

SMS API 集成手册

本文档将介绍如何使用 Infobip HTTP API 和 SMPP 接口将 Infobip 通信服务快速集成到各个解决方案中。本文档分为两个章节，第一章介绍 HTTP API，第二章介绍 SMPP 规范。请使用目录更快速、更轻松地浏览 HTTP API 和 SMPP 规范、示例以及教程。

要详细了解规范、各种编程语言的功能齐全的示例和交互式测试环境，请访问我们的开发人员中心，网址为：

dev.infobip.com

目录

技术文档	2
HTTP API 简介	5
基 URL	5
内容类型与 Accept 标头	5
授权	6
发送 SMS	7
发送 SMS 回复	8
获取发送报告	10
获取 SMS 日志	15
提取已接收消息	19
推送已接收消息	22
全功能文本消息	24
Number Lookup	28
回复代码	30
状态组	30
状态	31
错误组	33
GSM 错误代码	34
高级 HTTP API 教程	39
将 SMS 发送至多个目标地址	39
将多条 SMS 发送至多个目标地址	41
设定 SMS 的时间和有效期	43
通知 URL 上的发送报告	46
中期发送报告	50
Flash SMS	53
URL 缩短与跟踪解决方案	54
SMPP API	56

SMPP 参数.....	56
命令状态与 GSM 错误代码.....	58
定时发送.....	58
符合 SMPP 规范的 Number Context.....	59
符合 SMPP 规范的 Flash 通知.....	62
SMPP 命令状态与 GSM 错误代码.....	63
SMPP 命令状态.....	63
SMPP GSM 错误代码.....	63

HTTP API 简介

HTTP 应用程序编程接口 (HTTP API) 是使用标准化 REST 接口集成所有 Infobip 服务的最简便方式。HTTP API 可用于发送 SMS 消息、收集发送报告、执行 Number Lookup (号码验证) 请求和接收手机发送的入站 SMS 消息。

Infobip 的 API 基于 REST 标准，使您能够使用浏览器访问各个 URL。为了与我们的 API 实现交互，可使用任何编程语言的任何 HTTP 客户端。

注意：如果您还没有 Infobip 帐户，请访问我们的[注册](#)页面创建免费帐户。

基 URL

将所有请求提交至基 URL。所有请求都通过 HTTP POST、PUT 或 GET 方法提交。虽然您可以使用 HTTP 协议，但我们强烈建议您将所有请求通过 HTTPS 提交至 Infobip SMS API，以对流量进行加密并确保隐私。

基 URL：<https://api.infobip.com>

内容类型与 Accept 标头

Infobip SMS API 支持 JSON 和 XML 内容类型和 Accept 标准，应在标头中指明这些内容。如果未指明内容类型，您将收到常规错误提醒。根据请求标头中选择的 Accept 类型，回复中将应用与之相同的类型。

内容类型：[application/json](#) 或 [application/xml](#)

Accept 标头：[application/json](#) 或 [application/xml](#)

授权

我们支持使用 Base64 编码变量 [RFC2045-MIME](#) 的用户名和密码进行基本授权。

授权标头构成如下：

1. 用户名和密码组成字符串 `username:password`。
2. 生成的字符串使用 Base64 的 [RFC2045-MIME](#) 变量编码而成。

将授权方法和一个空格（如：“Basic”）放在编码的字符串之前。

示例：

用户名：`Aladdin`

密码：`open sesame`

Base64 编码的字符串：`QWxhZGRpbjpvGVuIHNIc2FtZQ==`

授权标头：`Basic QWxhZGRpbjpvGVuIHNIc2FtZQ==`

发送 SMS

我们将通过几个简单的步骤来解释如何使用 Infobip HTTP API 发送 SMS。

首先，您需要一个有效的 Infobip 帐户。在[注册帐户](#)时，您将设置用户名和密码。接下来，您的用户名和密码将使用 **base64** 进行编码，正如[授权部分](#)所述：

消息将仅发送给一个（或多个）有效的电话号码，即以**国际格式**书写的号码，如 41793026727。我们强烈建议使用 E.164 号码格式。E.164 号码经过了国际标准化处理，最长为 15 位数。电话号码前通常会有 +（加号）前缀，后跟国家/地区代码、网络代码和用户号码。根据手机或网络的不同，非 E.164 格式的电话号码也可能正常使用。

现在，您已准备好使用以下方法发送首条 SMS 消息：

POST <https://api.infobip.com/sms/1/text/single>

请求正文包含要发送的消息，包括 **from**、**to** 和 **text** 参数。

完整的 JSON 请求如下所示：

JSON

```
POST /sms/1/text/single HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json
Accept:application/json

{
  "from":"InfoSMS",
  "to":"41793026727",
  "text":"My first Infobip SMS"
}
```

发送 SMS 回复

将“发送 SMS” HTTP 请求提交至 Infobip SMS API 后，您将收到一条回复，其中包含一些有用的信息。正常情况下，所提供的 **200 OK** 回复应在回复正文中包含消息详情。

下面是发送单个 SMS 的请求示例：

JSON

```
POST /sms/1/text/single HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json
Accept:application/json

{
  "from":"InfoSMS",
  "to":"41793026727",
  "text":"My first Infobip SMS"
}
```

相应回复如下所示：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "messages":[
    {
      "to":"41793026727",
      "status":{
        "id":0,
        "groupId":0,
        "groupName":"ACCEPTED",
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":1,
      "messageId":"2250be2d4219-3af1-78856-aabe-1362af1edfd2"
    }
  ]
}
```


- **messages** 是最后一条请求中发送的所有 SMS 消息的阵列。在本例中，仅包含一条消息。
- **to** 是要将 SMS 消息发送至的手机号码
- 每条成功提交至 Infobip 平台的消息都会使用唯一的 **messageId** 进行标识。此外，消息 ID 可用于检查发送状态或已发送消息日志
- **smsCount** 是消息拆分后的条数
- **status** 是进一步描述已发送消息状态的对象。有关可用状态的完整列表，请参阅

获取发送报告

在发出一些消息后，您能够通过执行以下请求来检查这些消息是否成功发送：

`GET https://api.infobip.com/sms/1/reports`

可用的查询参数为：

- **bulkId**：唯一标识已发送 SMS 请求的 ID。此筛选器可让您仅使用一个请求收取所有消息的发送报告。发送 SMS 请求后，您会在回复中收到一个 **bulkId**，或者您可以设置一个自定义 **bulkId**。
- **messageId**：唯一标识已发送消息的 ID。发送消息后，您会在回复中收到一个 **messageId**，或者您可以设置一个自定义 **messageId**。
- **limit**：您想要获取的发送报告的最大数量。默认值为 **50**。

在回复中，您将获得未读发送报告的集合。

重要事项：发送报告只能检索一次。一旦检索了发送报告，您将不能使用此端点再次获得相同的报告。

下面是不使用任何查询参数获取报告的 JSON 请求示例：

JSON

```
GET /sms/1/reports HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpwGVuIHNlc2FtZQ==
Accept:application/json
```

您可以在下面看到发送报告请求的回复：

JSON

```

HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[
    {
      "bulkId":"80664c0c-e1ca-414d-806a-5caf146463df",
      "messageId":"bcfb828b-7df9-4e7b-8715-f34f5c61271a",
      "to":"41793026731",
      "sentAt":"2015-02-12T09:51:43.123+0100",
      "doneAt":"2015-02-12T09:51:43.127+0100",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.01,
        "currency":"EUR"
      },
      "callbackData":"User defined data.",
      "status":{
        "groupId":3,
        "groupName":"DELIVERED",
        "id":5,
        "name":"DELIVERED_TO_HANDSET",
        "description":"Message delivered to handset"
      },
      "error":{
        "groupId":0,
        "groupName":"OK",
        "id":0,
        "name":"NO_ERROR",
        "description":"No Error",
        "permanent":false
      }
    },
    {
      "bulkId":"08fe4407-c48f-4d4b-a2f4-9ff583c985b8",
      "messageId":"12db39c3-7822-4e72-a3ec-c87442c0ffc5",
      "to":"41793026727",
      "sentAt":"2015-02-12T09:50:22.221+0100",
      "doneAt":"2015-02-12T09:50:22.232+0100",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.01,
        "currency":"EUR"
      },
      "callbackData":"reset_password",
      "status":{
        "groupId":3,
        "groupName":"DELIVERED",
        "id":5,
        "name":"DELIVERED_TO_HANDSET",
        "description":"Message delivered to handset"
      },
      "error":{
        "groupId":0,
        "groupName":"OK",
        "id":0,
        "name":"NO_ERROR",
        "description":"No Error",
    
