联。
- 已经存 Ua 安全关联，密钥需要在这种场景下更新。
- 业务开展的发生可能远远晚于安全配置。
- 旧 NAF-keys 可能是根据 3GPP TS 33.220 的 GBA 运行中或者是 GBA Push 运行中所生成的 NAF-keys。
- 应支持 GBA\_ME，可能需要支持 GBA\_U，但是这并不适用于 OMA BCAST 及 MBMS，因为它们拥有自己的内置密钥更新方案。然而，GBA Push 可能对于 MBMS 是有用的。

### C.2.2 令牌分发

该使用场景非常类似于密钥分发场景，除了一个重要不同点，即令牌应当通过 IP 连接提供给业务提供商。这样，可以认为存在一个反向通道，而且该反向通道可以用来报告 GBA Push 的成功执行。此外，如果令牌的传递只允许发生在在线的终端上，这要求获取一个及时的成功报告并且排除了使用那些延时的传输信道，那么该方案的有用性将被削弱。

### C.2.3 MBMS GBA\_U 使用场景

GBA Push 对于 MBMS 是有用的。

GBA 的创建是为了保护 MBMS 以及允许依赖于 GBA\_U 的基于 UICC 的解决方案。在将 GBA Push 应用于 MBMS 的情况下，这使得发送用存储在 UICC 上密钥（即，与 Ks\_int\_NAF 相关的密钥）保护的 GBA Push 消息成为可能。GBA Push 解决方案应当支持 GBA\_U。

### C.2.4 OMA 相关的使用场景

OMA BCAST 为 GBA Push 提供了使用场景。其中的一个使用场景是 OMA BCAST 智能卡，这要求使用 GBA Push 解决方案并且终结点在 UICC 上。因此，GBA Push 解决方案应当支持 GBA\_U。特别地，OMA 提供了以下信息：

“基于智能卡的业务的保护使用 MBMS GBA 机制来实现注册和长期密钥的传递。这也将有利于网络侧发起的注册和长期密钥的传递。安全的 GBA Push 机制使得这样的解决方案成为可能。”

其他的 OMA 使用场景可以依赖基于 GBA\_U 的 GBA Push 解决方案。例如，OMA SEC 组认为基于 GBA Push 的密钥管理对于未来设备管理和客户端配置释放是一种很好的增强。该 GBA Push 解决方案可以依赖于密钥 Ks\_ext\_NAF/Ks\_int\_NAF，以实现增强安全。但是对于这些使用场景，还没有 OMA 要求来规定 GBA Push 解决方案应当应用于终结点在 UICC 的场景。

### C.2.5 网络侧发起的业务

很多业务都是由网络侧发起的，这要求终端连接到网络侧的服务器上。OMA 定义的具有这种做法的例子有：Device Management (DM)、Download DRM (DLDRM)、DRM、及 Secure User Plane Location (SUPL)。这些场景都假设触发推送消息可以通过 SMS 发送。



拥有一个有效的安全推送系统将允许更大的灵活性，因为可信的参数和密钥可以在触发消息中发送，而且可以避免配置类似服务器和发送者白名单列表的预配置参数。此外，安全推送也防护了重放攻击和 DoS 拒绝服务攻击。

因为业务要求终端连接到网络侧，这些场景中都存在一个反向通道。当终端连接到服务器，该发起 Push-NAF 将隐式地收到一个确认消息，用来指示 GBA Push 成功执行。然后该成功指示可以由服务器通过 Push-NAF 报告给 BSF。

还存在一种网络侧发起业务的特殊场景，运营商可能想要安全地更新终端上的信息，例如，设备管理信息或者客户端配置信息。而 UE 之前没有联系过运营商，因此不能够被安全地触发来执行自启动过程（即，使用 SMS 或 WAP Push）。设备管理信息应以安全的方式推送给 UE，而且该推送可能发生在某个固定时间点或某个预配置的时间段。应保证管理消息的发起者是被授权的以及只有正确的终端能够使用数据。

特性：

- 配置完成之后就可以直接使用面向 UE 的传递信息（叫做管理消息）。
- 管理消息的源头应是可以被确认的，例如，Push-NAF。
- 只有管理消息的授权接收方能够使用它。
- 安全关联的建立（Upa）和管理消息的传递（Ua）可能不会同时发生。运营商可能想要发送一些被同一个 SA 保护的消息。
- 管理信息的目标是终端，因此支持 Ks\_NAF 就足够了。

