交互式连接建立（ICE）  
网络地址转换（NAT）穿透协议

# 摘要

本文档介绍基于UDP的网络地址转换（NAT）穿透协议，称为交互式连接建立（ICE）协议。ICE利用了NAT协议的会话穿透工具（STUN）及其扩展—使用中继的NAT穿透（TURN）。

# 1 简介

在对等节点之间建立通信会话的协议通常需要为发送方和接收方交换IP地址和端口。但是，当通过网络地址转换（NAT）[RFC3235]时该操作存在一定障碍。这些协议还寻求在参与者之间创建直接的数据流，使它们之间没有应用层中介，从而减少数据延迟、降低丢包概率和缩减应用部署成本。但是，通过NAT实现这一点很困难。对此问题的原因进行全面分析不在本规范范围之内。

目前已定义了许多解决方案使协议能够穿透NAT运行。其中包括：应用层网关（ALG）、中间盒控制协议[RFC3303]、原始的简单UDP穿透NAT（STUN）协议[RFC3489]（RFC3489已更新为RFC5389），以及面向Realm的IP[RFC3102, RFC3103]以及使其正常工作所需的会话描述扩展，如实时控制协议（RTCP）[RFC3605] 的会话描述协议（SDP）属性[RFC4566]。不幸的是，这些技术都有优缺点，使某些技术在某些网络拓扑中是最佳的，但在其他网络拓扑中却是糟糕的选择。因此，网络管理员和实施者不得不对其解决方案部署的网络拓扑进行假设。导致系统的复杂性和脆弱性增加。

本规范将交互式连接建立（ICE）定义为基于UDP数据流的NAT穿透技术（尽管ICE已扩展可以处理其他传输协议，如TCP[RFC6544]）。ICE的基本原理是交换多个IP地址和端口，然后通过对等连接检查进行测试。IP地址和端口使用ICE专门的机制（例如要约/应答交换）进行交换，连接检查使用STUN[RFC5389]实现。ICE还利用了针对NAT的中继（TURN）[RFC5766]，这是 STUN 的扩展。由于ICE为每个媒体流交换多个IP地址和端口，因此它还允许为多宿主机和双堆栈主机选择地址。因此，RFC 5245弃用了RFC 4091和RFC 4092中先前定义的解决方案。

附录B提供有关设计ICE时所做的设计决策的背景信息和动机。

# 2 ICE概述

在典型的ICE部署中，有两个终结点（ICE代理）需要通信。请注意，ICE不适用于信令协议的NAT穿透，假定该信令协议通过其它机制提供。并且ICE假定代理之间能够建立信令连接。

一开始代理对自己的拓扑一无所知。特别是，代理可能位于或可能不位于NAT（或多层NAT）后面。ICE允许代理发现关于其拓扑的足够信息，从而能够找到一个或多个路径，以便代理之间可以建立数据会话。

图1显示了典型的ICE部署场景。代理标记为L和R。L和R可能都在各自的NAT后面，尽管他们可能没有意识到这一点。NAT的类型及其属性也未知。L和R能够参与候选地址交换过程，其目的是在L和R之间建立数据会话。通常，此交换将通过信令服务器（例如，SIP代理）完成。

除了代理、信令服务器和NAT之外，ICE通常与网络中的STUN或TURN服务器协同使用。每个代理可以有自己的STUN或TURN服务器，也可以共用。



图1: ICE部署场景

ICE的基本思想是：每个代理具有各种候选传输地址（特定传输协议的IP地址和端口的组合，在此规范中始终是UDP），可以用来与其他代理通信。

这些候选地址可能包括：

* 直接连接的网络接口上的传输地址
* NAT公共端的翻译传输地址（"服务器反射式"地址）
* 从TURN服务器分配的传输地址（"中继地址"）

任何L的候选传输地址都可能用于与R的任何候选传输地址进行通信。然而，在实践中，许多组合将不起作用。例如，如果L和R都位于NAT之后，则直接连接的接口地址不太可能直接通信（这就是为什么终究需要ICE的原因！ICE的目的是发现哪些地址对可用。ICE这样做的目的是系统地尝试所有可能的对（按仔细排序的顺序），直到找到一个或多个可用的组合。

## 2.1 采集候选项

为了执行ICE，ICE代理标识并采集一个或多个地址候选项。候选项具有传输地址——特定传输协议的IP地址和端口的组合（此处仅指定UDP）。有不同类型的候选项：有些来自物理或逻辑网络接口，另一些可以通过STUN和TURN发现。

第一类候选项是直接从本地接口获取的传输地址。这样的候选项被称为"主机候选项"。本地接口可以是以太网或Wi-Fi，也可以是通过隧道机制（如虚拟专用网络（VPN）或移动IP（MIP））获得的接口。在所有情况下，此类网络接口在代理看来是一个本地接口，可以从中分配端口（因此为候选项）。

接下来，代理使用STUN或TURN获取其他候选项。包括两种类型：在NAT公共端（服务器反射候选项）的已翻译地址和TURN服务器上的地址（中继候选项）。使用TURN服务器时，这两种类型的候选服务器都从TURN服务器获取。如果只使用STUN服务器，则仅从中获得服务器反射候选项。图2显示了这些候选项与主机候选项的关系。在此图中，使用TURN发现了这两种类型的候选项。其中，符号X:x表示IP地址X和UDP端口x。



图2: 候选项关系

当代理从IP地址和端口X:x发送TURN分配请求时，NAT（假设有一个）将创建一个绑定X1':x1'，将此服务器反射候选项映射到主机候选项X:x。从主机候选项发送的传出数据包将由NAT转换为服务器反射候选项。发送到服务器反射候选的传入数据包将由NAT转换为主机候选数据包并转发给代理。与给定服务器反射候选项关联的主机候选项称为"Base"。

注意："Base"是指代理为特定候选项发送的地址。因此，作为退化案例，主机候选项也有一个Base，但它与主机候选项相同。

当代理和TURN服务器之间有多个NAT时，TURN请求将在每个NAT上创建绑定，但代理将仅发现最外层的服务器反射候选项（最接近TURN服务器的候选项）。 如果代理不在NAT后面，则Base候选项将与服务器反射候选项相同，并且服务器反射候选项是冗余的，可以被删除。

然后，分配请求到达TURN服务器。TURN服务器从其本地IP地址Y分配端口y，并生成分配响应，通知代理此中继候选项。TURN服务器还将分配请求的源传输地址复制到分配响应中，通知服务器反射候选项X1':x1'的代理。TURN服务器充当数据包中继，在L和R之间转发流量。为了将流量发送到L，R在Y:y将流量发送到TURN服务器，TURN服务器将流量转发到X1':x1'，该服务器通过NAT，其中它被映射到X:x并传递到L。

当只使用STUN服务器时，代理会向其STUN服务器发送STUN绑定请求[RFC5389]。STUN服务器通过将绑定请求的源传输地址复制到绑定响应中，通知代理服务器反射候选项X1':x1'。

## 2.2 连接检查

一旦L采集了所有候选项，将按最高优先级到最低优先级进行排序，并通过信令信道将它们发送给R。当R从L接收候选项时，将执行相同的采集过程，并应答自己的候选项列表。在此过程结束时，每个ICE代理都有其候选项和对等节点候选项的完整列表。代理对它们进行配对，形成候选对。要查看哪些候选对可以工作，每个代理会安排一系列连接检查。每个检查都是一个STUN要约/应答事务，客户端通过将本地候选项的STUN请求发送到远程候选项，对特定候选项执行该要约/应答事务。

连接检查的基本原则很简单：

1. 按优先级顺序对候选对进行排序。

2. 按优先级顺序发送每个候选对的检查。

3. 确认从其他代理收到的检查。

当两个代理对候选对执行检查时，结果是4向握手：



图3：基本连接检查

务必注意，STUN请求利用与数据传输完全相同的IP地址和端口（例如 RTP、RTCP 或其他协议）发送。因此，代理针对数据包的内容（而不是接收数据包的端口）进行STUN和数据的解复用。

由于STUN绑定请求用于连接检查，因此STUN绑定响应将包含代理与其对等节点之间的任何NAT的公共侧的代理翻译传输地址。如果此传输地址不同于代理已采集的其他候选项，则它代表一个新的候选项（对等反射候选项），然后由ICE执行与任何其他候选项相同的检查。

由于上述算法搜索所有候选对，如果存在工作对，则无论候选项尝试的顺序如何，算法最终都会找到它。为了更快（更好）得生成结果，候选项按指定顺序排序。生成的已排序候选对的列表称为"清单"。

代理通过定期发送列表中下一对候选对的STUN请求来完成清单。这些称为"普通检查"。当STUN事务成功时，一个或多个候选对将成为所谓的"有效对"，并添加到称为"有效列表"的候选对列表中。

作为优化，R一旦收到L的检查消息，就会安排在同一候选对上将连接检查消息发送到L。这称为"触发检查"，它加快了查找有效对的过程。

在此握手结束时，L和R都知道他们可以从两个方向端到端地发送（和接收）消息。

通常，优先级算法的设计使相似类型的候选项获得类似的优先级，以便更直接的路由（即没有数据中继或NAT的路由）优先于间接路由（具有数据中继或NAT的路由）。但是，在这些准则中，代理在如何调整其算法方面拥有相当的酌处权。

数据流可能由多个组件（需要其自己的候选项的数据流部分，例如RTP和RTCP）组成。

## 2.3 提名候选项对和结束ICE

ICE分配一个ICE代理担任控制代理的角色，另一个分配受控代理的角色。对于数据流的每个组件，控制代理提名一个有效对（来自有效列表）用于数据通信。提名的确切时间依赖于本地策略。

进行提名时，控制代理允许继续检查，直到找到数据流的每个组件至少一个有效对，然后选择一个有效的对，通过该对发送STUN请求，并使用属性向受控对等节点指示已提名该对。如图4所示。



图4：提名

一旦受控代理收到具有该属性的STUN请求，将检查（除非检查已完成）同一对。如果上述事务成功，代理将为该有效对设置提名标志，并取消数据流中该组件将来的任何检查。一旦代理为数据流的每个组件设置了提名标志，提名对将成为选定对。之后，将仅使用选定对来发送和接收与该数据流关联的数据。

## 2.4 ICE重新启动

ICE完成后，任何ICE代理都可以在任何时候重新启动一个或所有数据流。这通过发送指示重新启动的更新候选项信息来实现。

## 2.5 lite实现

某些ICE代理始终连接到公共Internet，并具有公共IP地址，以便接收来自任何代理的数据包。为了使这些设备更容易支持ICE，ICE定义了一种称为"lite"的特殊实现类型（与正常的完整实现相对）。lite代理仅使用主机候选项，不生成连接检查或运行状态机，尽管它们需要能够响应连接检查。

# 3 ICE使用

本文档指定ICE的通用使用，提供在ICE代理之间交换候选项信息的协议。使用ICE（称为"使用协议"）的不同协议的具体细节（即如何编码候选项信息和实际候选项交换流程）在单独的使用文档中进行了说明。