```

```
        "permanent":false
    }
  }
]
}
```

在回复中，您将收到 **results** 阵列，其中包含：

- **to** 代表接收人的电话号码。您可以借此将发送报告与电话号码关联起来。
- **bulkId** 和 **messageId**，唯一标识已发送请求和消息的 ID。
- **sentAt** 和 **doneAt**
- **smsCount** 代表消息数量
- **price** 对象（含有 **pricePerMessage** 和 **currency** 参数）
- **callbackData** 对象（附带用户定义数据）
- **status** 和 **error** 对象

注意：如果您尝试再次执行相同的请求，您将获得空白的设置，因为所有发送报告均已读取：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[]
}
```

如果您发送了大量消息，但只想查看一条消息的发送报告，只需在请求中设置一个查询参数即可。

在请求 URL 上附加 `?messageId=ff4804ef-6ab6-4abd-984d-ab3b1387e852`，便能仅获得该消息的发送报告。

除了 `messageId` 以外，您还可以使用 `bulkId` 或直接设置想要检索的报告数量的限制 `limit`。下面是使用查询参数获取报告的 JSON 请求示例：

JSON

```
GET /sms/1/reports?messageId=ff4804ef-6ab6-4abd-984d-ab3b1387e852 HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGvUHNlc2FtZQ==
Accept:application/json
```

以下 JSON 将作为回复提供：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[
    {
      "bulkId":"8c20f086-d82b-48cc-b2b3-3ca5f7aca9fb",
      "messageId":"ff4804ef-6ab6-4abd-984d-ab3b1387e852",
      "to":"41793026731",
      "sentAt":"2015-02-12T09:58:20.323+0100",
      "doneAt":"2015-02-12T09:58:20.337+0100",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.01,
        "currency":"EUR"
      },
    },
    "status":{
```

```
    "id":5,
    "groupId":3,
    "groupName":"DELIVERED",

    "name":"DELIVERED_TO_HANDSET",
    "description":"Message delivered to handset"
  },
  "error":{
    "groupId":0,
    "groupName":"OK",
    "id":0,
    "name":"NO_ERROR",
    "description":"No Error",
    "permanent":false
  }
}
]
```

正如您所见，该消息已成功发送，无任何错误。与一次性的发送报告不同的是，**日志**可用于查看已发送的所有消息的历史记录。在本文档的下一部分中，您可以查看如何使用我们的 API 获取日志。

获取 SMS 日志

可使用一个请求针对所有消息请求获取已发送 SMS 消息历史记录的日志：

GET <https://api.infobip.com/sms/1/logs>.

与发送报告不同的是，您可以根据需要发送任意次数的日志请求。

让我们来看看不使用任何查询参数请求所有日志会发生的情况：

JSON

```
GET /sms/1/logs HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Accept:application/json
```

在回复中，您将获得以下结果：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[
    {
      "bulkId":"bafdeb3d-719b-4cce-8762-54d47b40f3c5",
      "messageId":"07e03aae-fabc-44ad-b1ce-222e14094d70",
      "to":"41793026727",
      "from":"InfoSMS",
      "text":"Test SMS.",
      "sentAt":"2015-02-23T17:41:11.833+0100",
      "doneAt":"2015-02-23T17:41:11.843+0100",
      "smsCount":1,
      "mccmnc":"22801",
      "price":{
        "pricePerMessage":0.01,
        "currency":"EUR"
      },
    },
    "status":{
      "groupId":3,
      "groupName":"DELIVERED",
      "id":5,
      "name":"DELIVERED_TO_HANDSET",
      "description":"Message delivered to handset"
    },
    "error":{
      "groupId":0,
      "groupName":"OK",
      "id":0,
      "name":"NO_ERROR",
    }
  ]
}
```

```

        "description": "No Error",
        "permanent": false
    }
},
{
    "bulkId": "06479ba3-5977-47f6-9346-fee0369bc76b",
    "messageId": "1f21d8d7-f306-4f53-9f6e-eddfce9849ea",
    "to": "41793026727",
    "from": "InfoSMS",
    "text": "Test SMS.",
    "sentAt": "2015-02-23T17:40:31.773+0100",
    "doneAt": "2015-02-23T17:40:31.787+0100",
    "smsCount": 1,
    "mccmnc": "22801",
    "price": {
        "pricePerMessage": 0.01,
        "currency": "EUR"
    },
    "status": {
        "groupId": 3,
        "groupName": "DELIVERED",
        "id": 5,
        "name": "DELIVERED_TO_HANDSET",
        "description": "Message delivered to handset"
    },
    "error": {
        "groupId": 0,
        "groupName": "OK",
        "id": 0,
        "name": "NO_ERROR",
        "description": "No Error",
        "permanent": false
    }
}
]
}

```

日志具有与发送报告类似的信息，并另外添加了一些字段。如果您需要关于这些回复字段的详细信息，请查看[回复代码](#)部分。

重要事项： 可获取最近 48 小时的 SMS 日志！

由于本日志示例适用于您于最近 **48 小时** 通过 Infobip 平台发送的所有消息，您可能需要使用一些筛选器进行搜索。可用的筛选器有：

参数	类型	说明
from	String	发送人地址。
to	String	目标地址。
bulkId	String[]	请求日志的批量 ID。
messageId	String[]	请求日志的消息 ID。
generalStatus	String	已发送 SMS 的状态。
sentSince	Date	发送 SMS 的日期和时间下限。
sentUntil	Date	发送 SMS 的日期和时间上限。
limit	int	返回日志中的最大消息数。默认值为 50 。
mcc	String	移动国家 / 地区代码。
mnc	String	移动网络代码。

现在，我们来尝试使用“from”、“to”和“limit”作为筛选器获取日志：

JSON

```
GET /sms/1/logs?from=InfoSMS&to=41793026727&limit=1 HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHhlc2FtZQ==
Accept:application/json
```

回复将为：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[
    {
      "bulkId":"82d1d36e-e4fb-4194-8b93-caeb053bd327",
      "messageId":"fc0cbfb8-7a72-40da-a76d-e2c2d9400835",
      "to":"41793026727",
      "from":"InfoSMS",
      "text":"Test SMS.",
      "sentAt":"2015-02-23T17:42:05.390+0100",
      "doneAt":"2015-02-23T17:42:05.390+0100",
      "smsCount":1,
      "mccmnc":"22801",
      "price":{
        "pricePerMessage":0,
        "currency":"EUR"
      },
      "status":{
        "groupId":5,
        "groupName":"REJECTED",
        "id":6,
        "name":"REJECTED_NETWORK",
        "description":"Network is forbidden",
        "action":"Contact account manager"
      },
      "error":{
        "groupId":0,
        "groupName":"OK",
        "id":0,
        "name":"NO_ERROR",
        "description":"No Error",
        "permanent":false
      }
    }
  ]
}
```

提取已接收消息

Infobip 提供不同的方法来收集客户的 GSM 手机发送的 SMS 消息。例如，我们可以将您的 GSM SIM 卡托管在我们的 GSM 调制解调器群中。当客户将 SMS 消息发送至 SIM 时，消息会发送到我们的系统中。要提取已接收消息，首先您需要一个电话号码，并对该号码 **设置提取操作**。一旦检索了某条已接收消息，您将不能使用此端点再次获得相同的消息。

有关更多详细信息，请联系您的帐户经理或我们的 **支持团队**。

要提取 SMS 消息，可使用以下方法：

<https://api.infobip.com/sms/1/inbox/reports>

参数	类型	说明
limit	Integer	返回的最大已接收消息数。

提取已接收消息示例

JSON

```
GET /sms/1/inbox/reports HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Accept:application/json
```

结果格式

JSON

