

VOLTE-SIP 代码意义

SIP 应答消息状态码与功能

类型 状态码 状态说明

临时应答(1XX) 100 Trying 正在处理中

180 Ringing 振铃

181 call being forwarder 呼叫正在前向

182 queue 排队

181* session progress 会话进行

会话成功(2XX) 200 OK 会话成功

重定向(3XX) 300 multiple 多重选择

301 moved permanently 永久移动

302 moved temporarily 临时移动

305 use proxy 用户代理

380 alternative service 替代服务

请求失败(4XX) 400 bad request 错误请求

401 unauthorized 未授权

402 payment required 付费要求

403 forbidden 禁止

404 not found 未发现

405 method not allowed 方法不允许

406 not acceptable 不可接受

407 proxy authentication required 代理需要认证

408 request timeout 请求超时

410 gone 离开

413 request entity too large 请求实体太大

414 request-url too long 请求 URL 太长

415 unsupported media type 不支持的媒体类型

416 unsupported url scheme 不支持的 URL 计划

420 bad extension 不良扩展

421 extension required 需要扩展

423 interval too brief 间隔太短

480 temporarily unavailable 临时失效

481 call/transaction does not exist 呼叫/事务不存在

482 loop detected 发现环路

483 too many hops 跳数太多

484 address incomplete 地址不完整

485 ambiguous 不明朗

486 busy here 这里忙

487 request terminated 请求终止

488 not acceptable here 这里请求不可接受

491 request pending 未决请求
493 undecipherable 不可辨识
服务器失败(5XX) 500 server internal error 服务器内部错误
501 not implemented 不可执行
502 bad gateway 坏网关
503 service unavailable 服务无效
504 server time-out 服务器超时
505 version not supported 版本不支持
513 message too large 消息太大
全局性错误(6XX) 600 busy everywhere 全忙
603 decline 丢弃
604 does not exist anywhere 不存在
606 not acceptable 不可接受
SIP 应答代码(以下是详细内容)

应答码是包含了, 并且扩展了 HTTP/1.1 应答码。并不是所有的 HTTP/1.1 应答码都适当应用, 只有在折里指出的是适当的。其他 HTTP/1.1 应答码不应当使用。并且, SIP 也定义了新的应答码系列, 6xx。

1 临时应答 1xx

临时应答, 也就是消息性质的应答, 标志了对方服务器正在处理请求, 并且还没有决定最后的应答。如果服务器处理请求需要花 200ms 以上才能产生终结应答的时候, 它应当发送一个 1xx 应答。

注意 1xx 应答并不是可靠传输的。他们不会导致客户端传送一个 ACK 应答。临时性质的(1xx) 应答可以包含消息体, 包含会话描述。

1.1 100 Trying

这个应答表示下一个节点的服务器已经接收到了这个请求并且还没有执行这个请求的特定动作(比如, 正在打开数据库的时候)。这个应答, 就像其他临时应答一样, 种植了 UAC 重新传送 INVITE 请求。100(Trying)应答和其他临时应答不同的是, 在这里, 它永远不会被有状态 proxy 转发到上行流中。

1.2 180 Ringing

UA 收到 INVITE 请求并且试图提示给用户。这个应答应当出世化一个本地回铃。

1.3 818 Call is Being Forwarded(呼叫被转发)

服务器可以用这个应答代码来表示呼叫正在转发到另一个目的地集合。

1.4 182 Queued

当 呼叫的对方暂时不能接收呼叫的时候, 并且服务器决定将呼叫排队等候, 而不是拒绝呼叫的时候, 那么就应当发出这个应答。当被叫方一旦恢复接收呼叫, 他会返回 合适的终结应答。对于这个呼叫状态, 可以有一个表示原因的短语, 比如: "5 calls queued;expected waiting time is 15minutes"。服务器可以给出好几个 182 (Queued) 应答告诉呼叫方排队的情况(比如排队靠前了等等)。