允许代理交换候选项信息的一种机制是使用作为SIP协议的一部分[RFC3261] [ICE-SIP-SDP]的要约/应答语义（基于[RFC3264]）。

[RFC7825]定义了实时流处理协议（RTSP）的ICE用法。但是请注意，ICE的用法基于RFC 5245。

# 4 术语

* ICE会话：ICE会话由从候选项采集开始的所有ICE相关操作组成，然后是ICE代理之间的交互（候选项交换、连接检查、提名和保活），直到发布所有候选项或触发ICE重新启动。
* ICE代理：ICE代理（有时简称为"代理"）是ICE候选项交换中涉及的协议实现。典型的候选项交换过程包含两个代理。
* 启动对等节点、启动代理、启动器：启动代理是启动ICE候选项交换过程的ICE代理。
* 响应对等节点、响应代理、响应者：响应代理是接收并响应启动代理启动的候选项交换过程的ICE代理。
* ICE候选项交换、候选项交换：ICE代理交换执行ICE所需的信息（例如，候选项和口令）的过程。要约/应答与SDP编码[RFC3264]是可用于交换候选项信息的协议的一个示例。
* 对等节点：从会话中一个ICE代理的角度来看，其对等节点是另一个代理。具体来说，从启动代理的角度来看，对等节点是响应代理。从响应代理的角度来看，对等节点是启动代理。
* 传输地址：IP地址和传输协议（如UDP或TCP）端口的组合。
* 数据、数据流、数据会话：当ICE用于设置数据会话时，将使用某种协议传输数据。媒体通常通过RTP传输，由RTP数据包流组成。数据会话是指在使用ICE创建和测试的路径上的对等节点之间交换的数据包。
* 候选项，候选项信息：即传输地址，是接收数据的潜在联系点。候选项也有属性，包括：类型（服务器反射、中继或主机）、优先级、基础和Base。
* 组件：组件是数据流的一部分。数据流可能包括多个组件，每个组件必须工作才能使整个数据流正常工作。对于RTP/RTCP数据流，除非RTP和RTCP在同一端口中复用，否则每个数据流有两个组件：一个用于RTP，另一个用于RTCP。组件具有候选对，其他组件不能使用该候选对。
* 主机候选项：通过绑定到主机上的IP地址的特定端口获得的候选项。包括物理接口和逻辑接口上的IP地址，例如通过VPN获取的IP地址。
* 服务器反射候选项：是NAT为ICE代理分配的IP地址和端口绑定，在通过NAT将数据包发送到服务器（如STUN服务器）后创建。
* 对等反射候选项：是NAT为ICE代理分配的IP地址和端口绑定，在通过NAT向其对等节点发送数据包后创建。
* 中继候选项：从中继服务器（如TURN服务器）获得的候选项。
* Base：ICE代理为特定候选项发送的传输地址。对于主机、服务器反射和对等反射候选项，Base与主机候选项相同。对于中继的候选项，Base与中继候选项相同（即TURN服务器使用的传输地址发送）。
* 相关地址和端口：与候选项相关的传输地址，可用于诊断和其他目的。如果候选项是服务器或对等反射，则相关地址和端口等于该服务器或对等反射候选项的Base。如果是中继候选项，则相关地址和端口等于为客户端提供中继候选项的"分配"响应中的映射地址。如果候选项是主机候选项，则相关地址和端口与主机候选项相同。
* 基础：冻结算法中使用的任意字符串，用于对相似候选项进行分组。对于具有相同类型、Base IP地址、协议（UDP、TCP等）和STUN或TURN服务器的两个候选项，则基础相同。如果这些不同，则基础不同。
* 本地候选项：ICE代理已获取并可发送给其对等节点的候选项。
* 远程候选项：ICE代理从其对等节点收到的候选项。
* 默认目标/候选：数据流组件的默认目标是ICE代理将使用的传输地址，而ICE代理则不知道。组件的默认候选项是传输地址与该组件的默认目标匹配的候选项。
* 候选对：包含本地候选项和远程候选项的对。
* 检查、连接检查、STUN检查：用于验证连接的STUN绑定请求。检查从本地候选项的Base发送到候选对的远程候选项。
* 清单：ICE代理用于生成检查的一组有序的候选对。
* 普通检查：ICE代理生成的连接检查，由计时器定期触发，指示代理发送检查。
* 触发检查：由于从对等节点收到连接检查而生成的连接检查。
* 有效对：其本地候选项等于成功连接检查相应的映射地址，其远程候选项等于向其发送连接检查请求的目标地址的候选对。
* 有效列表：已成功通过STUN事务验证的数据流的一组有序候选对。
* 清单集：所有清单的有序列表。顺序由每个ICE使用情况确定。
* 完整实现：执行此规范定义的完整功能集的ICE实现。
* lite实现：一种ICE实现，它省略了某些功能，仅实现使对等节点获得非lite实现类似的ICE好处的功能。lite实现不维护任何状态机，也不会生成连接检查。
* 控制代理：指定候选对的ICE代理。在任何会话中，始终有一个控制代理和一个受控代理。
* 受控代理：等待控制代理提名候选对的ICE代理。
* 提名：控制代理向受控代理指示将用于发送和接收数据的候选对的过程。本规范中定义的提名过程在RFC 5245中称为"定期提名"。RFC 5245中称为"积极提名"的提名过程已在此规范中被弃用。
* 被提名、提名标志：候选对提名成功后，则候选对被提名，其提名标志值设置为true。
* 选定对，选定候选对：用于发送和接收数据流组件数据的候选对称为"选定对"。在为数据流生成选定对之前，与数据流组件关联的任何有效对都可用于发送和接收组件数据。数据流的每个组件都有选定对后，选定对将成为数据流的选定对。与选定对关联的候选项称为"选定候选项"。
* 使用协议，ICE使用：将ICE用于NAT穿透的协议。使用协议定义了有关此处定义的过程如何应用于该协议的详细信息。
* 计时器Ta：用于生成新STUN或TURN事务的计时器。
* 计时器RTO（重传超时）：给定STUN或TURN事务的重传计时器。

# 5 ICE候选项采集和交换

作为ICE处理的一部分，启动和响应代理采集候选项、确定优先级和消除冗余候选项，并与使用协议定义的对等节点交换候选项信息（ICE使用）。候选项编码机制和候选项信息交换语义的具体内容不在此规范范围内。

## 5.1 完整实现的过程

### 5.1.1 采集候选项

ICE代理认为通信迫在眉睫时，会采集候选项。启动代理可以基于用户界面提示或启动会话的显式请求执行此操作。每个候选项都包含一个传输地址，并具有类型和Base。此规范定义和采集四种类型：主机候选项、服务器反射候选项、对等反射候选项和中继候选项。服务器反射候选项使用STUN或TURN采集，中继候选项通过TURN获得。由于连接性检查，在ICE的后期阶段获得对等反射候选项。

响应代理采集候选项的过程与启动代理相同。建议响应代理在收到候选项信息后立即启动此过程，以便提醒用户与ICE会话关联的应用程序。

#### 5.1.1.1 主机候选项

主机候选项通过绑定连接到主机上的接口（物理或虚拟，包括VPN接口）的IP地址上的端口来获取。

对于ICE代理希望使用的每个数据流的每个组件，代理应在主机具有的每个IP地址上获取候选项，但下面列出的除外。代理通过绑定到特定IP地址上的UDP端口来获取每个候选项。主机候选项（实际上每个候选项）始终与作为其候选项的特定组件相关联。

每个组件都有一个分配给它的ID，称为"组件ID"。对于RTP/RTCP数据流，除非RTP和RTCP在同一UDP端口（RTP/RTCP复用）中复用，否则RTP本身的组件ID为1，RTCP的组件ID为2。 在RTP/RTCP复用的情况下，RTP和RTCP都使用1的组件ID。

获得候选项时，除非代理确定将使用RTP/RTCP复用（即代理知道、支持并愿意使用RTP/RTCP复用），或者除非代理仅支持RTP/RTCP复用，否则代理必须为RTCP获取单独的候选项。如果代理获得了RTCP的候选项，并且最终使用RTP/RTCP复用，代理不需要对RTCP候选项执行连接检查。缺少组件ID 2并不意味着使用RTCP/RTP复用，因为它还可能意味着不使用RTCP。

如果代理对RTP和RTCP使用单独的候选项，而且代理具有K个IP地址，则最终将具有2\*K主机候选项。

请注意，响应代理在获取其候选项时，通常会知道另一个代理是否支持RTP/RTCP复用，在这种情况下不需要为RTCP获取单独的候选项。但是，缺少组件ID 2并不意味着使用RTCP/RTP复用，因为它还可能意味着不使用RTCP。

建议不要使用多个组件（除了RTP/RTCP流），因为它会增加ICE处理的复杂性。如果需要多个组件，组件ID应从1开始，每个组件增加1。

每个主机候选项的Base设置为候选项本身。

需要从所有IP地址采集主机候选项，但以下情况除外：

* 回环接口中的地址不得包含在候选地址中。
* 弃用IPv4兼容的IPv6地址[RFC4291]和IPv6站点本地单播地址[RFC3879]不得包含在地址候选项中。
* 除非使用ICE的应用程序不支持IPv4（即仅IPv6应用程序[RFC4038]），否则不应将IPv4映射的IPv6地址包含在地址候选项中。
* 如果采集一个或多个与使用阻止位置跟踪的机制生成的IPv6地址对应的主机候选项，则与允许位置跟踪的IPv6地址对应的主机候选项配置在同一接口上，并且不得采集同一网络前缀的一部分。同样，当使用阻止位置跟踪的机制生成的IPv6地址对应的主机候选项时，不得采集与IPv6链路本地地址[RFC4291]对应的主机候选项。

IPv6默认地址选择规范[RFC6724]指定临时地址[RFC4941]优先于永久地址。

#### 5.1.1.2 服务器反射和中继候选项

ICE代理应采集服务器反射和中继候选项。但是，在某些网络中，使用STUN和TURN服务器可能没有必要，并且使用TURN服务器可能非常昂贵，因此某些部署可能会选择不使用它们。如果代理未采集服务器反射或中继候选项，建议通过配置实现该功能并禁用该功能，以便在将来条件发生变化时通过配置重新启用该功能。

代理将每个主机候选项与STUN或TURN服务器配对，这些服务器已配置或通过某种方式将其与这些服务器进行配置或发现。建议配置域名，使用[RFC5389]中的DNS过程（使用SRV记录与"stun"服务一起）来发现STUN服务器，使用[RFC5766]中的DNS过程（使用带有"turn"服务的SRV记录）来发现TURN服务器。

当多个STUN或TURN服务器可用（或者通过DNS记录得知并返回多个结果时），代理可能会采集所有服务器的候选项，并应采集其中至少一个服务器（一个STUN服务器和一个TURN服务器）的候选项。代理通过将主机候选项与STUN或TURN服务器配对实现此目标，对于每个对，代理从主机候选项向服务器发送绑定或分配请求。对STUN服务器的绑定请求未经身份验证，并且忽略响应中的任何ALTERNATE-SERVER属性。代理必须支持[RFC5389]中定义的绑定请求的向后兼容性模式。分配请求应使用客户端通过其他方式获得的长期凭据进行身份验证。

采集过程使用计时器Ta进行控制。每次Ta过期，代理可以生成另一个新的STUN或TURN事务。此事务可以是以前因可恢复错误（如身份验证失败）失败而失败的事务的重试，也可以是新主机候选项和STUN或TURN服务器对的事务。代理不应比每个Ta过期更频繁地生成事务。有关如何设置Ta和STUN重新传输计时器RTO的指导，请参阅第14节。

代理将收到绑定或分配响应。成功的分配响应将为代理提供服务器反射候选项（从映射的地址获取）和XOR-RELAYED-ADDRESS属性中的中继候选项。如果分配请求由于服务器缺少资源来完成它而被拒绝，则代理应发送绑定请求以获取服务器反射候选项。绑定响应将仅为代理提供服务器反射候选项（也从映射的地址获得）。服务器反射候选项的Base是发送分配或绑定请求的主机候选项。中继候选项的Base是该候选项本身。如果中继候选项与主机候选项相同（这种情况在极少数情况下可能发生），则必须丢弃中继候选项。