```
{
  "results":[
    {
      "messageId":"ff4804ef-6ab6-4abd-984d-ab3b1387e823",
      "from":"38598111",
      "to":"41793026727",
      "text":"KEY Test message",
      "cleanText":"Test message",
      "keyword":"KEY",
      "receivedAt":"2015-02-15T11:43:20.254+0100",
      "smsCount":1
    }
  ]
}
```

回复正文参数

如果成功，回复标头 HTTP 状态代码将为 **200 OK**，且消息将返回到回复正文中。

如果您未经授权就尝试获取已接收消息，则您收到的回复将包含 HTTP 状态代码 **401 Unauthorized**。

SMS 回复

参数	类型	说明
results	Messages[]	报告集合，每条已接收消息会产生一个报告。

消息

参数	类型	说明
messageId	String	唯一标识已接收消息的 ID。
from	String	可为字母数字或数字的发送人 ID。
to	String	消息目标地址。
text	String	已接收消息的完整文本。
cleanText	String	无关键字的已接收消息文本（如果已发送关键字）。
keyword	String	从消息文本中提取的关键字。
receivedAt	Date	显示 Infobip 平台接收消息的时间。拥有以下格式：yyyy-MM-dd'T'HH:mm:ss.SSSXXX。
smsCount	int	已发送消息的分段数。

推送已接收消息

我们的系统收到消息后，可默认使用 HTTP GET 请求将其转发至您的服务器；也可以使用 POST 转发，但此操作需要根据各个请求完成。您必须提供一个我们应使用的 URL。这意味着您必须在 Web 服务器上准备这样一个 URL。

参数

参数	类型	说明
Sender	String	SMS 消息发送人 (GSM 电话号码)
Receiver	String	接收人号码 (如有)
Text	String	已接收消息文本
Bin	String	已接收消息的二进制内容
Datetime	Date	消息接收的日期和时间
MessageId	Date	特定 MO 消息的标识符
Datacoding	Integer	消息数据编码
Esmclass	Integer	消息的 ESM 级参数
output	String	期望输出值、支持值为 (可选) : xml: xml 格式的值 json: json 格式的值

您提供包含 bin 和 text 参数的 URL 时，请注意以下内容：如果 datacoding 参数为 “0”，我们将仅向您转发消息文本，bin 参数将设置为 “ ”（空白字符串）。如果 datacoding 不为 “0”（例如 “8” = Unicode 消息），我们将仅向您发送二进制内容，参数文本将设置为 “ ”（空白字符串）。

但是，如果您的 URL 中不支持这两种参数（bin 和 text）（当然，为了接收消息内容，您应该至少使用一个），无论 datacoding 参数为何值，我们都将提供所有消息内容。我们使用“仅发送二进制或仅发送文本”的逻辑，最大限度缩短 HTTP GET 请求的长度。

例如，如果您提供以下 URL：

```
https://some.server.com/incoming_sms.php?who=%sender%&what=%text%&output=xml
```

然后，我们的系统将执行以下 HTTP 请求（从名称显示为 “ABC” 的 +38598123123 接收消息后）：

```
https://some.server.com/incoming_sms.php?who=38598123123&what=ABC
```

请注意，“发送人”字段开头无“+”。如果您想使用“binary”参数而非“text”，应该提供以下 URL：

```
https://some.server.com/incoming_sms.php?who=%sender%&what=%bin%
```

如此可执行以下请求：

```
https://some.server.com/incoming_sms.php?who=38598123123&what=414243
```

全功能文本消息

对于高级 SMS 消息传送，您可以使用适用于全功能文本消息（包含所有可用功能和参数）的 API：

POST <https://api.infobip.com/sms/1/text/advanced>

参数

参数	类型	说明
bulkId	String	唯一标识请求的 ID。只有向一个以上目标地址发送消息时，才会收到批量 ID。
from	String	代表发送人 ID，可为字母数字或数字。字母数字发送人 ID 长度应在 3 到 11 个字符之间（示例： CompanyName ）。数字发送人 ID 长度应在 3 到 14 个字符之间。
to	String	消息目标地址。地址必须为国际格式（示例： 41793026727 ）。
messageId	String	唯一标识已发送消息的 ID。
text	String	将发送消息的文本。
flash	Boolean	可为 true 或 false。如果该值设置为 true，将发送 Flash SMS。反之，将发送普通 SMS。默认值为 false。
transliteration	String	消息文本从一种脚本转换为另一种脚本的过程。可能值：“ TURKISH ”、“ GREEK ”、“ CYRILLIC ”、“ CENTRAL_EUROPEAN ”和“ NON_UNICODE ”
languageCode	String	消息文本语言字符集代码。可能值：TR 代表土耳其语，ES 代表西班牙语，PT 代表葡萄牙语。
singleShift	Boolean	替代 GSM 7 位默认字母扩展表的单一移位表。
lockingShift	Boolean	替代标准 GSM 7 位默认字母表的锁定移位表。
intermediateReport	Boolean	将在回调服务器上发送的实时中期发送报告。可为 true 或 false。

notifyUrl	String	将发送发送报告的回调服务器上的 URL。
notifyContentType	String	首选发送报告内容类型。可为 <code>application/json</code> 或 <code>application/xml</code> 。
callbackData	String	将在 <code>notifyUrl</code> 上发送的其他客户端数据。
validityPeriod	Integer	消息有效期（以分钟为单位）。有效期过后，将无法发送相应消息。不支持超过 48 小时的有效期（在这种情况下，系统将自动设置为 48 小时）。
sendAt	DateTime	发送消息的日期和时间。用于设定 SMS 的发送时间（并非立即发送，而在规定时间发送的 SMS）。
track	String	表明是否需要跟踪消息的转换率。 可能值： <code>SMS</code> 和 <code>URL</code>
processKey	String	唯一标识转换跟踪进程的密钥。
type	String	用户定义的转换跟踪进程类型、流程类型或消息类型等。示例： <code>ONE_TIME_PIN</code> 或 <code>SOCIAL_INVITES</code>

示例

JSON

```

POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "bulkId":"BULK-ID-123-xyz",
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727",
          "messageId":"MESSAGE-ID-123-xyz"
        },
        {
          "to":"41793026731"
        }
      ],
      "text":"Artık Ulusal Dil Tanımlayıcısı ile Türkçe karakterli smslerinizi rahatlıkla iletebilirsiniz.",
      "flash":false,
      "language":{
        "languageCode":"TR",
        "singleShift":true,
        "lockingShift":false
      },
      "transliteration":"TURKISH",
      "intermediateReport":true,
      "notifyUrl":"http://www.example.com/sms/advanced",
      "notifyContentType":"application/json",
      "callbackData":"DLR callback data",
      "validityPeriod":720
    },
    {
      "from":"41793026700",
      "destinations":[
        {
          "to":"41793026785"
        }
      ],
      "text":"A long time ago, in a galaxy far, far away...It is a period of civil war. Rebel spaceships, striking from a hidden base, have won their first victory against the evil Galactic Empire.",
      "sendAt":"2015-07-07T17:00:00.000+01:00"
    }
  ],
  "tracking":{
    "track":"SMS",
    "type":"MY_CAMPAIGN"
  }
}

```

结果格式

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "bulkId":"BULK-ID-123-xyz",
  "messages":[
    {
      "to":"41793026727",
      "status":{
        "groupId":0,
        "groupName":"ACCEPTED",
        "id":0,
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":1,
      "messageId":"MESSAGE-ID-123-xyz"
    },
    {
      "to":"41793026731",
      "status":{
        "groupId":0,
        "groupName":"ACCEPTED",
        "id":0,
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":1,
      "messageId":"9304a5a3ab19-1ca1-be74-76ad87651ed25f35"
    },
    {
      "to":"41793026785",
      "status":{
        "groupId":0,
        "groupName":"ACCEPTED",
        "id":0,
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":2,
      "messageId":"5f35f87a2f19-a141-43a4-91cd81b85f8c689"
    }
  ]
}
```

Number Lookup

Number Lookup 可帮助您保持更新手机号码数据库。移动用户经常更换号码，进入漫游模式，以及在保持原电话号码不变的情况下更换供应商。了解哪些移动号码正在使用及可以使用，或者您的客户目前正在使用哪种网络，可以大幅提高许多企业类型的准确性和成本效益。

借助 Number Lookup，您可以确定：

- 当前处于活跃状态的号码
- 手机号码是否处于漫游状态
- 手机号码是否曾移植
- 消息和语音的最佳路由
- 号码类型（如陆线、机对机、手机等）

提供各种 **Number Lookup 程序包**，因此您可以选择最适合您企业的一款：

- **Database Cleaning**：我们的 Database Cleaning 程序包专为具有大规模号码数据库的公司设计，支持识别未使用和非活跃的号码。
- **Portability**：此程序包专为解决各种公司类型的号码可移植性问题而开发，支持实时号码可移植性查找，从而优化消息和语音路由。它包括 Database Cleaning 程序包的所有功能。
- **Roaming**：此程序包主要用于服务金融客户细分市场，提供漫游信息，从而优化路由、防止 ATM 诈骗等。