1.5 183 会话进度

183 (Session Progress) 应答用于提示建立对话的进度信息。Reason-Phrase (表达原因的句子)、头域或者消息体可以用于提示呼叫进度的更消息的信息。

2 成功信息 2xx

这个应答表示请求是成功的。

2.1 200 OK

请求已经处理成功。这个信息取决于不同方法的请求的应答。

3 转发请求 3XX

3xx 系列的应答是用于提示用户的新位置信息的,或者为了满足呼叫而转发的额外服务地点。

3.1 300 Multiple Choices

请求的地址有多个选择,每个选择都有自己的地址,用户或者(UA)可以选择合适的通讯终端,并且转发这个请求到这个地址。

应答可以包含一个具有每一个地点的在 Accept 请求头域中允许的资源特性,这样用户或者 UA 可以选择一个最合适的地址来转发请求。没有未这个应答的消息体定义 MIME 类型。

这些地址选择也应当在 Contact 头域中列出(20.10 节)。不同于 HTTP, SIP 应答可以包含多个 Contact 头域或者一个 Contact 头域中具有一个地址列表。UA 可以使用 Contact 头域来自自动转发或者要求用户确认转发。不过,本规范没有定义自动转发的标准。

如果被叫方可以在多个地址被找到,并且服务器不能或者不愿意转发请求的时候,可以使用这个应答来给呼叫方。

3.2 301 Moved Permently

当不能在 Request-URI 指定的地址找到用户的时候,请求的客户端应当使用 Contact 头域(20.10)所指出的新的地址重新尝试。请求者应当用这个新的值来更新本地的目录,地址本,和用户地址 cache,并且在后续请求中,发送到这个/这些列出的地址。

3.3 302 Moved Temporarily

请求方应当把请求重新发到这个 Contact 头域所指出的新地址(20.10)。新请求的 Request-URI 应当用这个应答的 Contact 头域所指出的值。

在应答中的 Expires(20.19 节)或者 Contact 头域的 expires 参数定义了这个 Contact URI 的生存周期。UA 或者 proxy 在这个生存周期内 cache 这个 URI。如果没有严格的有效时见,那么这个地址仅仅本次有效,并且不能在以后的事务中保存。

如果 cache 的 Contact 头域的值失败了,那么被转发请求的 Request-URI 应当再次尝试一次。临时 URI 可以比超时时间更快的失效,并且可以有一个新的临时 URI。

3.4 305 Use Proxy

请求的资源必须通过 Contact 头域中指出的 proxy 来访问。Contact 头域指定了一个 proxy 的 URI。接收到这个应答的对象应当通过这个 proxy 重新发送这个单个请求。305 (UseProxy) 必须是 UAS 产生的。

3.5 380 Alternative Service

呼叫不成工,但是可以尝试另外的服务。另外的服务在应答的消息体中定义。消息体的格式在这里没有定义,可能在以后的规范中定义。

4 请求失败 4xx

4xx 应答定义了特定服务器响应的请求失败的情况。客户端不应当在不更改请求的情况下重新尝试同一个请求。(例如,增加合适的认证信息)。不过,同一个请求交给不同服务器也许就会成功。

4.1 400 Bad Request

请求中的语法错误。Reason-Phrase 应当标志这个详细的语法错误,比如”Missing Call-ID header field”。

4.2 401 Unauthorized

请求需要用户认证。这个应答是由 UAS 和注册服务器产生的,当 407 (Proxy Authentication Required) 是 proxy 服务器产生的。

4.3 402 Payment Required

保留/以后使用

4.4 403 Forbidden

服务端支持这个请求，但是拒绝执行请求。增加验证信息是没有必要的，并且请求应当不被重试。

4.5 404 Not Found

服务器返回最终信息：用户在 **Request-URI** 指定的域上不存在。当 **Request-URI** 的 **domain** 和接收这个请求的 **domain** 不匹配的情况下，也会产生这个应答。