如果仅IPv6的代理位于使用NAT64[RFC6146]和DNS64[RFC6147]技术的网络中，它还可以从仅IPv4 STUN或TURN服务器采集IPv4服务器反射和/或中继候选项。仅IPv6代理还应利用IPv6前缀发现[RFC7050]来发现NAT64使用的IPv6前缀（如果有），并相应地为每个仅IPv6接口生成服务器反射候选项。NAT64服务器反射候选项的优先级与IPv4服务器反射候选项类似。

#### 5.1.1.3 计算基础

ICE代理为每个候选项分配一个基础。当以下所有情况都为真时，两个候选项具有相同的基础：

* 它们具有相同的类型（主机、中继、服务器反射或对等反射）。
* 其Base具有相同的IP地址（端口可能不同）。
* 对于反射式和中继候选服务器，用于获取它们的STUN或TURN服务器具有相同的IP地址（代理用于联系STUN或TURN服务器的IP地址）。
* 它们是使用相同的传输协议（TCP、UDP）获得的。

同样，如果两个候选服务器的类型不同，其Base具有不同的IP地址，则用于获取它们的STUN或TURN服务器具有不同的IP地址（代理用于联系STUN或TURN服务器的IP地址），或者其传输协议不同。

#### 5.1.1.4 候选项保活

分配服务器反射和中继候选项后，必须保持活动状态，直到ICE处理完成，如第8.3节所述。对于通过绑定请求了解的服务器反射候选项，绑定必须通过对服务器的其他绑定请求保持活动状态。分配的刷新使用刷新事务完成，如[RFC5766]中所述。刷新请求还将刷新服务器反射候选项。

主机候选项不会超时，但候选项地址可能会由于多种原因更改或消失。ICE代理应监视其使用的接口，使Base已经消失的候选项无效，并在出现新的IP地址（在新或当前使用的接口上）时酌情获取新的候选项。

### 5.1.2 确定候选项的优先级

优先级确定过程为每个候选项分配优先级。数据流的每个候选项必须具有唯一的优先级，该优先级必须是介于1和之间的正整数。ICE将使用此优先级来确定连接检查的顺序和候选项的相对偏好。较大值比较小值的优先级更高。

ICE代理应使用第5.1.2.1节中的公式计算此优先级，并使用第5.1.2.2节中的准则选择其参数。如果代理选择使用不同的公式，ICE可能需要更长时间才能收敛，因为代理在检查中不会协调。

在启动和响应代理中，确定候选项优先级的过程很常见。

#### 5.1.2.1 推荐公式

推荐公式结合了候选项的类型偏好（服务器反射式、对等反射式、中继式和主机）、获取候选项的IP地址的位置偏好以及组件ID，使用以下公式计算：

Priority = \*(类型偏好) + \*(位置偏好) + \*(256 - 组件ID)

类型偏好必须为从0（最低，包括）到126（最高，包括）的整数，对于同类型的所有候选项必须相同，并且不同类型的候选项必须不同。对等反射候选项的类型偏好必须高于服务器反射候选项的类型偏好。将值设置为0表示此类型的候选项仅用作最后手段。请注意，根据第5.1.1节的程序采集的候选项永远不会是对等反射的候选项；对等反射候选项是从ICE执行的连接检查中学习得到的。

位置偏好必须为整数，从0（最低，包括）到65535（最高，包括）。当只有一个IP地址时，此值应设置为65535。如果特定数据流的特定组件有多个具有相同类型的候选项，则位置偏好必须每个组件唯一。如果ICE代理是双栈，则应根据[RFC8421]中描述的当前最佳做法设置位置偏好。

组件ID必须是1到256之间的整数（包括）。

#### 5.1.2.2 选择类型和位置偏好的指导

类型偏好的推荐值126表示主机候选项，110表示对等反射候选项，100表示服务器反射候选项，0表示中继候选项。

如果ICE代理是多宿主的，并且具有多个IP地址，则应遵循[RFC8421]中的建议。如果使用多个TURN服务器，则从TURN服务器获得的候选项的位置偏好选择与多宿主本地候选项相似：位置偏好的值用于指示不同服务器之间的偏好，但每个服务器的偏好必须是唯一的。

选择类型偏好时，代理可能会考虑延迟、数据包丢失、成本、网络拓扑、安全性、隐私等因素。

### 5.1.3 消除冗余候选项

接下来，ICE代理（启动和响应）将消除冗余候选项。如果两个候选项具有相同的传输地址，但Base不同，则这些地址不会被视为冗余的。通常，当代理不在NAT后面时，那么服务器反射候选项和主机候选项是冗余的。当且仅当候选项的传输地址和Base与另一个候选项相同时，候选项才是多余的。代理应消除优先级较低的冗余候选项。

## 5.2 lite实现的过程

lite实现仅利用主机候选项。对于每个IP地址，不管其IP地址类型，必须有零或一个候选项。对于lite实现，ICE不能在候选项之间动态选择。因此，不推荐特定IP地址类型包含多个候选项，因为只有连接检查才能真正确定是使用一个地址还是另一个地址。相反，建议具有多个公共IP地址的代理运行完整的ICE实现，以确保其地址的最佳使用。

每个组件都被分配一个ID，称为"组件ID"。对于RTP/RTCP数据流，除非RTCP在同一端口中与RTP复用，否则RTP本身的组件ID为1，RTCP的组件ID为2。如果代理使用RTCP而不进行复用，则必须获取其候选项。但是，缺少组件ID 2并不意味着使用RTCP/RTP复用，因为它还可能意味着不使用RTCP。

为每个候选项分配一个基础。对于从不同IP地址分配的两个候选项，基础必须不同；否则必须相同。对每个IP地址赋予一个简单的整数增量即可。此外，必须为同一数据流的所有候选项分配相同的优先级。如果用第5.1.2.1节中的公式计算优先级，则应将类型偏好设置为126。如果主机仅为IPv4，则应将位置偏好设置为65535。如果主机是IPv6或双栈，则应将位置偏好设置为RFC 6724中描述的IP地址的优先级值。

接下来，代理为每个数据流的每个组件选择默认候选项。如果主机仅是IPv4，则每个数据流的每个组件只有一个候选组件；因此该候选项是默认值。如果主机仅为IPv6，则默认候选项通常是全局范围的IPv6地址。无论IPv4还是IPv6用于默认候选项，双栈主机都应允许配置，并且配置需要基于其管理员认为在当前网络环境中成功的可能性更高。

本节中的过程在启动和响应代理中很常见。

## 5.3 交换候选项信息

ICE代理（启动和响应）需要交换以下有关候选项的信息。每个ICE使用都必须定义如何交换使用协议的信息。本节介绍需要交换的信息。

* 候选项：一个或多个候选项。对于每个候选项：
  + 地址：候选项的IP地址和传输协议端口。
  + 传输：候选项的传输协议。如果使用协议仅通过单个传输协议运行，则可以省略此内容。
  + 基础：最多32个字符的序列。
  + 组件ID：候选项的组件ID。如果使用协议不使用组件的概念，则可以省略此内容。
  + 优先级：候选项的32位优先级。
  + 类型：候选项的类型。
  + 相关地址和端口：候选项的相关IP地址和端口。如果代理不想显示这些值，例如出于隐私原因，可以省略这些值或设置为无效值。
  + 扩展参数：使用协议可以定义将来添加新的每个候选项ICE参数的方法。
* lite或完整：代理是lite代理还是完整代理。
* 连接检查步进值：代理希望使用的连接检查的步进值。如果代理希望使用定义的默认值，则可以省略此值。
* 用户名片段和口令：用于执行连接检查的值。该值必须是不可猜测的，至少128位随机数生成器输出用于生成口令，并且至少24位输出用于生成用户名片段。
* 扩展：新的媒体流或会话级属性（ICE可选项）。

如果使用协议易受ICE不匹配（第5.4节）的影响，并且能够检测到，则检测代理需要一种方法将此信息传达给其对等节点。这是一个布尔标识。

使用协议可能需要（也可能不需要）处理与不支持ICE的旧实现的向后兼容性。如果支持并使用对非ICE的回退机制，则假定使用协议提供了一种除ICE参数之外传达默认候选项（其IP地址和端口）的方法。

一旦代理发送了候选项信息，必须准备好接收每个候选项上的STUN和数据包。如第12.1节所述，数据包在数据的默认目标出现之前，可以发送给候选项。

## 5.4 ICE不匹配

某些中间盒（如ALG）可能以破坏ICE的方式（例如，通过在SDP中重写IP地址）来更改信令信息。这称为"ICE不匹配"。如果使用协议容易受到ICE不匹配的影响，则响应代理需要能够检测到它，并将ICE不匹配通知对等ICE代理。

每个使用协议都需要定义使用协议是否容易受到ICE不匹配的影响、如何检测ICE不匹配以及检测到ICE不匹配时是否需要采取特定操作。

# 6 ICE候选项处理

一旦ICE代理采集了候选项并与对等节点交换候选项（第5节），它将决定自己的角色。此外，完整实现将形成清单，并开始与对等节点执行连接检查。

## 6.1 完整实现的过程

### 6.1.1 确定角色

对于每个会话，每个ICE代理（启动和响应）都扮演一个角色。有两种角色：控制代理和受控代理。控制代理负责选择用于通信的最终候选对。以下各节详细介绍控制和受控代理遵循的实际程序。

确定角色及其对行为影响的规则如下：

* 两个代理都为完整代理：启动ICE处理的启动代理必须担任控制角色，另一个代理必须担任受控角色。两个代理将组成清单、运行ICE状态机并生成连接检查。控制代理将执行第8.1节中的逻辑，提名将成为（如果与提名关联的连接检查成功）选定对，然后两个代理都终止ICE，如第8.1.2节中所述。
* 一个代理为完整代理，一个为lite代理：完整代理必须担任控制角色，lite代理必须担任受控角色。完整代理将组成清单、运行ICE状态机并生成连接检查。该代理将执行第8.1节中的逻辑，提名将成为（如果与提名关联的连接检查成功）选定对，并使用第8.1.2节中的逻辑结束ICE。lite代理将只侦听、接收和响应连接检查，然后完成第8.2节中所述的ICE。对于lite实现，每个数据流的ICE处理状态被视为正在运行，ICE的总体状态为"正在运行"。
* 两个lite代理：启动ICE处理的启动代理必须担任控制角色，另一个必须担任受控角色。在这种情况下，不会发送任何连接检查。相反，一旦交换了候选项，每个代理执行第8节中描述的处理，而不进行连接检查。两个代理有可能都认为它们是受控的或控制的。在后一种情况下，通过启动后选项交换的信令协议的眩光检测功能解决冲突。每个数据流的ICE处理状态被视为正在运行，ICE的总体状态为"正在运行"。

确定会话的角色后，这些角色将在会话的整个生命周期中保留。角色可以作为ICE重新启动的一部分（第9节）重新确定，然而ICE代理不得作为ICE重新启动的一部分重新确定角色，除非满足以下一个或多个条件：

* 完整代理变为lite代理：如果控制代理本来是完整的，但是切换到lite，则必须重新确定角色（如果对等代理也是完整的）。
* 角色冲突：如果ICE重新启动导致角色冲突，则角色可能会由于第7.3.1.1 节中的角色冲突过程而重新确定。

注：对于某些第三方呼叫控制（3PCC）[RFC3725]方案，ICE重新启动可能会导致角色冲突。

注：在确定角色之前，代理需要相互通知它们是完整的还是lite。其机制特定于信令协议，不在文档的范围之内。

如果对等节点发起重新确定角色，即使不符合这样做的条件，代理也必须接受。如果对等节点符合RFC 5245，则可能发生这种情况。

### 6.1.2 形成清单

每个数据流有一个清单。要形成清单，启动和响应ICE代理形成候选对、计算对优先级、按优先级排序对、修剪对、删除低优先级对以及设置清单状态。如果候选项被添加到清单（例如，由于检测到对等反射候选项），则代理将为更新的清单重新执行这些步骤。