下面的示例展示了如何使用我们的 API 获取 Number Lookup 信息：

JSON

```
POST /number/1/query HTTP/1.1
Host:api.infobip.com
Authorization:Basic dXNlcm5hbWU6cGFzc3dvcnQ=
Content-Type:application/json
Accept:application/json

{
  "to":["41793026727"]
}
```

to 参数是您想要查看的所有号码的列表。

结果如下：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "results":[
    {
      "to":"41793026727",
      "mccMnc":"22801",
      "imsi":"228012120181810",
      "originalNetwork":{"
        "networkPrefix":"79",
        "countryPrefix":"41"
      }},
      "ported":false,
      "roaming":false,
      "status":{"
        "groupId":3,
        "groupName":"DELIVERED",
        "id":5,
        "name":"DELIVERED_TO_HANDSET",
        "description":"Message delivered to handset"
      }},
      "error":{"
        "groupId":0,
        "groupName":"OK",
        "id":0,
        "name":"NO_ERROR",
        "description":"No Error",
        "permanent":false
      }
    }
  ]
}
```

说明：根据程序包的不同，部分信息可能无法访问。有关程序包变更的信息，请联系您的帐户经理。

回复代码

请查看回复代码列表，了解 Infobip 可能提供的状态和 GSM 错误。

状态对象示例

JSON

```
{
  "groupId":3,
  "groupName":"DELIVERED",
  "id":5,
  "name":"DELIVERED_TO_HANDSET",
  "description":"Message delivered to handset"
}
```

状态组

GroupId	GroupName	说明
0	ACCEPTED	消息已接收。
1	PENDING	消息处于待接收状态。
2	UNDELIVERABLE	消息无法发送。
3	DELIVERED	消息已发送。
4	EXPIRED	消息已过期。
5	REJECTED	消息被拒绝。

状态

Id	GroupId	名称	说明	操作
1	1	PENDING_TIME_VIOLATION	违反时间期限规定	无
2	3	DELIVERED_TO_OPERATOR	消息已发送至运营商	无
3	1	PENDING_WAITING_DELIVERY	消息已发送，等待发送报告	无
4	2	UNDELIVERABLE_REJECTED_OPERATOR	消息被运营商拒绝	无
5	3	DELIVERED_TO_HANDSET	消息已发送至手机	无
6	5	REJECTED_NETWORK	网络被禁用	联系帐户经理
7	1	PENDING_ENROUTE	消息已发送至下一个实例	无
8	5	REJECTED_PREFIX_MISSING	号码前缀丢失	无
9	2	UNDELIVERABLE_NOT_DELIVERED	所发消息未成功发送	无
10	5	REJECTED_DND	DND 列表上的目标地址	无
11	5	REJECTED_SOURCE	源地址无效	无
12	5	REJECTED_NOT_ENOUGH_CREDITS	信用不足	无
13	5	REJECTED_SENDER	按发送人	将发送人从黑名单移除
14	5	REJECTED_DESTINATION	按目标地址	将目标地址从黑名单移除
15	4	EXPIRED_EXPIRED	消息已过期	无

Id	GroupId	名称	说明	操作
16	5	REJECTED_NOT_REACHABLE	网络无法获取	无
17	5	REJECTED_PREPAID_PACKAGE_EXPIRED	预付套餐已过期	充值帐户以延长有效期
18	5	REJECTED_DESTINATION_NOT_REGISTERED	目标地址未注册	无
19	5	REJECTED_ROUTE_NOT_AVAILABLE	路由不可用	联系帐户经理
20	5	REJECTED_FLOODING_FILTER	洪泛被拒绝	阻止垃圾信息
21	5	REJECTED_SYSTEM_ERROR	系统错误	再试一次
22	4	EXPIRED_UNKNOWN	未知原因	无
23	5	REJECTED_DUPLICATE_MESSAGE_ID	重复消息 ID 被拒绝	无
24	5	REJECTED_INVALID_UDH	无效 UDH 被拒绝	无
25	5	REJECTED_MESSAGE_TOO_LONG	消息过长被拒绝	无
26	1	PENDING_ACCEPTED	等待接收	无
27	1	PENDING_APPROVAL	等待批准	无
28	5	REJECTED_NOT_SENT	未发送消息被拒绝	无
29	4	EXPIRED_DLR_UNKNOWN	DLR 未知过期	无
30	3	DELIVERED	MO 转发操作已完成	无
31	2	UNDELIVERABLE_NOT_SENT	消息未发送	无
51	5	MISSING_TO	目标地址丢失	检查参数。
52	5	REJECTED_DESTINATION	目标地址无效。	检查参数。

错误对象示例

JSON