4.6 405 Method Not Allowed

服务器支持 **Request-Line** 中的方法，但是对于这个 **Request-URI** 中的地址来说，是不允许应用这个方法。

应答必须包括一个 **Allow** 头域，这个头域包含了指定地址允许的方法列表。

4.7 Not Acceptable

请求中的资源只会导致产生一个在请求中的 **Accept** 头域外的，内容无法接收的错误。

4.8 407 Proxy Authentication Required

这个返回码和 **401 (Unauthorized)** 很类似，但是标志了客户端应当首先在 **proxy** 上通过认证。**SIP** 对认证的访问请参见 26 节和 22.3 节。

这个返回码用于应用程序访问通讯网关（比如，电话网关），而很少用于被叫方要求认证。

4.9 408 Request Timeout

在一段时间内，服务器不能产生一个终结应答，例如，如果它无法及时决定用户的位置。客户端可以在稍后不更改请求的内容然后重新尝试请求。

4.10 410 Gone

请求的资源在本服务器上已经不存在了，并且不知道应当把请求转发到哪里。这个问题将会使永久性的。如果服务器不知道，或者不容易检测，这个资源消失是临时性质的还是永久性质的，那么应当返回一个 **404 (Not Found)**。

4.11 413 请求实体过大

服务器拒绝处理请求，因为这个请求的实体超过了服务器希望或者能够处理的大小。这个服务器应当关闭连接避免客户端重发这个请求。

如果这个情况是暂时的，那么服务端应当包含一个 **Retry-After** 头域来表明这是一个暂时的故障，并且客户端可以过一段时间再次尝试。

4.12 414 Request-URI Too Long

服务器拒绝这个请求，因为 **Request-URI** 超过了服务器能够处理的长度。

4.13 415 Unsupported Media Type

服务器由于请求的消息体的格式本服务器不支持，所以拒绝处理这个请求。这个服务器必须根据内容的故障类型，返回一个 **Accept**, **Accept-Encoding**, 或者 **Accept-Language** 头域列表。

UAC 根据 8.1.3.5 节定义的方法处理这个应答。

4.14 416 Unsupported URI Scheme

服务器由于不支持 **Request-URI** 中的 **URI** 方案而终止处理这个请求。客户端处理这个应答参照 8.1.3.5。

4.15 Bad Extension

服务器不知道在请求中的 **Proxy-Require(20.29)** 或者 **Require(20.32)** 头域所指出的协议扩展。

服务器必须在 **Unsupported** 头域中列出不支持的扩展。UAC 处理这个应答请参见 8.1.3.5

4.16 421 Extension Required

UAS 需要特定的扩展来处理这个请求,但是这个扩展并没有在请求的 **Supported** 头域中列出。具有这个应答码的应答必须包含一个 **Require** 头域列出所需要的扩展。

UAS 不应当使用这个应答除非它真的不能给客户端提供有效的服务。相反,如果在 **Support** 头域中没有列出需要的扩展,服务器应当根据基准的 **SIP** 兼容的方法和客户端支持的扩展来进行处理。

4.17 423 Interval Too Brief

服务器因为在请求中设置的资源刷新时间(或者有效时间)过短而拒绝请求。这个应答可以用于注册服务器来拒绝那些 **Contact** 头域有效期过短的注册请求。这个应答的用法和相关的 **Min-Expires** 头域在 10.2.8,10.3,20.23 节中介绍和说明。

4.18 480 Temporarily Unavailable

请求成功到达被叫方的终端系统,但是被叫方当前不可用(例如,没有登陆,或者登陆了但是状态是不能通讯,或者有“请勿打扰”的标记)。应答应当在 **Retry-After** 中标志一个合适的重发时间。这个用户也有可能在其他地方是有效的(在本服务器中不知道)。**Reason-Phrase**(原因短句)应当提示更详细的原因,为什么被叫方暂时不可用。这个值应当是可以被 UA 设置的。状态码 486 (**Busy Here**)可以用来更精确的表示本请求失败的特定原因。

这个状态码也可以是转发服务或者 proxy 服务器返回的,因为他们发现 **Request-URI** 指定的用户存在,但是没有一个给这个用户的合适的当前转发的地址。