#### 6.1.2.1 清单状态

每个清单都有一个状态，该状态捕获与清单关联的数据流的ICE检查状态。 状态为：

* 正在运行：清单既未完成，也未失败。清单最初设置为"正在运行"状态。
* 已完成：该清单包含数据流中每个组件的提名对。
* 失败：清单对于数据流的每个组件都没有有效的对，并且清单中的所有候选对都处于"失败"或"成功"状态。换句话说，清单中至少一个组件的候选对都处于"失败"状态，意味着该组件已失败，同时意味着清单已失败。

#### 6.1.2.2 形成候选对

ICE代理将每个本地候选项与具有相同IP地址类型的同一数据流的同一组件的每个远程候选项配对。一些本地候选项可能无法与远程候选项配对，并且一些远程候选项不会与本地候选项配对。如果一个代理不包括数据流中所有组件的候选项，则可能发生此情况。如果发生这种情况，该数据流的组件数将有效减少，并且被视为等于每个代理在数据流的所有组件之间提供的最大组件ID的最小值。

对于RTP，当一个代理提供RTCP的候选项，而另一个代理不提供候选项时，就会发生这种情况。另一个示例是，启动代理可以在同一端口[RFC5761]上复用RTP和RTCP。但是，由于启动代理不知道对等代理是否可以执行此类复用，因此它包括单独端口上的RTP和RTCP候选项。如果对等代理可以执行此类复用，则每个候选项仅包含单个组件：对于组合的RTP/RTCP复用。ICE最终将充当此候选项的单个组件。

对于IPv6，通常主机的每个接口具有多个主机候选项。为了保持结果候选对的数量合理，并避免极不可能起作用的候选对，IPv6链接本地地址不得与链接本地地址以外的地址配对。

本地和远程候选项都是特定组件的默认候选项的候选对称为该组件的"默认候选对"。如果两个代理都未进行ICE感知，那么这就是用于传输数据的对。

图5显示了传输地址、候选项、候选对和清单的属性和关系。



图 5：清单的示意图

#### 6.1.2.3 计算对优先级和排序对

ICE代理为每个候选对计算优先级。令G为控制代理提供的候选项的优先级。D为受控代理提供的候选项的优先级。则对的优先级计算如下：

pair priority = + +

代理按候选对优先级的递减顺序对每个清单进行排序。如果两个对具有相同的优先级，则它们之间的顺序任意。

#### 6.1.2.4 修剪对

已排序的候选对列表用于确定将执行的连接检查序列。每个检查都涉及将来自本地候选项的请求发送给远程候选项。由于ICE代理不能直接从反射候选项（服务器反射或对等反射）发送请求，只能从其Base发送请求，因此代理接下来将遍历已排序的候选对列表。对于本地候选项具有反射性的每对，必须用其Base替换候选项。

代理修剪每个清单。如果候选对与同一清单中的优先级较高的候选对冗余，则删除该候选对。如果本地候选项的本地Base相同且远程候选项相同，则两个候选对是冗余的。结果是排序的候选对序列，称为该数据流的"清单"。

#### 6.1.2.5 删除优先级较低的对

为了限制第19.5.1节中所述的攻击，ICE代理必须限制代理在清单集中执行的连接检查总数。这是通过限制清单集中的候选对总数来实现的。清单集的候选对的默认限制为100，但该值必须可配置。限制在每个清单内强制实施，丢弃优先级较低的候选对，使清单集中的候选对总数小于限制值。丢弃应均匀进行，以便每个清单中的候选对减少相同的数量。

建议尽可能选择比默认值更低的限制值，并将该值设置为在实际部署配置中可能创建的最大可能候选对数。可配置需求旨在提供一种手段，以便在部署后发现存在问题时能现场修复此值。

#### 6.1.2.6 计算候选对状态

清单中的每个候选对都有一个基础（对中本地和远程候选项基础的组合）和以下状态之一：

* 等待：尚未为此对发送检查，但该对未冻结。
* 正在进行中：已为此对发送检查，但事务正在进行中。
* 成功：已发送此对的检查，并生成成功的结果。
* 失败：已为此对发送检查，但失败（从未收到对检查的响应或收到失败响应）。
* 已冻结：尚未发送此对的检查，并且在取消冻结并移动到"等待"状态之前无法发送该检查。

对在状态之间移动，如图6所示。



图 6：配对状态有限状态机（FSM）

通过执行以下步骤计算清单中每对的初始状态：

1. 清单被放置在一个有序的列表中（顺序由每个ICE使用情况决定），称为"清单集"。

2. ICE代理最初将所有候选对置于冻结状态。

3. 代理将清单中的所有清单设置为"运行"状态。

4. 对于每个基础，代理将一个候选对的状态设置为"等待"状态（解冻）。通过在第一个清单（按照使用情况定义的清单集顺序）中查找第一对候选对（按最低组件ID排序，然后在组件ID相等时按最高优先级）选择解冻的候选对。

注：上述过程与RFC 5245不同，RFC 5245最初仅将第一个清单中的候选对置于"等待"状态。现在，它适用于具有该基础的第一个清单中的候选对，即使清单不是清单集中的第一个。

下表说明了一个示例。

表图例：每行（m1，m2,...）表示与数据流关联的清单。m1表示清单集中的第一个清单。每列（f1、f2,...）表示基础。给定列中的每个候选对共享相同的基础。f-cp 表示处于"冻结"状态的候选对。w-cp 表示处于"等待"状态的候选对。

1. 代理将清单中的所有对设置为"冻结"状态。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | f1 | f2 | f3 | f4 | f5 |
| m1 | f-cp | f-cp | f-cp |  |  |
| m2 | f-cp | f-cp | f-cp | f-cp |  |
| m3 | f-cp |  |  |  | f-cp |

2. 对于每个基础，具有最低组件ID的候选对将置于"等待"状态，除非与同一基础关联的候选对已置于检查清单集中的其他清单之一的"等待"状态。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | f1 | f2 | f3 | f4 | f5 |
| m1 | w-cp | w-cp | w-cp |  |  |
| m2 | f-cp | f-cp | f-cp | w-cp |  |
| m3 | f-cp |  |  |  | w-cp |

表 1：配对状态示例

在第一个清单（m1）中，每个基础的候选对置于"等待"状态，因为尚未将相同基础的对置于"等待"状态。

在第二个清单（m2）中，基础f4的候选对置于"等待"状态。 基础f1、f2和f3的候选对保留在冻结状态，因为这些基础的候选对已置于等待状态（在清单m1中）。

在第三个清单（m3）中，基础f5的候选对置于"等待"状态。基础f1的候选对保留在冻结状态，因为该基础的候选对已置于"等待"状态（在清单m1中）。

处理每个清单后，清单集中每个基础的一对候选项已置于"等待"状态。

### 6.1.3 ICE状态

ICE代理的状态由清单的状态确定。如果所有清单都已完成，则状态为"完成"，如果所有清单都失败，则状态为"失败"，或者“运行”。

### 6.1.4 检查安排

#### 6.1.4.1 触发检查队列

一旦ICE代理计算了清单并创建了清单集（如第6.1.2节中所述），代理将开始执行连接检查（普通和触发）。 对于触发的连接检查，代理为每个清单维护FIFO队列，称为"触发检查队列"，其中包含候选对，将在下一个可用机会发送检查。触发检查队列最初为空。

#### 6.1.4.2 执行连接检查

普通和触发连接检查的生成由计时器Ta控制。一旦设置了清单集中的候选对的初始状态，按照第7节中的程序，在"运行"状态中的第一个清单内对候选对执行检查。之后，每当Ta触发清单集中的"运行"状态下一个清单时，都会选取该清单中的候选项进行检查。处理清单集中的"运行"状态下的最后一个清单后，将再次选取第一个清单，等等。

每当Ta触发时，ICE代理将在检查清单中通过执行以下步骤选取的候选对执行检查：

1. 如果与清单关联的触发检查队列包含一个或多个候选对，则代理将从队列中删除顶部对，对该对执行连接检查，将候选对状态置于"正在进行"状态，并中止后续步骤。

2. 如果"等待"状态中没有候选对，并且"冻结"状态中有一个或多个对，则代理将检查与处于"冻结"状态的每个对关联的基础。对于给定的基础，如果"等待"或"正在"状态中没有对（在清单集中的任何清单中），则代理将候选对状态设置为"等待"，并继续执行下一步。

3. 如果"等待"状态中有一个或多个候选对，则代理将选择优先级最高的候选对（如果有多个具有相同优先级的对，选择具有最低组件ID的对），对执行连接检查，将候选对状态置于"正在进行"状态，并中止后续步骤。

4. 如果达到此步骤，则无法对选取的清单进行检查。因此，无需等待计时器Ta再次过期，请选择处于"运行"状态的下一个清单，然后返回到步骤#1。如果"正在运行"状态中的每个清单都发生这种情况，这意味着没有执行连接检查的剩余候选对，则中止这些步骤。

代理选择要执行连接检查的候选对后，代理将启动检查，并从与配对的本地候选项关联的Base中将绑定请求发送到对的远程候选项，如第7.2.4节所述。

根据本地策略，代理可以选择终止执行随时设置的清单中一个或多个清单的连接检查。但是，只允许控制代理结束ICE（第8节）。

要计算检查的消息完整性，代理使用从其对等节点获取的候选项信息中获取的远程用户名片段和口令。代理自己的候选项的本地用户名片段直接获得。

## 6.2 lite实现的过程

lite实现跳过第6节中的大部分步骤，除非验证对等节点的ICE支持并确定其在ICE处理中的角色。

如果lite实现是控制代理（仅当对等ICE代理也是lite实现时才会发生），它将根据候选项交换中的候选对（对于IPv4，只有一对）选择候选对，然后使用反映该选择的新候选项信息更新对等节点（如果需要，但IPv4主机从不需要）。

# 7 执行连接检查

本节介绍如何执行连接检查。

ICE代理必须符合[RFC5389]。完整实现既充当STUN客户端，又充当STUN服务器，而lite实现仅充当STUN服务器（因为它不生成连接检查）。

## 7.1 STUN扩展

ICE扩展STUN的属性：优先级、使用-候选项、ICE控制和ICE受控。这些属性在第16.1节中正式定义。本节介绍属性的用法。

这些属性仅用于ICE连接检查。

### 7.1.1 优先级

优先级属性必须包含在绑定请求中，并设置为第5.1.2节中算法为本地候选项计算的值，但具有对等反射候选项的类型偏好。

### 7.1.2 使用-候选项

控制代理必须包括使用-候选项属性才能提名候选对（第8.1.1节）。受控代理不得在绑定请求中包含使用-候选项属性。

### 7.1.3 ICE-控制与ICE-受控

控制代理必须在绑定请求中包括ICE-控制属性。受控代理必须在绑定请求中包括ICE-受控属性。

当ICE角色冲突发生时，任一属性的内容都用作绑定决胜属性值（第7.3.1.1节）。

## 7.2 STUN客户程序

### 7.2.1 为中继候选项创建权限

如果连接检查是使用中继的本地候选项发送的，则客户端必须首先创建权限（如果以前尚未创建）。如果它告诉TURN服务器为给定的中继候选项创建指向远程候选项的IP地址的权限，它之前就会创建一个权限。要创建权限，ICE代理遵循[RFC5766]中定义的过程。必须针对远程候选项的IP地址创建权限。建议代理将TURN通道的创建推迟到ICE完成，在这种情况下，通常使用CreatePermission请求创建连接检查的权限。一旦建立，代理必须保持权限激活，直到ICE结束。