```
{  
  "groupId":0,  
  "groupName":"OK",  
  "id":0,  
  "name":"NO_ERROR",  
  "description":"No Error",  
  "permanent":false  
}
```

错误组

GroupId	GroupName	说明
0	OK	无错误。
1	HANDSET_ERRORS	出现手机错误。
2	USER_ERRORS	出现用户错误。
3	OPERATOR_ERRORS	出现运营商错误。

GSM 错误代码

Id	GroupId	名称	说明	permanent 状态
0	0	NO_ERROR	无错误	false
1	1	EC_UNKNOWN_SUBSCRIBER	未知用户	true
5	1	EC_UNIDENTIFIED_SUBSCRIBER	未定义用户	false
6	1	EC_ABSENT_SUBSCRIBER_SM	缺席用户	false
9	1	EC_ILLEGAL_SUBSCRIBER	非法用户	true
10	3	EC_BEARER_SERVICE_NOT_PROVISIONED	不支持承载业务	true
11	1	EC_TELESERVICE_NOT_PROVISIONED	不支持电信业务	true
12	1	EC_ILLEGAL_EQUIPMENT	非法设备	true
13	1	EC_CALL_BARRED	呼叫被禁止	false
20	3	EC_SS_INCOMPATIBILITY	SS 不兼容	false
21	1	EC_FACILITY_NOT_SUPPORTED	不支持设施	false
27	1	EC_ABSENT_SUBSCRIBER	缺席用户	false
31	1	EC_SUBSCRIBER_BUSY_FOR_MT_SMS	Mt SMS 用户忙	false
32	1	EC_SM_DELIVERY_FAILURE	SM 发送失败	false
33	1	EC_MESSAGE_WAITING_LIST_FULL	消息等待队列满	false
34	1	EC_SYSTEM_FAILURE	系统故障	false
35	1	EC_DATA_MISSING	数据丢失	false
36	1	EC_UNEXPECTED_DATA_VALUE	意外数据值	false
51	3	EC_RESOURCE_LIMITATION	资源限制	true
71	3	EC_UNKNOWN_ALPHABET	未知字母表	false

Id	GroupId	名称	说明	permanent 状态
72	1	EC_USSD_BUSY	Ussd 忙	true
255	1	EC_UNKNOWN_ERROR	未知错误	false
256	1	EC_SM_DF_MEMORYCAPACITYEXCEED D	超过 SM DF 存储容量	false
257	1	EC_SM_DF_EQUIPMENTPROTOCOLERRO R	SM DF 设备协议错误	false
258	1	EC_SM_DF_EQUIPMENTNOTSM_EQUIPPE D	SM DF 设备不具备 SM	false
259	1	EC_SM_DF_UNKNOWNSERVICECENTRE	SM DF 未知服务中心	false
260	1	EC_SM_DF_SC_CONGESTION	SM DF SC 拥堵	false
261	1	EC_SM_DF_INVALIDSME_ADDRESS	SM DF InvalidSME 地址	false
262	1	EC_SM_DF_SUBSCRIBERNOTSC_SUBSCR IBER	SM DF Subscribernotsc 用户	false
500	1	EC_PROVIDER_GENERAL_ERROR	供应商一般性错误	false
501	3	EC_INVALID_RESPONSE_RECEIVED	接收的回复无效	false
502	1	EC_NO_RESPONSE	无回复	false
503	1	EC_SERVICE_COMPLETION_FAILURE	服务完成失败	false
504	1	EC_UNEXPECTED_RESPONSE_FROM_PE ER	意外对等回复	false
507	1	EC_MISTYPED_PARAMETER	参数输入错误	false
508	1	EC_NOT_SUPPORTED_SERVICE	服务受支持	false
509	1	EC_DUPLICATED_INVOKE_ID	调用 Id 重复	false
511	1	EC_INITIATING_RELEASE	发起释放	true
1024	1	EC_OR_APPCONTEXTNOTSUPPORTED	应用程序上下文不受支 持	false

Id	GroupId	名称	说明	permanent 状态
1025	1	EC_OR_INVALIDDESTINATIONREFERENC E	目标地址引用无效	false
1026	1	EC_OR_INVALIDORIGINATINGREFERENCE	原始引用无效	false
1027	1	EC_OR_ENCAPSULATEDAC_NOTSUPPORT ED	封装 AC 不受支持	false
1028	1	EC_OR_TRANSPORTPROTECTIONNOTAD EQUATE	传输保护不足	false
1029	1	EC_OR_NOREASONGIVEN	未提供原因	false
1030	1	EC_OR_POTENTIALVERSIONINCOMPATIBI LITY	潜在版本不兼容性	false
1031	1	EC_OR_REMOTENODENOTREACHABLE	无法获取远程节点	false
1152	1	EC_NNR_NOTTRANSLATIONFORANADDRE SSOFSUCHNATURE	无此类性质的地址的翻 译	false
1153	1	EC_NNR_NOTTRANSLATIONFORTHISSPECI FICADDRESS	无此特定地址的翻译	false
1154	1	EC_NNR_SUBSYSTEMCONGESTION	子系统拥堵	false
1155	1	EC_NNR_SUBSYSTEMFAILURE	子系统故障	false
1156	1	EC_NNR_UNEQUIPPEDUSER	未装载用户	false
1157	1	EC_NNR_MTPFAILURE	MTP 故障	false
1158	1	EC_NNR_NETWORKCONGESTION	网络拥堵	false
1159	1	EC_NNR_UNQUALIFIED	不合格	false
1160	1	EC_NNR_ERRORINMESSAGETRANSPORT XUDT	消息传输 XUDT 出现错 误	false
1161	1	EC_NNR_ERRORINLOCALPROCESSINGXU DT	本地处理 XUDT 出现错 误	false

Id	GroupId	名称	说明	permanent 状态
1162	1	EC_NNR_DESTINATIONCANNOTPERFORM REASSEMBLYXUDT	目标地址无法执行重组 XUDT	false
1163	1	EC_NNR_SCCPFAILURE	SCCP 故障	false
1164	1	EC_NNR_HOPCOUNTERVIOLATION	违反点计数器规定	false
1165	1	EC_NNR_SEGMENTATIONNOTSUPPORTE D	细分不受支持	false
1166	1	EC_NNR_SEGMENTATIONFAILURE	细分故障	false
1281	1	EC_UA_USERSPECIFICREASON	特定于用户的原因	false
1282	1	EC_UA_USERRESOURCELIMITATION	用户资源限制	false
1283	1	EC_UA_RESOURCEUNAVAILABLE	资源不可用	false
1284	1	EC_UA_APPLICATIONPROCEDURECANCE LLATION	应用程序规程取消	false
1536	1	EC_PA_PROVIDERMALFUNCTION	供应商故障	false
1537	1	EC_PA_SUPPORTINGDIALOGORTRANSAC TIONRELEASED	支持对话或交易已释放	false
1538	1	EC_PA_RESSOURCELIMITATION	资源限制	false
1539	1	EC_PA_MAINTENANCEACTIVITY	维护活动	false
1540	1	EC_PA_VERSIONINCOMPATIBILITY	版本不兼容性	false
1541	1	EC_PA_ABNORMALMAPDIALOG	映射对话异常	false
1792	1	EC_NC_ABNORMALEVENTDETECTEDBY PEER	对等检测的事件异常	false
1793	1	EC_NC_RESPONSEREJECTEDBYPEER	回复被对等拒绝	false
1794	1	EC_NC_ABNORMALEVENTRECEIVEDFR OPEER	异常事件被对等点接收	false
1795	1	EC_NC_MESSAGECANNOTBEDELIVERED TOPEER	消息无法发送给对等点	false

Id	GroupId	名称	说明	permanent 状态
1796	1	EC_NC_PROVIDEROUTOFINVOKE	供应商无法调用	false
2048	3	EC_TIME_OUT	超时	false
2049	2	EC_IMSI_BLACKLISTED	IMSI 已列入黑名单	true
2050	3	EC_DEST_ADDRESS_BLACKLISTED	DND 已列入黑名单	true
2051	3	EC_INVALIDMSCADDRESS	文本已列入黑名单	false
4096	2	EC_INVALID_PDU_FORMAT	PDU 格式无效	true
4097	3	EC_NOTSUBMITTEDTOGSMSC	未提交至 GSMSC	false
4100	2	EC_MESSAGE_CANCELED	消息已取消	true
4101	2	EC_VALIDITYEXPIRED	有效期已过期	true
4102	3	EC_NOTSUBMITTEDTOSMPPCHANNEL	未提交至 SMPP 渠道	true
5000	0	VOICE_ANSWERED	呼叫由人员接听	true
5001	0	VOICE_ANSWERED_MACHINE	呼叫由机器接听	true
5002	2	EC_VOICE_USER_BUSY	尝试呼叫时用户忙	true
5003	2	EC_VOICE_NO_ANSWER	用户已收到通知，但未 应答呼叫	true
5004	2	EC_VOICE_ERROR_DOWNLOADING_FILE	针对呼叫提供的文件无 法下载	true
5005	2	EC_VOICE_ERROR_UNSUPPORTED_AUDIO_FORMAT	针对呼叫提供的文件格 式不受支持	true

高级 HTTP API 教程

将 SMS 发送至多个目标地址

要向多个手机号码发送相同的消息，您需要执行：

POST <https://api.infobip.com/sms/1/text/single>.

在示例中，我们的请求将仅包含两个电话号码以便于理解：

JSON

```
POST /sms/1/text/single HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "from":"WineShop",
  "to":[
    "41793026727",
    "41793026834"
  ],
  "text":"Wine shop grand opening at Monday 8pm.Don't forget glasses."
}
```

请求的**标头**部分应该包含 authorization 和 content type :

- Authorization: **Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==**
- Content-Type: **application/json**

请求正文中有三项参数：

- **from** 参数代表 SMS 消息的发送人 - 可为字母数字或数字。**字母数字**发送人 ID 长度应在 3 到 13 个字符之间（示例：**CompanyName**）。**数字**发送人 ID 长度应在 3 到 14 个字符之间。
- **to** 参数为消息目标地址阵列。目标地址必须为国际格式（示例：**41793026727**）。
- **text**：将发送消息的文本。

这将向两个地址发送一条包含相同内容和发送人的 SMS。

您将收到的回复将如下所示：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "bulkId":"f5c4322c-10e7-a41e-5528-34fa0b032134",
  "messages":[
    {
      "to":"41793026727",
      "status":{
        "id":0,
        "groupId":0,
        "groupName":"ACCEPTED",
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":1,
      "messageId":"4a54f0242f19-b832-1c39-a7e7a2095f351ed2"
    },
    {
      "to":"41793026834",
      "status":{
        "id":0,
        "groupId":0,
        "groupName":"ACCEPTED",
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      },
      "smsCount":1,
      "messageId":"9404a69cef19-7a31-ba39-92ace76a5f351ed2"
    }
  ]
}
```

在回复中，您将收到一个 **bulkId** 和一个 **messages** 阵列：

- **bulkId** 用于获取发送至多个目标地址的 SMS 消息的发送报告。
- **messages** 阵列由发送 SMS 回复详细信息组成：
 - **to** 参数表示消息接收人
 - **status** 对象表示消息状态
 - **smsCount** 代表发送至一个目标地址的 SMS 消息数量
 - **messageId** 可唯一标识已发送消息

将多条 SMS 发送至多个目标地址

通过仅调用一次某个 API 方法，将几条特定的消息发送至多个目标地址。您的请求应如下所示：

JSON

```
POST /sms/1/text/multi HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNIc2FtZQ==
Content-Type:application/json

{
  "messages":[
    {
      "from":"WineShop",
      "to":"41793026727",
      "text":"Hey Mike, delicious Istrian Malvazija is finally here.Feel free to visit u
s and try it for free!"
    },
    {
      "from":"WineShop",
      "to":"41793026834",
      "text":"Hi Jenny, we have new French Merlot on our shelves.Drop by our store for a
free degustation!"
    }
  ]
}
```

借此，您可以通过单一的请求将几条特定的 SMS 消息发送至多个目标地址。

您收到的回复将包含关于已发出的所有消息的信息：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "bulkId":"f5c4322c-10e7-a41e-5528-34fa0b032134",
  "messages":[
    {
      "to":"41793026727",
      "status":{"
        "id":0,
        "groupId":0,
        "groupName":"ACCEPTED",
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      }},
      "smsCount":1,
      "messageId":"4a54f0242f19-b832-1c39-a7e7a2095f351ed2"
    },
    {
      "to":"41793026834",
      "status":{"
        "id":0,
        "groupId":0,
        "groupName":"ACCEPTED",
        "name":"MESSAGE_ACCEPTED",
        "description":"Message accepted"
      }},
      "smsCount":1,
      "messageId":"9404a69cef19-7a31-ba39-92ace76a5f351ed2"
    }
  ]
}
```

发送这些消息后，您能够获取详细的统计信息并执行流量分析。例如，您可以通过获取发送报告来衡量收到 SMS 邀请的客户数量。

设定 SMS 的时间和有效期

如果您想要设定 SMS 时间以便稍后发送，可以使用之前提及的全功能文本消息 API 方法中的 `sendAt` 参数。这将设定发送消息的具体日期和时间。SMS 定时的日期和时间格式：`2015-07-07T17:00:00.000+01:00`。

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNIc2FtZQ==
Content-Type:application/json

{
  "messages":[
    {
      "from":"41793026700",
      "destinations":[
        {
          "to":"41793026785"
        }
      ],
      "text":"A long time ago, in a galaxy far, far away...",
      "sendAt":"2015-07-07T17:00:00.000+01:00"
    }
  ]
}
```