4.19 481 Call/Transaction Does Not Exist

这个状态表示了 UAS 接收到请求,但是没有和现存的对话或者事务匹配。

4.20 482 Loop Detected

服务器检测到了一个循环(16.3/4)

4.21 483 Too Many Hops

服务器接收到了一个请求包含的 **Max-Forwards**(20.22)头域是 0

4.22 484 Address InComplete

服务器接收到了一个请求,它的 **Request-URI** 是不完整的。在原因短语中应当有附加的信息说明。这个状态码可以和拨号交叠。在和拨号交叠中,客户端不知道拨号串的长度。它发送增加长度的字串,并且提示用户输入更多的字串,直到不再出现 484 (**Address Incomplete**) 应答为止。

4.23 485 Ambiguous

Request-URI 是不明确的。应答可以在 **Contact** 头域中包含一个可能的明确的地址列表。这个提示列表肯囊个在安全性和隐私性对用户或者组织造成破坏。必须能够由配置决定是否以 404 (**NotFound**) 代替这个应答,又或者禁止对不明确的地址使用可能的选择列表。

给带有 **Request-URI** 的请求的一个应答例子:

sip: lee@example.com:

SIP/2.0 485 Ambiguous

Contact: Carol Lee <sip:carol.lee@example.com>

Contact: Ping Lee <sip:p.lee@example.com>

Contact: Lee M.Foote <sips:lee.foote@example.com>

部分 email 和语音邮箱系统提供了这个功能。这个状态码和 3xx 状态码不同:对于 300 来说,它是假定同一个人或者服务有不同的地址选择。所以对 3xx 来说,自动选择系统或者连续查找就有效,但是对 485 (**Ambiguous**) 应答来说,一定要用户的干预。

4.24 486 Busy Here

当成功联系到被叫方的终端系统，但是被叫方当前在这个终端系统上不能接听这个电话，那么应答应当回给呼叫方一个更合适的时间在 **Retry-After** 头域 重试。这个用户也许在其他地方有效，比如电话邮箱系统等等。如果我们知道没有其他终端系统能够接听这个呼叫，那么应当返回一个状态码 600 (**Busy Everywhere**)。

4.25 487 Request Terminated

请求被 **BYE** 或者 **CANCEL** 所终止。这个应答永远不会给 **CANCEL** 请求本身回复。

4.26 488 Not Acceptable Here

这个应答和 606 (**Not Acceptable**) 有相同的含义，但是只是应用于 **Request-URI** 所指出的特定资源不能接受，在其他地方请求可能可以接受。

包含了媒体兼容性描述的消息体可以出现在应答中，并且根据 **INVITE** 请求中的 **Accept** 头域进行规格化（如果没有 **Accept** 头域，那么就是 **application/sdp**）。这个应答就像给 **OPTIONS** 请求的 200(OK)应答的消息体一样。

4.27 491 Request Pending

在同一个对话中，**UAS** 接收到的请求有一个依赖的请求正在处理。**14.2** 描述了这种情况应当怎样解决。

4.28 493 Undecipherable

UAS 接收到了一个请求，包含了一个加密的 **MIME**，并且不知道或者没有提供合适的解密密钥。这个应答可以包含单个包体，这个包体包含了合适的公钥，这个公钥用于给这个 **UAS** 通讯中加密包体使用的。细节描述在 **23.2** 节。

5 Server Failure 5xx

5xx 应答是当服务器本身故障的时候给出的失败应答。

5.1 500 Server Internal Error

服务器遇到了未知的情况，并且不能继续处理请求。客户端可以显示特定的错误情况，并且可以在几秒钟以后重新尝试这个请求。

如果这个情况是临时的，服务器应当在 **Retry-After** 头域标志客户端过多少秒钟之后重新尝试这个请求。

5.2 501 Not Implemented

服务器没有实现相关的请求功能。当 **UAS** 不认识请求的方法的时候，并且对每一个用户都无法支持这个方法的时候，应当返回这个应答。（**proxy** 不考虑请求的方法而转发请求）。