### 7.2.2 形成凭据

连接检查绑定请求必须使用STUN短期凭据机制。

凭据的用户名是通过将对等节点的用户名片段与发送请求的ICE代理的用户名片段相关联（用冒号"："分隔）而形成的。

口令则是对等节点提供的。

例如，假设ICE代理L是启动代理，ICE代理R是响应代理。代理L候选项包括其用户名片段LFRAG和口令LPASS。代理R的用户名片段RFRAG和口令RPASS。从L到R的连接检查使用用户名RFRAG：LFRAG和口令RPASS。从R到L的连接检查使用用户名LFRAG：RFRAG和口令LPASS。响应使用与请求相同的用户名和口令（请注意，响应中不存在USERNAME属性）。

### 7.2.3 差异服务处理

如果代理在发送的数据包中使用差异化服务代码点（DSCP）标记[RFC2475]，则代理应对其发送的绑定请求和响应应用相同的标记。

如果在数据包上使用了多个DSCP标记，代理应选择其中一个标记用于连接检查。

### 7.2.4 发送请求

通过从与本地候选项关联的Base发送绑定请求到远程候选项，生成连接检查。[RFC5389]描述了如何构造和生成绑定请求。

执行连接检查时，不得假定支持向后兼容RFC 3489。必须使用FINGERPRINT机制进行连接检查。

### 7.2.5 处理响应

本节定义用于处理特定ICE连接检查的绑定响应的其他过程。

收到绑定响应时，它与使用事务ID[RFC5389]的相应绑定请求相关，后者随后将响应与发送绑定请求的候选对关联。之后，根据角色冲突、失败或成功的程序，根据以下程序处理响应。

#### 7.2.5.1 角色冲突

如果绑定请求产生487（角色冲突）错误响应（第7.3.1.1节），并且ICE代理在请求中包含ICE-受控属性，则代理必须切换到控制角色。如果代理在请求中包含ICE-控制属性，则代理必须切换到受控角色。

代理更换其角色后，代理必须将其检查生成的487错误响应的候选对添加到与该对所属的清单关联的触发检查队列，并将候选对状态设置为"等待"。稍后执行触发连接检查时，绑定请求的ICE-控制/ICE-受控属性将指示代理的新角色。代理必须更改决胜属性的值。

注：角色切换要求代理重新计算对优先级（第6.1.2.3节），因为优先级值取决于角色。

注：角色切换还将影响代理是否负责提名候选对，以及一旦ICE完成，代理是否负责与对等节点启动候选项交换进行信息更新。

#### 7.2.5.2 失败

本节介绍候选对状态设置为"失败"的情况。

注：当ICE代理由于连接检查错误而将候选对状态设置为"失败"时，代理不会更改具有相同**基础**的其他候选对的状态。

##### 7.2.5.2.1 非对称传输地址

ICE代理必须检查绑定请求和响应中的源和目标传输地址是对称的。也就是说，响应的源IP地址和端口必须等于向其发送绑定请求的目标IP地址和端口，响应的目标IP地址和端口必须等于发送绑定请求的源IP地址和端口。如果地址不是对称的，则代理必须将候选对状态设置为"失败"。

##### 7.2.5.2.2 ICMP错误

ICE代理可能支持处理ICMP错误以进行连接检查。如果代理支持处理ICMP错误，并且绑定请求生成硬ICMP错误，则代理应将候选对的状态设置为"失败"。 实施者在决定如何处理和是否处理ICMP错误时需要注意，ICMP错误可被用作拒绝服务（DoS）攻击的方法。

##### 7.2.5.2.3 超时

如果绑定请求事务超时，ICE代理必须将候选对状态设置为"失败"。

##### 7.2.5.2.4 不可恢复的STUN响应

如果绑定请求生成的STUN错误响应不可恢复[RFC5389]，ICE代理应将候选对状态设置为"失败"。

#### 7.2.5.3 成功

如果以下每个条件都为真，则视为连接检查成功：

* 绑定请求生成成功响应；
* 绑定请求和响应中的源和目标传输地址是对称的。

如果检查成功，ICE代理将执行以下描述的操作（按顺序排列）。

##### 7.2.5.3.1 发现对等反射候选项

ICE代理必须从STUN响应中检查映射的地址。如果传输地址与代理知道的任何本地候选项不匹配，则映射的地址表示一个新候选项：对等反射候选项。与其他候选项一样，对等反射候选项具有类型、Base、优先级和基础。它们的计算方式如下：

* 该类型是对等反射。
* Base是发送绑定请求的候选对的本地候选项。
* 优先级是绑定请求中的优先级属性的值。
* 第5.1.1.3节描述了基础。

然后，对等反射候选项将添加到数据流的本地候选项列表中。用户名片段和口令与该数据流的所有其他本地候选项相同。

ICE代理不需要将对等反射候选项与远程候选项配对，因为将按照第7.2.5.3.2节中的程序创建有效的对。如果代理希望将对等反射候选项与将生成的有效对中远程候选项以外的远程候选项配对，则代理可能会向包含对等反射候选项的对等节点提供更新的候选信息。这将导致对等反射候选项与所有其他远程候选项配对。

##### 7.2.5.3.2 构建有效对

ICE代理构造一个候选对，其本地候选项等于响应的映射地址，其远程候选项等于请求发送到的目标地址。这称为"有效对"。

有效对可能等于生成连接检查的对、清单中的不同对或当前不在清单中的对。

代理维护一个单独的列表，称为"有效列表"。清单集中的每个清单都有一个有效列表。有效列表包含有效对。每个有效列表初始为空。

有效列表中的每个有效对都有一个标志，称为"提名标志"。将有效对添加到有效列表时，提名标志值设置为"false"。

有效对将添加到有效列表中，如下所示：

1. 如果有效对等于生成检查的对，则该对将添加到与该对所属的清单关联的有效列表中；或者

2. 如果有效对等于清单中的另一对，则该对将添加到与该对的清单关联的有效列表中。生成检查的对不会添加到有效列表中；或者

3. 如果有效对不在任何清单中，则代理使用第6.1.2节中的算法根据每个候选项的优先级计算对的优先级。本地候选项的优先级取决于其类型。除非类型是对等反射的，否则优先级等于在候选项交换中该候选项发出的优先级。如果类型是对等反射的，则它等于代理在刚刚完成的绑定请求中设定的优先级属性。远程候选项的优先级取自对等节点的候选项信息。如果候选项未显示，则该检查已触发对新远程候选项的检查。在这种情况下，优先级被视为绑定请求中优先级属性的值，该请求触发了刚刚完成的检查。然后，该对将添加到有效列表中。

注：有效对不会出现在任何清单中是非常常见的。回想一下，清单有一对本地候选项从不反射；这些对把他们的本地候选项转换为反射候选项的Base，如果他们是冗余的，将被裁减掉。当绑定请求的响应到达时，如果两者之间有NAT，则映射的地址将是反射的。在这种情况下，有效对将具有与清单中的任何对不相同的本地候选项。

##### 7.2.5.3.3 更新候选对状态

ICE代理将生成检查的候选对和构造的有效对（可能不同）的状态设置为"成功"。

代理必须在所有清单中具有相同基础的所有其他冻结候选对设置为"等待"状态。

注：在给定的清单中，具有相同基础的候选对通常具有不同的组件ID。

##### 7.2.5.3.4 更新提名标志

如果控制代理发送带有使用-候选项属性集的绑定请求，并且ICE代理成功收到请求所对应的响应，则代理将该对的提名标志设置为true。如果请求失败（第7.2.5.2节），代理必须从有效列表中删除候选对，将候选对状态设置为"失败"，并将清单状态设置为"失败"。

如果受控代理成功收到其发送的绑定请求所对应的响应，并且绑定请求由收到的具有使用-候选项属性集（第7.3.1.4节）的绑定请求触发，则代理将对的提名标志设置为true。如果触发的请求失败，代理必须从有效列表中删除候选对，将候选对状态设置为"失败"，并将清单状态设置为"失败"。

为数据流的组件设置提名标志后，将结束该组件的ICE处理（第8节）。

#### 7.2.5.4 清单状态更新

无论连接检查是成功还是失败，完成检查可能需要更新清单状态。对于清单集中的每个清单，如果所有候选对都处于"失败"或"成功"状态，并且与清单关联的数据流的每个组件的有效列表中都没有有效的对，则清单的状态将设置为"失败"。 如果有效列表中的每个组件都有一个有效的对，则清单的状态将设置为"成功"。

## 7.3 STUN服务器过程

ICE代理（lite版或完整版）必须准备好在其最新的候选项交换中包含的每个候选项的Base上接收绑定请求。

代理必须使用短期凭据机制（即消息-完整性属性）对请求进行身份验证并执行消息完整性检查。同样，必须使用短期凭据机制进行响应。如果用户名由两个值与冒号分隔，则代理必须考虑该用户名有效，其中第一个值等于正在进行会话的代理在候选项交换中生成的用户名片段。启动代理可能会（事实上很可能）在从对等节点接收候选项之前收到绑定请求。如果发生这种情况，代理必须立即生成响应（包括计算映射的地址，如第7.3.1.2节所述）。此时，代理有足够的信息来生成响应；不需要来自对等节点的口令。收到响应后，必须继续执行所需的其余步骤；有关完整实现，请参阅第7.3.1.3、7.3.1.4和7.3.1.5节。在响应之前收到多个STUN请求的情况下，可能会导致触发检查队列中几个对排队。

代理不得使用ALTERNATE-SERVER机制，也不得支持RFC 5389中定义的向后兼容性机制（用于使用RFC 3489中的协议）。它必须利用指纹机制。

如果代理在其数据包中使用DSCP标记[RFC2475]，则应对绑定响应应用相同的标记。这同样适用于终结点可能应用于数据包的任何第2层标记。

### 7.3.1 完整实现的其他过程

本小节定义当完整实现收到绑定请求时，适用于完整实现的其他服务器过程。

#### 7.3.1.1 检测和修复角色冲突

在ICE的某些用法（如3PCC）中，两个ICE代理最终可能选择相同的角色，从而导致角色冲突。本节介绍检测和修复角色冲突的机制。使用文档必须指定是否需要此机制。

代理必须检查ICE-控制属性或ICE-受控属性的绑定请求。必须遵循以下程序：

* 如果代理处于控制角色，并且请求中存在ICE-控制属性：
  + 如果代理的决胜属性值大于或等于ICE-控制属性的内容，则代理将生成绑定错误响应，并包含值为487（角色冲突）但保留其角色的错误代码属性。
  + 如果代理的决胜属性值小于ICE-控制属性的内容，则代理将切换到受控角色。
* 如果代理处于受控角色中，并且请求中存在ICE-受控属性：
  + 如果代理的决胜属性值大于或等于ICE-受控属性的内容，则代理将切换到控制角色。
  + 如果代理的决胜属性值小于ICE-受控属性的内容，则代理将生成绑定错误响应，并包含值为487（角色冲突）但保留其角色的错误代码属性。
* 如果代理处于受控角色，并且请求中存在ICE-控制属性，或者如果代理位于控制角色中，并且请求中存在ICE-受控属性，则不存在冲突。

角色更改需要代理重新计算对优先级（第6.1.2.3节），因为这些优先级是角色的函数。角色的改变还将影响代理是否负责选择选定对，并在ICE结束时启动更新的候选项信息的交流。

如果代理成功响应绑定请求，即使代理更改了角色，则遵循第7.3.1节中的规范。

#### 7.3.1.2 计算映射地址

对于在中继候选服务器上收到的请求，用于STUN处理的源传输地址（即生成XOR-MAPPED-ADDRESS属性）是TURN服务器看到的传输地址。如果绑定请求是通过数据指示消息传递的，则该源传输地址将存在于数据指示消息的XOR-PEER-ADDRESS属性中。如果绑定请求是通过ChannelData消息传递的，则源传输地址是绑定到通道的地址。