您将收到如下回复：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json
{
  "bulkId":"b86c5f0f-40ed-47b7-9b7f-57eb9707b104",
  "messages":[
    {
      "to":"41793026785",
      "status":{
        "groupId":1,
        "groupName":"PENDING",
        "id":26,
        "name":"PENDING_ACCEPTED",
        "description":"Pending Accepted"
      },
      "messageId":"36397015-149a-41de-bccc-f7e365a7f89a"
    }
  ]
}
```

除了设定消息时间以外，您还可以为每条使用全功能文本消息 API 方法发送的 SMS 设置有效期。

消息 `validityPeriod` 参数应以分钟为单位进行设置。所设期限过期后将无法发送消息。最长有效期为 48 小时；如果您输入更长的时间，我们会将其自动设置为 48 小时。

下面是如何为您的消息设置有效期的示例：

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNLc2FtZQ==
Content-Type:application/json

{
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727",
        },
        {
          "to":"41793026731"
        }
      ],
      "text":"The time-traveling is just too dangerous.Better that I devote myself to study the other great mystery of the universe: women!",
      "validityPeriod":1440
    }
  ]
}
```

您将收到如下回复：

JSON

```
HTTP/1.1 200 OK
Content-Type:application/json

{
  "bulkId": "a6e6bf50-ade1-4dd5-8900-d5b3bb18c0cc",
  "messages": [
    {
      "to": "41793026727",
      "status": {
        "groupId": 1,
        "groupName": "PENDING",
        "id": 7,
        "name": "PENDING_ENROUTE",
        "description": "Message sent to next instance"
      },
      "smsCount": 1,
      "messageId": "239a1bed-91d0-4454-a437-6663938465aa"
    },
    {
      "to": "41793026731",
      "status": {
        "groupId": 1,
        "groupName": "PENDING",
        "id": 7,
        "name": "PENDING_ENROUTE",
        "description": "Message sent to next instance"
      },
      "smsCount": 1,
      "messageId": "77a6e601-c12d-4a66-8e16-d8ec8c5fbc03"
    }
  ]
}
```

通知 URL 上的发送报告

在 DR API 方法中，需要在回复中提取和接收报告，与之不同的是，您能够在自己的回调服务器上设置通知 URL，而我们会在该 URL 上推送发送报告。可将通知 URL 设置为全功能文本消息 API 方法的参数之一。

Infobip 系统收到已发送消息的发送报告后，会将这些报告转发至回调服务器上的指定通知 URL。除了通知 URL 以外，您也可以指定发送报告的通知内容类型。

支持的内容类型：

- application/json
- application/xml

您可以为每条已发送消息设置自定义批量 ID 和消息 ID，使在通知 URL 上推送的每个发送报告的 `messageId` 和 `bulkId` 属性都与所发送的相应消息相同。如果您不使用自定义 `messageId` 和 `bulkId`，Infobip 系统将为推送的发送报告生成这些属性。

除了可标识已发送消息的自定义批量 ID 和消息 ID 以外，您还能够设置 `callbackData`，作为在通知 URL 上发送的其他用户定义的数据。还可将回调数据设置为全功能文本消息 API 方法的参数之一。

下面的示例展示了如何设置 `notifyURL`、发送报告的 `notifyContentType` 以及用户的 `callbackData`。

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "bulkId":"BULK-ID-123-xyz",
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727",
          "messageId":"MESSAGE-ID-123-xyz"
        },
        {
          "to":"41793026731"
        }
      ],
      "text":"Mama always said life was like a box of chocolates.You never know what you
're gonna get.",
      "notifyUrl":"http://www.example.com/sms/advanced",
      "notifyContentType":"application/json",
      "callbackData":"There's no place like home."
    }
  ]
}
```

您在通知 URL 上接收的结果将与通过 API 方法获取的发送报告相同（XML 根元素名称除外，该名称在所有报告中都相同，即 `<reportResponse>`）。

JSON

```
{
  "results":[
    {
      "bulkId":"BULK-ID-123-xyz",
      "messageId":"c9823180-94d4-4ea0-9bf3-ec907e7534a6",
      "to":"41793026731",
      "sentAt":"2015-06-04T13:01:52.933+0000",
      "doneAt":"2015-06-04T13:02:00.134+0000",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.0001000000,
        "currency":"EUR"
      },
      "status":{
        "groupId":3,

```

```

    "groupName": "DELIVERED",
    "id": 5,

    "name": "DELIVERED_TO_HANDSET",
    "description": "Message delivered to handset"
  },
  "error": {
    "groupId": 0,
    "groupName": "OK",
    "id": 0,
    "name": "NO_ERROR",
    "description": "No Error",
    "permanent": false
  },
  "callbackData": "There's no place like home."
},
{
  "bulkId": "BULK-ID-123-xyz",
  "messageId": "MESSAGE-ID-123-xyz",
  "to": "41793026727",
  "sentAt": "2015-06-04T13:01:52.937+0000",
  "doneAt": "2015-06-04T13:02:01.204+0000",
  "smsCount": 1,
  "price": {
    "pricePerMessage": 0.0001000000,
    "currency": "EUR"
  },
  "status": {
    "groupId": 3,
    "groupName": "DELIVERED",
    "id": 5,
    "name": "DELIVERED_TO_HANDSET",
    "description": "Message delivered to handset"
  },
  "error": {
    "groupId": 0,
    "groupName": "OK",
    "id": 0,
    "name": "NO_ERROR",
    "description": "No Error",
    "permanent": false
  },
  "callbackData": "There's no place like home."
}
]
}

```


发送报告推送重试周期

如果您的通知 URL 出于任何原因而不可用，系统将根据公式继续尝试： $1\text{min} + (1\text{min} * \langle\text{retryNumber}/\rangle * \langle\text{retryNumber}/\rangle)$ 。前几次重试尝试示例如下表所示。最大重试次数为 20，也就是说，最后一次重试将在首次重试的 41:30 小时后进行。如果您的 URL 在这整个时间段内都不可用，发送报告将会丢失，唯一能够获取报告的方法是获取 SMS 日志。

重试次数	间隔时间	累计时间
0	01 分钟	00:01 小时
1	02 分钟	00:03 小时
2	05 分钟	00:08 小时
3	10 分钟	00:18 小时
4	17 分钟	00:35 小时
5	26 分钟	01:01 小时
6	37 分钟	01:38 小时

中期发送报告

在通知 URL 上收到最终发送报告之前，每当出现任何非永久性 GSM 错误，您都会收到一份实时中期发送报告。

可将中期发送报告设置为全功能文本消息 API 方法的参数之一。它将在与最终发送报告相同的通知 URL 上推送。

Infobip 系统收到任何非永久性 GSM 错误（如 `EC_ABSENT_SUBSCRIBER`）后，会将相应错误转发至您的回调服务器上的指定通知 URL。除 GSM 错误以外，您还将收到实时定价信息、消息状态、网络和国家/地区代码。

注意：有关 GSM 错误代码和消息状态的更多信息，请查阅[回复](#)部分。

下面的示例展示了如何设置即时消息信息的 `intermediateReport`，以及 `notifyURL`、报告的 `notifyContentType` 和用户的 `callbackData`。

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "bulkId":"BULK-ID-123-xyz",
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727",
          "messageId":"MESSAGE-ID-123-xyz"
        },
        {
          "to":"41793026731"
        }
      ]
    },
    "text":"Mama always said life was like a box of chocolates.You never know what you
're gonna get.",
    "intermediateReport":true,
    "notifyUrl":"http://www.example.com/sms/advanced",
```

```

    "notifyContentType":"application/json",
    "callbackData":"There's no place like home."
  }
]
}