注意 **405** (**Method Not Allowed**) 是因为服务器实现了这个请求方法，但是这个请求方法在特定请求中不被支持。

5.3 502 Bad Gateway

如果服务器，作为 **gateway** 或者 **proxy** 存在，从下行服务器上接收到了一个非法的应答（这个应答对应的请求是本服务器为了完成请求而转发给下行服务器的）。

5.4 503 Service Unavailable

由于临时的过载或者服务器管理导致的服务器暂时不可用。这个服务器可以在应答中增加一个 **Retry-After** 来让客户端重试这个请求。如果没有 **Retry-After** 指出，客户端必须就像收到了一个 **500** (**Server Internal Error**) 应答一样处理。

客户端（**proxy** 或者 **UAC**）收到 **503** (**Service Unavailable**) 应当尝试转发这个请求到另外一个服务器处理。并且在 **Retry-After** 头域中指定的时间内，不应当转发其他请求到这个服务器。作为 **503**(**Service Unavailable**)的替代，服务器可以拒绝连接或者把请求扔掉。

5.5 504 Server Time-out

服务器在一个外部服务器上没有收到一个及时的应答。这个外部服务器是本服务器用来访问

处理这个请求所需要的。如果从上行服务器上收到的请求中的 Expires 头域超时，那么应当返回一个 408（Request TimeOut）错误。

5.6 505 Version Not Supported

服务器不支持对应的 SIP 版本。服务器是无法处理具有客户端提供的相同主版本号的请求，就会导致这样的错误信息。

5.7 Message Too Large

服务器无法处理请求，因为消息长度超过了处理的长度。

6 Global Failures 6xx

6xx 应答意味着服务器给特定用户有一个最终的信息，并不只是在 Request-URI 的特定实例有最终信息。

6.1 600 Busy Everywhere

成功联系到被叫方的终端系统，但是被叫方处于忙的状态，并不打算接听电话。这个应答可以通过增加一个 Retry-After 头域更明确的告诉呼叫方多久以后可以继续呼叫。如果被叫方不希望提示拒绝的原因，被叫方应当使用 603（Decline）。只有当终端系统知道没有其他终端节点（比如语音邮箱系统）能够访问到这个用户的时候才能使用这个应答。否则应当返回一个 486（Busy Here）的应答。

6.2 603 Decline

当成功访问到被叫方的设备，但是用户明确的不想应答。这个应答可以通过增加一个 Retry-After 头域更明确的告诉呼叫方多久以后可以继续呼叫。只有当终端知道没有其他任何终端设备能够响应这个呼叫的势能才能给出这个应答。

6.3 604 Does Not Exist Anywhere

服务器验证了在请求中 Request-URI 的用户信息，哪里都不存在

6.4 606 Not Acceptable

当成功联系到一个 UA，但是会话描述的一些部分比如请求的媒体，带宽，或者地址类型不被接收。

606（NotAcceptable）应答意味着用户希望通讯，但是不能充分支持会话描述。606（Not Acceptable）应答可以在 Warning 头域中包含一个原因列表，用于解释为何会话描述不能被支持。警告原因代码在 20.43 节中列出。

在应答中，可以出现一个包含媒体兼容性描述的消息体，这个消息体的格式根据 INVITE 请求中的 Accept 头域指出的格式进行规格化（如果没有 Accept 头域，那么就是 application/sdp），就像给 OPTIONS 请求的 200(OK)应答中的消息一样。

我们希望这些媒体协商不要经常需要，并且当一个新用户被邀请加入已经存在的会话的时候，这个媒体协商可能不需要。这取决于邀请的初始化者是否需要处理 606（Not Acceptable）。

这个应答只有当客户端知道没有其他终端能够处理这个请求的时候才能发出。

微信扫描以下二维码，免费加入【5G 俱乐部】，还赠送整套：5G 前沿、NB-IoT、4G+ (VoLTE) 资料。