#### 7.3.1.3 学习对等反射候选项

如果请求的源传输地址与任何现有远程候选项不匹配，则表示新的对等反射远程候选项。此候选项的结构如下：

* 类型是对等反射。
* 优先级是绑定请求中的优先级属性的值。
* 基础是任意值，不同于所有其他远程候选项的基础。如果任何后续的候选项交换都包含此对等反射候选项，则表示为该候选项的实际基础。
* 组件ID是请求发送到的本地候选项的组件ID。

此候选项将添加到远程候选列表中。但是，ICE代理不会将此候选项与任何本地候选项配对。

#### 7.3.1.4 触发检查

接下来，代理构造一对，本地候选项具有收到的STUN请求的传输地址（如代理所看到的），远程候选项（可能是刚刚学习的对等反射远程候选项）与请求中的源传输地址相等。本地候选项可以是主机候选项（对于未通过中继接收请求的情况），也可以是中继候选项（对于通过中继接收请求的情况）。本地候选项永远不能是服务器反射候选项。由于代理都知道两个候选项，因此可以获取其优先级并计算候选对的优先级。然后，在清单中查到此对。可以有以下几个结果之一：

* 当对已经在清单上时：
  + 如果该对的状态为"成功"，则不执行任何进一步操作。
  + 如果该对的状态为"正在进行"，则代理将取消正在进行的事务。取消意味着代理不会重新传输与连接检查事务关联的绑定请求，不会将尚未响应视为失败，而是等待事务超时的持续时间以获得响应。此外，代理必须将与清单关联的触发清单中的对进行排队，并将对的状态设置为"等待"，以便触发此对的新连接检查。创建新的连接检查可尽快验证正在进行的对，而无需等待与原始连接检查事务关联的绑定请求的重新传输。
  + 如果该对的状态为"等待"、"冻结"或"失败"，则代理必须将与清单关联的触发清单中的对进行排队（如果不存在），并将配对的状态设置为"等待"，以便触发此对的新连接检查。请注意，对的状态从"失败"更改为"等待"也可能触发关联清单的状态更改。

当两个代理都位于NAT之后时，执行这些步骤是为了方便快速完成ICE。

* 如果对尚未在清单上：
  + 根据对的优先级插入到清单中。
  + 其状态设置为"等待"。
  + 将此对进入触发检查队列进行排队。

发送触发检查时，将按照第7.2.4节所述进行构造和处理。这些过程要求代理知道对等节点的传输地址、用户名片段和口令。远程候选项的用户名片段等于刚刚收到的绑定请求中USERNAME的冒号之后的部分。使用该用户名片段，代理可以检查从对等节点接收的候选项（在分叉的情况下可能有多个候选项），并查找此用户名片段。然后选取相应的口令。

#### 7.3.1.5 更新提名标志

如果受控代理收到带有使用-候选项属性集的绑定请求，并且ICE代理接受该请求，则基于第7.3.1.4节中计算的对的状态执行以下操作：

* 如果此对的状态为"成功"，则意味着此对以前发送的检查生成了成功的响应并生成了有效的对（第7.2.5.3.2节）。代理将有效对的提名标志值设置为true。
* 如果收到的绑定请求触发了在触发检查队列（第7.3.1.4节）中排队的新检查，则一旦发送检查并生成成功响应并生成有效对，代理将该对的提名标志设置为true。如果请求失败（第7.2.5.2节），代理必须从有效列表中删除候选对，将候选对状态设置为"失败"，并将清单状态设置为"失败"。

如果受控代理不接受来自控制代理的请求，则受控代理必须通过适当错误代码响应（例如400）[RFC5389]拒绝提名请求。

为数据流的组件设置提名标志后，它将结束该组件的ICE处理。见第8节。

### 7.3.2 lite实现的其他过程

如果受控代理收到带有使用-候选项属性集的绑定请求，并且ICE代理接受该请求，则代理将构造一个候选对，其本地候选项具有接收请求的传输地址，其远程候选项等于收到的请求的源传输地址。此候选对被分配任意优先级，并放入关联的清单的有效列表中。代理将该对的提名标志设置为true。

为数据流的组件设置提名标志后，它将结束该组件的ICE处理。见第8节。

# 8 完成ICE处理

本节介绍ICE代理如何完成ICE。

## 8.1 完整实现的过程

完成ICE涉及由控制代理提名对以及更新状态机制。

### 8.1.1 提名对

在提名之前，控制代理进行连接检查，直到满足某些停止条件。之后，根据评估标准，控制代理在有效提名列表中的有效对中选取一对。

一旦控制代理选择了有效的对进行提名，它将重复生成此有效对的连接检查（通过将生成检查的对在触发检查队列中排队），这一次包含使用-候选项属性（第7.2.5.3.4节）。第7.3.1.5节介绍了受控代理的程序。

最终，如果提名成功，控制代理和受控代理在数据流的每个组件的有效列表中都有一个提名对。一旦ICE代理将清单的状态设置为"已完成"（当数据流的每个组件都有提名对时），该对将成为该代理的选定对，用于发送和接收该数据流中组件的数据。

如果代理无法为数据流的每个组件生成选定对，则代理必须采取适当的操作来通知其他代理，例如删除流。具体的操作方法超出了本规范的范围。

停止连接检查和选择一对进行提名的准则超出了本规范的范围。它们是局部优化的问题。唯一的要求是代理最终必须选择一个候选对，并包含使用-候选项属性集生成该对的检查。

一旦控制代理成功提名候选对（第7.2.5.3.4节），代理不得为ICE会话中数据流的同一组件提名另一对。这样做需要重新启动ICE。

不支持此规范的控制代理（即根据RFC 5245实现）可能会提名多个候选对。在RFC 5245中称为"积极提名"。如果控制代理提名了多个候选对，并且受控代理接受多个提名请求，则代理必须生成选定对并使用具有最高优先级的对。

支持此规范的终结点使用"ice2"ICE选项（第10节），应防止根据RFC 5245实现的控制代理使用积极提名。

注：在RFC 5245中，"积极提名"的使用允许代理在最终选择对之前持续提名对，以便发送这些对上的数据。在此规范中，数据始终可以在任何有效对上发送，而无需提名。因此不再需要积极提名。

### 8.1.2 更新清单和ICE状态

对于控制代理和受控代理，当数据流组件的候选对被提名时，可能会影响与数据流关联的清单中的其他对。还可能影响清单的状态：

* 一旦提名数据流组件的候选对，并且与数据流关联的清单的状态为"正在运行"，ICE代理必须从清单和触发检查队列中删除同一组件的所有候选对。如果对的状态为"正在进行"，则代理将取消正在进行的事务。取消意味着代理不会重新传输与连接检查事务关联的绑定请求，不会将尚未响应视为失败，而是等待事务超时的持续时间以获得响应。
* 一旦提名数据流的每个组件的候选对，并且与数据流关联的清单的状态为"正在运行"，ICE代理将清单的状态设置为"已完成"。
* 一旦提名数据流组件的候选对，代理必须继续响应其可能收到的提名对以及与数据流关联的清单中的任何剩余候选对的任何绑定请求。如第7.3.1.4节所定义，当对的状态成功时，代理在接收对的绑定请求时将不再生成触发检查。

一旦清单集中每个清单的状态都是"完成"，代理将ICE会话的状态设置为"已完成"。

如果清单的状态为"失败"，ICE没有成功完成与清单关联的数据流的过程。正确的行为取决于清单集中清单的状态。如果控制代理希望继续会话而不与"失败"清单关联的数据流，并且在"正在运行"或"已完成"模式下仍有一个或多个清单，则代理可以让ICE处理继续。代理必须采取适当的操作来删除失败的数据流。如果控制代理不想继续会话，并且必须终止会话，则ICE会话的状态设置为"失败"。

如果清单集中每个清单的状态为"失败"，则ICE会话的状态将设置为"失败"。除非控制代理希望在没有数据流的情况下继续会话，否则它必须终止会话。

## 8.2 lite实现的过程

当ICE结束时，lite版ICE代理可以释放ICE未使用的主机候选项，如第8.3节所述。

如果对等节点是完整代理，则一旦lite代理收到候选对的提名请求，lite代理视该对为被提名。一旦数据流的每个组件都有提名对，这些对将成为数据流组件的选定对。lite代理为所有数据流的所有组件生成选定对后，ICE会话状态将设置为"已完成"。

如果对等节点是lite代理，则代理将本地候选项与具有相同数据流，同时具有相同组件、传输协议和IP地址族的远程候选项配对。对于每个数据流的每个组件，如果只有一个候选对，则该对将添加到有效列表中。如果有多个对，建议代理遵循RFC6724的规范选择一对并将其添加到有效列表中。

如果所有数据流的所有组件都有一对，则ICE处理的状态将完成。否则，控制代理必须发送更新的候选项列表，以协调选择不同候选对的不同代理。ICE处理仅在更新的候选项交换完成后完成。

## 8.3 释放候选项

### 8.3.1 完整实现的过程

本节中的规则描述代理何时可以安全地停止发送或接收未成为选定候选项的候选项检查（即与选定对无关）以及何时释放候选项。

一旦清单达到"已完成"状态，代理应再等待三秒钟，然后停止响应检查或生成触发检查，以检查所有未成为选定候选项的本地候选项。一旦所有ICE会话停止使用给定的本地候选项（候选项可能由多个ICE会话使用，例如，在分叉方案中），代理可以释放该候选项。三秒钟的延迟用来处理积极提名的情况，选定对可在ICE完成后快速更改。

释放服务器反射候选项永远不是明确的；它发生在缺乏保活的情况下。

### 8.3.2 lite实现的过程

Lite实现可以释放未成为选定候选项的候选项，一旦ICE对使用这些候选项的所有ICE会话处理达到"完成"状态。

# 9 ICE重新启动

ICE代理可能重新启动现有数据流的ICE。ICE重新启动会导致刷新数据流所有以前的状态，除了代理的角色。ICE重新启动和全新数据会话的唯一区别是，在重新启动期间，可以继续使用现有数据会话发送数据，而全新数据会话始终需要确定角色。

只能通过使用ICE重新启动来完成以下操作（代理必须使用ICE重新启动才能执行此操作）：

* 更改数据流的目标。
* 从lite实现更改为完整实现。
* 从完整实现更改为lite实现。

要重新启动ICE，代理必须更改要重新启动的数据流的口令和用户名片段。

重新启动ICE时，为新ICE会话设置的候选项可能包括当前ICE会话中使用的部分、无或所有候选项。

如第6.1.1节所述，代理不得将重新确定角色作为ICE重新启动的一部分，除非满足需要重新确定角色的某些准则。

# 10 ICE选项

本节定义一个新ICE选项，"ice2"。当ICE代理在候选项交换中包含"ice2"时，ICE选项指示它遵循此规范。例如，代理不会使用RFC 5245中定义的积极提名程序。此外，它将确保符合RFC 5245的对等节点也不会按照RFC 5245第14节的要求，对接收未知ICE选项的对等节点使用积极提名。

遵循此规范的代理必须使用"ice2"选项告知对等节点合规性。

注："ice2"选项的编码，以及用于将其发送到对等节点的消息都与协议相关。SDP[RFC4566]的编码在[ICE-SIP-SDP]中定义。

# 11 保活机制

所有终结点必须为每个数据会话发送保活。保活的目的是为数据会话保持NAT绑定活动状态。保活应使用其对等节点支持的格式发送。ICE终结点允许基于STUN的UDP流保活，因此，当ICE代理是完整实现并与支持ICE的对等节点（lite或完整）通信时，必须使用STUN保活。