#### C.2.6 广播场景下的 BSF 及 HSS 负载均衡

GBA 适用于 MBMS 及 OMA BCAST。广播的主要优点可以在同一时间向大量设备提供相同的内容。典型的广播场景包括足球赛、奥运会、欧洲歌唱大赛、以及其他一些拥有大众兴趣的事件。如果大部分 UEs 在事件之前就执行了 GBA bootstrapping，则 BSF 服务器和 HSS 需要处理大量负载。因此，在 BSF 处于低负载时（例如，事件发生的前一天晚上），网络能够触发注册及长期密钥传递和 UE 能够配置 GBA 信任状（NAF-keys）看起来是合适的。接收终端也需要支持 GBA，如果在终端接收到广播消息时没有可用的安全关联，则发起一次 GBA。

特性：

- 很多 UEs 在某个特定时间为业务请求安全关联。
- 如果 GBA Push 和 GBA 会话同时建立完成，那么 BSF 可以重用为 GBA Push 创建的 Ks 来执行 3GPP TS 33.220 中的 GBA。相反地，3GPP TS 33.220 中建立的 Ks 可以用于 GBA Push。如果没有上行通道，那么不能进行这种重用。
- 很可能没有可用的上行通道，或者不希望（因为网络负载）很多 UEs 在同一时刻同一地点发起上行接入。
- 业务开展的发生可能远远晚于安全配置。
- 终端应支持 3GPP TS 33.220 中的 GBA，以允许在由 GBA Push 所建立的 SA 没有到达的情况下使用服务。
- 只有在没有接收到 SA 的情况下才使用 3GPP TS 33.220 中的 GBA（如果支持）。
- 需要负载均衡的业务可能要求 UE 支持 GBA 或 GBA\_U。



### C.2.7 付款凭单/票券的下载

一种讨论过的用途是，通过向消费者的移动电话推送付款凭单/票券，来为不同类型的公共事件或实物商品的收费分发付款凭单/票券。可以安排在事件发生后及时进行分发以降低付款凭单/票券被删除或丢失、被复制的风险，或者可以提前进行分发以便分配网络侧的负载。

由于付款凭单/票券通常不是个人化的，因此它们应安全地传递给合法接收者。

如果分发机制是可信的，那就没有必要报告接收答复消息给发送者。即使是对于不可信的传递通道，也很可能不需要报告接收答复消息，因为用户会注意到票券没有被传递进而会联系发送者并要求重传票券。

付款凭单/票券的使用不能用作作为一种报告下载成功的可信方式。存在两个原因，第一个原因是付款凭单/票券很可能通过脱机设备来使用，第二个原因是付款凭单/票券可能在传递完成的很长一段时间之后才会被使用。

请注意这种使用场景应存在消息（包含付款凭单/票券）的延时传输。如果接收者关机了，其应能够在开机后马上接收到这条消息。

### C.2.8 新闻/资讯/命令的分发

需要保护分发给员工的新闻/资讯/命令（如股票价格或工作指令），特别需要进行完整性保护和来源认证，而在某些场景下还需要加密保护。这种信息更适合通过一个支持延时传递的系统来分发，这样可以使得业务提供者避免跟踪用户状态以及在用户开机后马上激活相关信息。

### C.2.9 机顶盒使用场景

还存在这种使用场景，机顶盒配备有 UICC 读卡器而没有返回通道。这种场景意味着 GBA Push 消息用 UICC 保护。

### C.2.10 总结

以上所述使用场景及其特性导致了以下要求：

- 存在 Ua 协议终结在 UICC 以及终结在 ME 的使用场景。
- GBA Push 应支持 GBA\_ME 和 GBA\_U。
- 应支持 Ua 消息和 Upa 消息的单独传输。
- Upa bootstrapping 应能够支持单向的消息传递（例如，广播场景的 Upa）。



附 录 D  
(资料性附录)

本标准与 3GPP TS 33.223 和 3GPP TS 33.224 章节对应关系

本标准章条	对应3GPP规范	对应3GPP规范的章节
5.1	3GPP TS 33.223	4.1
5.2	3GPP TS 33.223	4.2
5.3	3GPP TS 33.223	4.3
5.4.1	3GPP TS 33.224	4.1
5.4.2	3GPP TS 33.224	4.2
6	3GPP TS 33.223	5
7	3GPP TS 33.224	5
附录A	3GPP TS 33.223	附录A
附录B	3GPP TS 33.223	附录B
附录C	3GPP TS 33.224	附录A







中华人民共和国  
通信行业标准

数字蜂窝移动通信网通用认证架构通用自启动架构推送功能及推送层协议

YD/T 2922-2015

\*

人民邮电出版社出版发行

北京市丰台区成寿寺路 11 号邮电出版大厦

邮政编码: 100164

北京康利胶印厂印刷

版权所有 不得翻印

\*

开本: 880×1230 1/16

2016 年 1 月第 1 版

印张: 2.25

2016 年 1 月北京第 1 次印刷

字数: 77 千字

15115 • 851

定价: 25 元

本书如有印装质量问题, 请与本社联系 电话: (010)81055492