```

您在通知 URL 上收到的结果将与 **最终发送报告** 的结构相同。

JSON

```

{
  "results":[
    {
      "bulkId":"BULK-ID-123-xyz",
      "messageId":"c9823180-94d4-4ea0-9bf3-ec907e7534a6",
      "to":"41793026731",
      "sentAt":"2015-10-04T13:01:52.933+0000",
      "doneAt":"2015-10-04T13:02:00.134+0000",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.0001000000,
        "currency":"EUR"
      },
      "status":{
        "groupId":3,
        "groupName":"PENDING",
        "id":1,
        "name":"PENDING_WAITING_DELIVERY",
        "description":"Message sent, waiting for delivery report"
      },
      "error":{
        "groupId":1,
        "groupName":"HANDSET_ERRORS",
        "id":27,
        "name":"EC_ABSENT_SUBSCRIBER",
        "description":"Absent Subscriber",
        "permanent":false
      },
      "callbackData":"There's no place like home."
    },
    {
      "bulkId":"BULK-ID-123-xyz",
      "messageId":"MESSAGE-ID-123-xyz",
      "to":"41793026727",
      "sentAt":"2015-06-04T13:01:52.937+0000",
      "doneAt":"2015-06-04T13:02:01.204+0000",
      "smsCount":1,
      "price":{
        "pricePerMessage":0.0001000000,
        "currency":"EUR"
      },
      "status":{
        "groupId":3,
        "groupName":"PENDING",
        "id":1,
        "name":"PENDING_WAITING_DELIVERY",
        "description":"Message sent, waiting for delivery report"
      },
    }
  ]
}

```

```
"error":{
  "groupId":1,
  "groupName":"HANDSET_ERRORS",
  "id":27,
  "name":"EC_ABSENT_SUBSCRIBER",
  "description":"Absent Subscriber",
  "permanent":false
},
"callbackData":"There's no place like home."
]
}
```

Flash SMS

除了标准 SMS 消息以外，您还能通过全功能文本消息 API 方法发送 **Flash** 消息。

在收到 **Flash SMS** 时，消息会直接显示在用户手机上。消息可存储在手机上，并带有发送人 ID。要发送 Flash 消息，请将 `flash` 参数设置为 `true`。

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727"
        }
      ],
      "text":"Toto, I've got a feeling we're not in Kansas anymore.",
      "flash":true
    }
  ]
}
```

URL 缩短与跟踪解决方案

开发一款应用程序后，下一步要进行宣传。进行宣传最安全可靠的方式之一是使用 SMS 进行，但如果 URL 太长，该怎么办？

URL 缩短与跟踪解决方案 不仅能自动缩短较长的链接，还可以跟踪您的用户点击率。只需将原始 URL 输入消息正文中，然后设置 **track** 参数，剩余操作都将由 Infobip 平台完成。

这些功能可在 **tracking** 对象中进行设置，该对象位于 **全功能文本消息 API** 方法中：

JSON

```
POST /sms/1/text/advanced HTTP/1.1
Host:api.infobip.com
Authorization:Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==
Content-Type:application/json

{
  "bulkId":"BULK-ID-123-xyz",
  "messages":[
    {
      "from":"InfoSMS",
      "destinations":[
        {
          "to":"41793026727",
          "messageId":"MESSAGE-ID-123-xyz"
        },
        {
          "to":"41793026731"
        }
      ],
      "text":"Hey, take a look at this awesome app.Can you beat my score: http://www.example.com/awesomeApp/someUserId"
    },
    {
      "tracking":{
        "track":"URL",
        "type":"SOCIAL_INVITES"
      }
    }
  ]
}
```

用户将收到带有缩短的 URL 的原始 SMS 消息：

Hey, take a look at this awesome app. Can you beat my score: <http://eel.nu/NJxE/cJ9UH>

重要事项：每条已发送消息都会获得唯一的 Infobip 短 URL，此 URL 将占消息的 25 个字符。

用户点击链接后，我们就能自动**向您发送用户手机号码**，以便您立即了解转换的发生时间。

另外，我们会收集点击率和其他有用的信息：

- 用户的手机号码
- 网络前缀
- 国家 / 地区前缀
- 手机操作系统

SMPP API

短消息点对点协议 (SMPP) 是电信行业使用的开放式行业标准协议，用于在短消息服务中心 (SMSC) 和 SMS 应用程序系统之间交换 SMS 消息。该协议为 7 级 TCP/IP 协议，支持快速发送 SMS 消息。

连接应用程序和 Infobip SMPP 服务器的是 **SMPP 3.4 版** (不支持 3.3 版)。

SMPP 参数

名称	说明
<code>system_id</code>	必要 。为每个客户提供。
<code>password</code>	必要 。为每个客户提供。
<code>IP address</code>	必要 。主要连接点：smpp3.infobip.com 次要连接点：smpp1.infobip.com SSL 连接点：smpp2.infobip.com
<code>port</code>	必要 。8888 (主要和次要) /8887 (ssl)
<code>timeout (keep alive or msg)</code>	必要 。30 秒
<code>system_type</code>	可选 。<r:route_code>

重要事项：您可以绑定为**传送人**、**接收人**或**收发人**。要接收发送报告，您必须绑定为**收发人**或**接收人**。仅当路由提供发送报告功能时，您才会收到发送报告。发送报告将向您当前所有能够接收报告的可用会话等量发送 (**收发人**或**接收人**)。

默认情况下，您可以绑定 **4** 个会话。

支持的 PDU：

- bind_transmitter
- bind_receiver
- bind_transceiver
- unbind
- submit_sm
- deliver_sm
- enquire_link

发送报告格式

Format

```
"id:<message_id> sub:<message_sub> dlvr:<message_dlvr>  
submit date:<message_submit_date> done date:<message_done_date>  
stat:<message_stat> err:<message_err>"
```

发送状态 (MESSAGE_STAT)：

- DELIVRD
- EXPIRED
- DELETED
- UNDELIV
- ACCEPTD
- DELIVERY UNKNOWN
- REJECTED

数据编码方案

如果您在发送消息时设置 **DCS 0** 或 **DCS 1**，我们会将其视为 **默认 GSM7 编码**（SMSC 默认字母表或 **IA5**）。

对于 Latin1 (ISO-8859-1)，发送消息时请使用 **DCS 3** 和 **DCS 8** 作为 Unicode (ISO/IEC-10646)。

命令状态与 GSM 错误代码

发送到我们系统中的每个请求都将在 **submit_sm resp PDU** 中收到确认消息，且请求完全符合 SMPP 3.4 版标准。一些专有平台命令状态和特定于平台的发送报告错误代码在“SMPP 命令状态与 GSM 错误代码”部分有所描述。

定时发送

可以使用 **相对时间格式** 通过 SMPP 协议定时发送消息。

示例：

“**070605040302100R**” – 这表示该消息将在距今 7 年 6 个月 5 天 4 小时 3 分 2 秒又十分之一秒后发送。

符合 SMPP 规范的 Number Context

通过使用 Infobip SMPP 帐户，可以请求 **Number Context** 数据(IMSIs)。要使用 Number Context，您可以使用您的默认 **system_id** 和 **password**，在 Bind PDU 中设置 **system_type = "HLR"**（不使用引号）。

SubmitSM PDU 用于提交 Number Context 请求，从而在所需目标地址中设置 **destAddress** 参数。将忽略所有其他参数（**srcAddress**、**TON/NPI** 等）。

Infobip Number Context 子系统将使用定期 **SubmitSMResp**（包含 **message-id** 引用）进行回复。

Infobip 系统上的 Number Context 请求最终确定后，您将收到 DeliverSM PDU，其中包含：

- 所需 **destAddress** 的 **IMSI**，或
- **错误代码**（若失败）。

DeliverSM 将包含：

- 短消息数据和定期发送报告
- **IMSI** 部分（“**IMSI:xxxxxxxx**”）
- 服务 **MSC**
- 其他**可选信息字段**（取决于您的程序包）。

可选信息字段	类型	十六进制	十进制
原始网络名称	TLVString	0x1412	5138
原始网络前缀	TLVString	0x140B	5131
原始国家 / 地区	TLVString	0x1422	5154
原始国家 / 地区代码	TLVString	0x1423	5155
原始国家 / 地区前缀	TLVString	0x1424	5156
移植网络名称	TLVString	0x1413	5139
移植国家 / 地区前缀	TLVString	0x1442	5186
移植网络前缀	TLVString	0x143e	5182
移植网络国家 / 地区名称	TLVString	0x143f	5183
号码是否移植	TLVInt	0x1421	5153
漫游网络名称	TLVString	0x1414	5140
漫游网络前缀	TLVString	0x1419	5145
漫游国家 / 地区名称	TLVString	0x1415	5141
漫游国家 / 地区代码	TLVString	0x1417	5143
漫游国家 / 地区前缀	TLVString	0x1420	5152
MCCMNC	TLVString	0x1416	5142
消息单价 出于兼容原因，消息单价乘以 100	TLVInt	0x1418	5144
服务 HLR	TLVString	0x1409	5129
号码是否正确	TLVInt	0x1425	5157

说明：除了 DeliverSM.shortMessage 以外，我们还添加了 IMSI 作为额外可选参数：

SMPP_VENDOR_SPECIFIC_IMSI = 0x1403

示例：

如果 Number Context 请求成功，DeliverSM 将如下所示(**IMSI 21910110053751**)：

Groovy