如果在过去Tr秒内未通过候选对发送数据包，则代理必须在用于发送数据的每个候选对上发送保活。代理应使用15秒的Tr值。代理可以使用更大的值，但不得使用小于15秒的值。

为数据流生成选定的对后，则仅在选定对上发送保活。

如果删除数据流，代理必须停止在数据流上发送保活。如果ICE会话终止，代理必须停止在所有数据流上发送保活。

代理可能设置Tr为其它值，例如，基于配置或网络/NAT特征。例如，如果代理具有发现中间NAT的绑定生存期的动态方法，则可以使用该值来确定Tr。在控制更严格的网络环境中部署ICE的管理员应将Tr设置为其环境中尽可能长的持续时间。

当用STUN实现保活时，应使用STUN绑定指示[RFC5389]。该指示不得使用任何身份验证机制。它应包含用来消除多路复用的FINGERPRINT属性，但不应包含任何其他属性。它仅用于保持NAT绑定的生存。绑定指示采用与数据传输相同的本地和远程候选项发送。尽管绑定指示用于保活，但代理还必须准备好接收连接检查。如果收到连接检查，则生成响应，如[RFC5389]中所述，除此之外对ICE处理没有影响。

默认情况下，代理必须使用STUN保活。单个ICE用法和ICE扩展可能采用与使用/扩展相关的保活。

# 12 数据处理

## 12.1 发送数据

为数据流生成选定对之前，ICE代理可在任何有效对上发送数据。

为数据流生成选定对后，ICE代理必须仅在这些对上发送数据。

代理将数据从本地候选项的Base发送到远程候选项。 对于本地中继候选项，数据使用[RFC5766]中定义的过程通过Base（位于TURN服务器中）转发。

如果本地候选项是中继候选项，则建议代理在TURN服务器上创建面向远程候选项的通道。这是使用[RFC5766]第11节中定义的通道创建过程完成的。

数据流组件的选定对为：

* 空。如果数据流的清单状态为"正在运行"，并且由于ICE重新启动，该组件没有以前的选定对；
* 数据流组件以前的一个选定对。如果数据流的清单状态为"正在运行"，并且由于ICE重新启动，该组件具有一个以前的选定对。

除非代理能够为与数据流关联的每个组件生成选定对，否则代理不得继续为与该数据流关联的任何组件发送数据。

### 12.1.1 lite实现的过程

lite实现在该数据流每个组件的有效列表包含候选对之前，不得发送数据。发生这种情况后，ICE代理可以开始发送数据包。为此，它将数据发送到对中的远程候选项（数据包的目标地址和端口设置为与该远程候选项相等），并通过与发送数据的候选对关联的Base发送。对于中继候选项，数据从代理发送，并使用[RFC5766] 中定义的过程通过Base（位于TURN服务器中）转发。

## 12.2 接收数据

即使ICE代理仅允许使用有效的候选对发送数据（并且，一旦生成选定对，仅在该选定对上），ICE实现默认情况下应准备接收与对等节点的最新候选项交换中提供的任何候选项的数据。ICE用法可以定义不同的规则，例如，定义在为数据流生成选定对之前不发送数据。

当代理收到具有特定RTP/RTCP数据流的新源或目标IP地址的RTP数据包时，建议代理调整其抖动缓冲区。

RFC 3550的第8.2节描述了一种用于检测同步源（SSRC）碰撞和循环的算法。该算法部分基于使用同一SSRC查看不同的源传输地址。但是，当使用ICE时，数据流在候选项之间切换时会发生此类更改。对于进行媒体数据传输的STUN交换，代理将可以确定数据流来自相同的对等节点。因此，如果源传输地址更改，但媒体数据包来自同一对等代理，则不能将其视为SSRC冲突。

# 13 可扩展性考虑

此规范对会话中的两个ICE代理如何协调以得出为数据选定候选对集进行了非常具体的描述。可能未来的规范会改变这些算法，无论是计时器调整之类简单的更改，还是优先级算法改造这样重大的变化。进行更改时，在会话中提供两个代理之间的互操作性至关重要。

首先，ICE提供了ICE选项概念。对ICE的每个扩展或更改都与ICE选项相关联。当代理支持此类扩展或更改时，它将ICE选项作为候选项交换的一部分向对等代理提供。

实现互操作性的一个复杂因素是ICE依赖于在两个代理上运行的分布式算法收敛到一组商定的候选对。如果两个代理运行不同的算法，则很难保证收敛到同一候选对。第8节中描述的提名程序通过将选择算法完全委派给控制代理来消除一些紧密协调的问题，这样即使两个ICE代理使用不同的对优先级算法，也会完全收敛。实现收敛的关键之一是触发检查，以确保提名对由两个代理验证。

ICE还可扩展到RTP以外的其他数据流和UDP以外的传输协议。 对于非RTP数据流，ICE的扩展需要指定它们使用多少个组件，并为其分配组件ID，从1开始，用于最重要的组件。新传输协议的规范必须定义ICE处理中的各个步骤与UDP有何不同。

# 14 设置Ta和RTO

## 14.1 一般情况

在ICE采集阶段（第5.1.1节），当ICE执行连接检查（第7节）时，ICE代理触发STUN和TURN事务。这些事务以Ta指示的速率定时进行，每个事务的转播间隔根据STUN事务（RTO）[RFC5389]的重新传输计时器计算。

本节介绍在ICE采集阶段以及ICE连接检查期间如何计算Ta和RTO值。

注：以前，在RFC 5245中，根据ICE是否用于实时数据流（例如RTP），为计算Ta和RTO定义了不同的公式。

下面的公式会导致一种行为，代理在执行重新传输之前，将发送其每个连接检查的第一个数据包。这可以从RTO的公式（表示重新传输间隔）中看到。这些公式随N（要执行的检查数）变化。因此，ICE很好的保持了恒定的速率，但它对数据包丢失变得更加敏感。丢失任何用于连接检查的第一个数据包可能会导致验证该对持续很长时间，相反，优先级较低的检查（但没有数据包丢失）更有可能首先完成。这导致ICE性能不理想，选定优先级较低的对而不是高优先级对。

## 14.2 Ta

ICE代理应使用默认Ta值50 ms，但可能根据关联数据的特征使用其它值。

如果代理想要使用默认值以外的Ta值，则代理必须在ICE会话建立期间向其对等节点指示建议的值。两个代理必须使用建议值的较高值。如果一个代理不提出建议值，则在比较哪个值较高时，该代理将使用默认值。

无论每个代理选择的Ta值如何，所有代理的所有事务（如果给定实现运行多个并发代理）的组合不能超过每5毫秒发送一次（有一个全局Ta值用于设置所有代理）。有关使用ICE的值5毫秒的背景，请参阅附录B.1。

注：附录C显示了使用不同Ta值所需的带宽的示例。

## 14.3 RTO

在ICE采集阶段，ICE代理使用以下公式计算RTO值：

RTO = MAX (500ms, Ta \* (Num-Of-Cands))

其中，Num-Of-Cands: 服务器反射和中继候选项的数量

对于连接检查，代理使用以下公式计算RTO值：

RTO = MAX (500ms, Ta \* N \* (Num-Waiting + Num-In-Progress))

其中，N：要执行的连接检查的总数。

Num-Waiting: 清单中处于"等待"状态的检查数。

Num-In-Progress: 清单中处于"正在进行"状态的检查数。

请注意，随着"等待"和"正在进行"状态的检查数改变，每个事务的RTO将有所不同。

代理可以使用其他机制计算RTO值。代理不得使用小于500毫秒的RTO值。

# 15 例子

本节显示了两个ICE示例：一个使用IPv4地址，另一个使用IPv6地址。

为便于理解，使用具有助记符名称的变量表示传输地址。名称的格式为实体-类型-序列号："实体"是指传输地址位于其上的IP地址，传输地址为"L"、"R"、"STUN"或"NAT"。"类型"可以是"PUB"（对于公共传输地址）或"PRIV"（对于私有传输地址[RFC1918]）。最后，"序列号"是特定实体上相同类型的每个传输地址不同的序列编号。每个变量都有一个IP地址和端口，分别用varname.IP和varname.PORT表示，其中varname是变量的名称。

在调用流中，STUN消息使用多个属性进行标注。"S="属性指示消息的源传输地址。"D="属性指示消息的目标传输地址。"MA="属性用于STUN绑定响应消息，索引映射的地址。"USE-CAND"表示存在使用-候选项属性。

调用流示例省略了STUN身份认证操作，专注于两个完整实现之间的单个数据流。

## 15.1 IPv4地址示例

下面的示例使用图7所示的拓扑。



图 7：拓扑示例

在此示例中，ICE代理L和R是完整的ICE实现。两个代理都具有一个IPv4地址，并且都配置了相同的STUN服务器。NAT具有与终结点无关的映射属性和与地址相关的筛选属性。ICE代理、STUN服务器和NAT的IP地址如下所示：



L NAT STUN R

|STUN alloc. | | |

|(1) STUN Req | | |

|S=$L-PRIV-1 | | |

|D=$STUN-PUB-1 | | |

|------------->| | |

| |(2) STUN Req | |

| |S=$NAT-PUB-1 | |

| |D=$STUN-PUB-1 | |

| |------------->| |

| |(3) STUN Res | |

| |S=$STUN-PUB-1 | |

| |D=$NAT-PUB-1 | |

| |MA=$NAT-PUB-1 | |

| |<-------------| |

|(4) STUN Res | | |

|S=$STUN-PUB-1 | | |

|D=$L-PRIV-1 | | |

|MA=$NAT-PUB-1 | | |

|<-------------| | |

|(5) L's Candidate Information| |

|------------------------------------------->|

| | | | STUN

| | | | alloc.

| | |(6) STUN Req |

| | |S=$R-PUB-1 |

| | |D=$STUN-PUB-1 |

| | |<-------------|

| | |(7) STUN Res |

| | |S=$STUN-PUB-1 |

| | |D=$R-PUB-1 |

| | |MA=$R-PUB-1 |

| | |------------->|

|(8) R's Candidate Information| |

|<-------------------------------------------|

| | (9) Bind Req |Begin

| | S=$R-PUB-1 |Connectivity

| | D=$L-PRIV-1 |Checks

| | <-------------------|

| | Dropped |

|(10) Bind Req | | |

|S=$L-PRIV-1 | | |

|D=$R-PUB-1 | | |

|------------->| | |

| |(11) Bind Req | |

| |S=$NAT-PUB-1 | |

| |D=$R-PUB-1 | |

| |---------------------------->|

| |(12) Bind Res | |

| |S=$R-PUB-1 | |

| |D=$NAT-PUB-1 | |

| |MA=$NAT-PUB-1 | |

| |<----------------------------|

|(13) Bind Res | | |

|S=$R-PUB-1 | | |

|D=$L-PRIV-1 | | |

|MA=$NAT-PUB-1 | | |

|<-------------| | |

|Data | | |

|===========================================>|

| | | |

| |(14) Bind Req | |

| |S=$R-PUB-1 | |

| |D=$NAT-PUB-1 | |

| |<----------------------------|

|(15) Bind Req | | |

|S=$R-PUB-1 | | |

|D=$L-PRIV-1 | | |

|<-------------| | |

|(16) Bind Res | | |

|S=$L-PRIV-1 | | |

|D=$R-PUB-1 | | |

|MA=$R-PUB-1 | | |

|------------->| | |

| |(17) Bind Res | |

| |S=$NAT-PUB-1 | |

| |D=$R-PUB-1 | |

| |MA=$R-PUB-1 | |

| |---------------------------->|

|Data | | |

|<===========================================|

| | | |

.......