```
addr:0 0 38591xxxxxxx
addr:0 0 000000000

msg:id:40072910491427628 sub:001 dlvrld:001 submit date:1007291049 done date:1007291049 sta
t:DELIVRD err:000
IMSI:219101100935850 MSC:38591016 HLR:38591xxxxxxx ORN:VipNet PON:VipNet RON:VipNet ROC:HR
MCCMNC:21910

opt:(oct:(tlv:1059) 030000) (byte:(tlv:1063) 2) (str:(tlv:30) 40072910491427628) (str:(tlv:
5129)38591xxxxxxx) (str:(tlv:5138) VipNet) (str:(tlv:5139) VipNet) (str:(tlv:5140) VipNet)
(str:(tlv:5141) Croatia ) (str:(tlv:5143) HR) (str:(tlv:5142) 21910) (int:(tlv:5144) 1) (st
r:(tlv:5145) 91) (str:(tlv:5152) 385) (int:(tlv:5153) 1) (str:(tlv:5154) Croatia ) (str:(tl
v:5155) HR) (str:(tlv:5156) 385) (int:(tlv:5157) 1) ) (extraopt:(oct:(tlv:5123) 32313931303
1313030393335383530) (oct:(tlv:5126) 3338353931303136) )
```

如果出现错误，DeliverSM 将如下所示：

Groovy

```
addr:0 0 385915369423
addr:0 0 000000000

msg:id:40072910491419819 sub:001 dlvrld:001 submit date:1007291049 done date:1007291049 sta
t:UNDELIV err:001
IMSI:MSC:ORN:VipNet MCCMNC:

opt:(oct:(tlv:1059) 030001) (byte:(tlv:1063) 5) (str:(tlv:30) 40072910491419819) (str:(tlv:
5138) VipNet) (str:(tlv:5142) ) (int:(tlv:5144) 1) (int:(tlv:5153) 0) (str:(tlv:5154) Croat
ia ) (str:(tlv:5155) HR) (str:(tlv:5156) 385) (int:(tlv:5157) 1) )
```

符合 SMPP 规范的 Flash 通知

您可以使用 Infobip SMPP 帐户发送 **Flash 通知**。此类通知在到达时会立即显示在手机屏幕上，且不会存储在设备内存中。

要使用 Flash 通知，您可以使用您的默认 `system_id` 和 `password`，在 Bind PDU 中设置 `system_type = "NSMS"`（不使用引号）。

提交 Flash 通知的步骤与普通 SMS 完全相同，都使用 SubmitSM PDU。Infobip 系统将使用您提交的消息参数自动将消息转换为 Flash 通知。

系统将使用 DeliverSM PDU 将发送报告发送给您。

注意：Flash 通知不支持长 SMS 功能。

SMPP 命令状态与 GSM 错误代码

SMPP 命令状态

在特殊情况下，您将收到命令状态作为 **Submit_SM** 的回复，具体见下表。

值 (十六进制 / 十进制)	说明
0x00000022/34	网络未覆盖
0x000000FF/255	帐户余额不足
0x0000000a	Invalid_Source_Address
0x0000000c	Duplicate_Message_ID
0x000004a1	System_Error 或 Channel_Disabled

SMPP GSM 错误代码

Infobip 可能提供的 SMPP GSM 错误。

Id	简要说明	permanent 状态
0	NO_ERROR	无
1	EC_UNKNOWN_SUBSCRIBER	1
5	EC_UNIDENTIFIED_SUBSCRIBER	0
6	EC_ABSENT_SUBSCRIBER_SM	0
9	EC_ILLEGAL_SUBSCRIBER	1
10	EC_BEARER_SERVICE_NOT_PROVISIONED	0
11	EC_TELESERVICE_NOT_PROVISIONED	1
12	EC_ILLEGAL_EQUIPMENT	1
13	EC_CALL_BARRED	0
20	EC_SS_INCOMPATIBILITY	0
21	EC_FACILITY_NOT_SUPPORTED	0

Id	简要说明	permanent 状态
27	EC_ABSENT_SUBSCRIBER	0
31	EC_SUBSCRIBER_BUSY_FOR_MT_SMS	0
32	EC_SM_DELIVERY_FAILURE	0
33	EC_MESSAGE_WAITING_LIST_FULL	0
34	EC_SYSTEM_FAILURE	0
35	EC_DATA_MISSING	1
36	EC_UNEXPECTED_DATA_VALUE	1
51	EC_RESOURCE_LIMITATION	0
71	EC_UNKNOWN_ALPHABET	1
72	EC_USSD_BUSY	1
255	EC_UNKNOWN_ERROR	1
256	EC_SM_DF_memoryCapacityExceeded	0
257	EC_SM_DF_equipmentProtocolError	0
258	EC_SM_DF_equipmentNotSM_Equipped	0
259	EC_SM_DF_unknownServiceCentre	0
260	EC_SM_DF_sc_Congestion	0
261	EC_SM_DF_invalidSME_Address	0
262	EC_SM_DF_subscriberNotSC_Subscriber	0
500	EC_PROVIDER_GENERAL_ERROR	0
502	EC_NO_RESPONSE	0
503	EC_SERVICE_COMPLETION_FAILURE	0
504	EC_UNEXPECTED_RESPONSE_FROM_PEER	0
507	EC_MISTYPED_PARAMETER	0
508	EC_NOT_SUPPORTED_SERVICE	0
509	EC_DUPLICATED_INVOKE_ID	0
511	EC_INITIATING_RELEASE	0
1024	EC_OR_appContextNotSupported	0
1025	EC_OR_invalidDestinationReference	0
1026	EC_OR_invalidOriginatingReference	0

Id	简要说明	permanent 状态
1027	EC_OR_encapsulatedAC_NotSupported	0
1028	EC_OR_transportProtectionNotAdequate	0
1029	EC_OR_noReasonGiven	0
1030	EC_OR_potentialVersionIncompatibility	0
1031	EC_OR_remoteNodeNotReachable	0
1152	EC_NNR_noTranslationForAnAddressOfSuchNature	0
1153	EC_NNR_noTranslationForThisSpecificAddress	0
1154	EC_NNR_subsystemCongestion	0
1155	EC_NNR_subsystemFailure	0
1156	EC_NNR_unequippedUser	0
1157	EC_NNR_MTPfailure	0
1158	EC_NNR_networkCongestion	0
1159	EC_NNR_unqualified	0
1160	EC_NNR_errorInMessageTransportXUDT	0
1161	EC_NNR_errorInLocalProcessingXUDT	0
1162	EC_NNR_destinationCannotPerformReassemblyXUDT	0
1163	EC_NNR_SCCPfailure	0
1164	EC_NNR_hopCounterViolation	0
1165	EC_NNR_segmentationNotSupported	0
1166	EC_NNR_segmentationFailure	0
1281	EC_UA_userSpecificReason	0
1282	EC_UA_userResourceLimitation	0
1283	EC_UA_resourceUnavailable	0
1284	EC_UA_applicationProcedureCancellation	0
1536	EC_PA_providerMalfunction	0
1537	EC_PA_supportingDialogOrTransactionReleased	0
1538	EC_PA_ressourceLimitation	0
1539	EC_PA_maintenanceActivity	0
1540	EC_PA_versionIncompatibility	0

Id	简要说明	permanent 状态
1541	EC_PA_abnormalMapDialog	0
1792	EC_NC_abnormalEventDetectedByPeer	0
1793	EC_NC_responseRejectedByPeer	0
1794	EC_NC_abnormalEventReceivedFromPeer	0
1795	EC_NC_messageCannotBeDeliveredToPeer	0
1796	EC_NC_providerOutOfInvoke	0
2048	EC_TIME_OUT	0
2049	EC_IMSI_BLACKLISTED	1
2050	EC_DEST_ADDRESS_BLACKLISTED	1
2051	EC_InvalidMscAddress	0
4096	EC_invalidPduFurmat	1
4097	EC_NotSubmittedToGMSC	1
4100	EC_Cancelled	1
4101	EC_ValidityExpired	1
4102	EC_NotSubmittedToSmppChannel	0