| | | |

|(18) Bind Req | | |

|S=$L-PRIV-1 | | |

|D=$R-PUB-1 | | |

|USE-CAND | | |

|------------->| | |

| |(19) Bind Req | |

| |S=$NAT-PUB-1 | |

| |D=$R-PUB-1 | |

| |USE-CAND | |

| |---------------------------->|

| |(20) Bind Res | |

| |S=$R-PUB-1 | |

| |D=$NAT-PUB-1 | |

| |MA=$NAT-PUB-1 | |

| |<----------------------------|

|(21) Bind Res | | |

|S=$R-PUB-1 | | |

|D=$L-PRIV-1 | | |

|MA=$NAT-PUB-1 | | |

|<-------------| | |

| | | |

图 8：示例流

消息1-4：代理L从其本地IP地址采集主机候选项，并从该地址向STUN服务器发送STUN绑定请求。请求创建NAT绑定。绑定的NAT公共IP地址将成为代理L的服务器反射候选项。

消息5：代理L使用与ICE所采用的关联的信令协议将其本地候选项信息发送到代理R。

消息6-7：代理R从其本地IP地址采集主机候选项，并从中向STUN服务器发送STUN绑定请求。由于代理R不在NAT后面，R的服务器反射候选项与主机候选项相同。

消息8：代理R使用与ICE所采用的关联的信令协议将其本地候选信息发送到代理L。

由于两个代理都是完整的ICE实现，因此启动代理（代理L）成为控制代理。

代理L和R分别进行候选项配对。两个代理最初都有两对。但是，代理L将修剪包含其服务器反射候选的对，导致只有一个（L1）。在代理L中，此对具有$L\_PRIV\_1的本地候选项和$R\_PUB\_1的远程候选项。在代理R处，有两对。优先级最高的对（R1）具有$R\_PUB\_1的本地候选项和$L\_PRIV\_1的远程候选项，第二对（R2）的本地候选项为$R\_PUB\_1，远程候选$NAT\_PUB\_1。如下所示（对号仅用于参考目的）：



消息9：代理R启动对R1的连接检查。由于对的远程候选项是代理L的专用地址，因此检查不会成功，因为请求无法从R路由到L，并且将由网络删除。

消息10-13：代理L启动对L1的连接检查。检查成功，L将创建新对（L2）。 新对的本地候选项为$NAT\_PUB\_1，远程候选项为$R\_PUB\_1。对（L2）添加到代理L的有效列表中，代理L现在可以发送和接收对（L2）上的数据（如果愿意）。



消息14-17：当代理R收到来自代理L的绑定请求（消息11）时，它将启动触发连接检查。配对与代理R的现有对之一（R2）匹配。 检查成功，并且对（R2）添加到代理R的有效列表中，现在可以发送和接收对（R2）上的数据，如果愿意的话。



消息18-21：在某些时候，控制代理（代理L）决定在有效列表中提名一对（L2）。它对对（L2）执行连接检查，并在绑定请求中包括使用-候选项属性。当检查成功时，代理L将对（L2）的提名标志值设置为"true"，代理R将匹配对（R2） 的提名标志值设置为"true"。由于没有与流关联的更多组件，因此提名对将成为选定对。因此，此流的处理将进入"已完成"状态。ICE过程也进入"已完成"状态。

## 15.2 IPv6地址示例

下面的示例使用图9所示的拓扑。



图 9：拓扑示例

在此示例中，ICE代理L和R是完整的ICE实现。两个代理都有一个IPv6地址，并且都配置了相同的STUN服务器。ICE代理和STUN服务器的IP地址如下所示：



L STUN R

|STUN alloc. | |

|(1) STUN Req | |

|S=$L-PUB-1 | |

|D=$STUN-PUB-1 | |

|---------------------------->| |

|(2) STUN Res | |

| S=$STUN-PUB-1 | |

| D=$L-PUB-1 | |

| MA=$L-PUB-1 | |

|<----------------------------| |

|(3) L's Candidate Information| |

|------------------------------------------->|

| | | STUN

| | | alloc.

| |(4) STUN Req |

| |S=$R-PUB-1 |

| |D=$STUN-PUB-1 |

| |<-------------|

| |(5) STUN Res |

| |S=$STUN-PUB-1 |

| |D=$R-PUB-1 |

| |MA=$R-PUB-1 |

| |------------->|

|(6) R's Candidate Information| |

|<-------------------------------------------|

|(7) Bind Req | |

|S=$L-PUB-1 | |

|D=$R-PUB-1 | |

|------------------------------------------->|

|(8) Bind Res | |

|S=$R-PUB-1 | |

|D=$L-PUB-1 | |

|MA=$L-PUB-1 | |

|<-------------------------------------------|

|Data | |

|===========================================>|

| | |

|(9) Bind Req | |

|S=$R-PUB-1 | |

|D=$L-PUB-1 | |

|<-------------------------------------------|

|(10) Bind Res | |

|S=$L-PUB-1 | |

|D=$R-PUB-1 | |

|MA=$R-PUB-1 | |

|------------------------------------------->|

|Data | |

|<===========================================|

| | |

.......

| | |

|(11) Bind Req | |

|S=$L-PUB-1 | |

|D=$R-PUB-1 | |

|USE-CAND | |

|------------------------------------------->|

|(12) Bind Res | |

|S=$R-PUB-1 | |

|D=$L-PUB-1 | |

|MA=$L-PUB-1 | |

|<-------------------------------------------|

| | |

图10：流示例

消息1-2：代理L从其本地IP地址采集主机候选项，并通过该地址向STUN服务器发送STUN绑定请求。由于代理L不在NAT后面，因此L的服务器反射候选项将与主机候选项相同。

消息3：代理L使用与ICE使用关联的信令协议将其本地候选项信息发送到代理R。

消息4-5：代理R从其本地IP地址采集主机候选项，并从中向STUN服务器发送STUN绑定请求。由于代理R不在NAT后面，R的服务器反射候选项将与主机候选项相同。

消息6：代理R使用与ICE使用关联的信令协议将其本地候选信息发送到代理L。

由于两个代理都是完整的ICE实现，因此启动代理（代理L）成为控制代理。

代理L和R都进行候选项配对。两个代理最初各有一对。在代理L中，对（L1）具有$L\_PUB\_1的本地候选项和$R\_PUB\_1的远程候选项。在代理R中，对（R1）具有$R\_PUB\_1的本地候选项和$L\_PUB\_1的远程候选项。对如下所示（对号仅用于参考目的）：



消息7-8：代理L启动对L1的连接检查。检查成功，对（L1）将添加到代理L的有效列表中。代理L现在可以发送和接收对（L1）上的数据（如果愿意）。



消息9-10：当代理R收到来自代理L的绑定请求时（消息7），它将启动触发连接检查。对与代理R的现有对（R1）匹配。检查成功，对（R1）将添加到代理R的有效列表中。代理R现在可以发送和接收对（R1）上的数据（如果愿意）。



消息11-12：在某些时候，控制代理（代理L）决定在有效列表中提名一对（L1）。它对对（L1）执行连接检查，并在绑定请求中包括使用-候选项属性。如果检查成功，代理L将对（L1）的提名标志值设置为"true"，代理R将匹配对（R1）的提名标志值设置为"true"。

由于没有与流关联的更多组件，因此提名对将成为选定对。因此，此流的处理将进入"已完成"状态。ICE过程也进入"已完成"状态。

# 16 STUN 扩展

## 16.1 属性

此规范定义了四个STUN属性：优先级、使用-候选项、ICE-控制和ICE-受控。

如果此检查发现一个优先级，则优先级属性指示与对等反射候选项关联的优先级。它是一个32位无符号整数，属性值为0x0024。

使用-候选项属性表示此检查产生的候选对将用于数据传输。该属性没有内容（属性的长度字段为零）;它充当一个标志。其属性值为0x0025。

ICE-受控属性存在于绑定请求中。该属性指示客户端认为它当前处于受控角色。属性的内容是网络字节顺序中64位无符号整数，其中包含一个随机数。该数字用于解决角色冲突，此时称之为"决胜属性值"。ICE代理必须在ICE会话中对所有绑定请求使用相同的编号，除非它已收到487响应，在这种情况下，它必须更改该编号（第7.2.5.1节）。当ICE重新启动时，代理可能会更改编号。

ICE-控制属性存在于绑定请求中。该属性指示客户端认为它当前处于控制角色。属性的内容是网络字节顺序中64位无符号整数，其中包含一个随机数。对于 ICE-受控属性，该编号用于解决角色冲突。代理必须在ICE会话内对所有绑定请求使用相同的编号，除非它已收到487响应，在这种情况下，它必须更改该编号（第7.2.5.1节）。当ICE重新启动时，代理可能会更改编号。

## 16.2 新的错误响应代码

此规范定义了唯一一个错误响应代码：

487（角色冲突）：绑定请求包含ICE-控制或ICE-受控属性，指示ICE角色与服务器冲突。远程服务器比较客户端和服务器的绑定设置值，并确定客户端需要切换角色。

# 17 运营考虑

本节讨论与网络运营商相关的问题，其中网络终结点使用ICE。

## 17.1 NAT和防火墙类型

ICE旨在与现有的NAT和防火墙设备配合使用。因此，为了便于部署ICE，无需更换或重新配置现有的防火墙和NAT设备。事实上，ICE被开发用于在IP语音（VoIP）运营商无法控制IP网络基础结构（包括防火墙和NAT）的环境中部署。

ICE在NAT设备"符合"要求的环境中工作最佳，即符合[RFC4787]和[RFC5382]中定义的建议。在行为兼容的NAT网络中，ICE无需使用TURN服务器即可工作，从而提高语音质量、缩短呼叫设置时间并减少网络运营商的带宽需求。

## 17.2 带宽要求

ICE的部署可以与运营商需要考虑的可用网络容量进行多次交互。

### 17.2.1 STUN和TURN服务器容量规划

首先，ICE使用TURN和STUN服务器，这些服务器通常位于数据中心。STUN服务器需要的带宽相对较少。对于每个数据流的每个组件，每个客户端到STUN服务器将有一个或多个STUN事务。在基本的仅语音的IPv4 VoIP部署中，每个呼叫将有四个事务（一个用于RTP，一个用于RTCP，分别对应于呼叫方和被叫方）。每个事务都是单个请求和单个响应，前者为20字节长，后者为28字节。

因此，如果系统有N个用户，并且每个用户在繁忙小时内进行四次呼叫，则需要N=1.7bps。对于100万用户，就是1.7 Mbps，这是一个很小的数字（相对而言）。

TURN的流量庞大的多。除了实际数据的流量外，TURN服务器还将处理类似于STUN的流量（事实上，如果部署了TURN，则不需要单独的STUN服务器）。数据中继需要TURN的呼叫量高度依赖于网络拓扑，并且会随时间而变化。在100%符合NAT要求的网络中，它完全为零。

上述规划考虑在多媒体情景（例如音频和视频会议）和会话参与者数量增加时变得尤其重要。

### 17.2.2 采集和连接检查

采集候选项和执行连接检查的过程会占用大量带宽。ICE的设计旨在调整这两个流程的节奏。采集和连接检查的目的是，实现与ICE过程完成后数据流量本身消耗的大致相同的带宽生成流量。这样做是为了确保，如果网络旨在支持特定类型的通信流量（语音、视频或纯文本），它将有足够的能力支持ICE对该数据的检查。一旦ICE结束，随后的ICE保活将导致总带宽利用率略有增加；而且，这通常是一个极小的增加。

由于采集和检查阶段而导致的拥塞已证明是部署中未使用节奏控制的问题。通常，终结点以尽可能快的速度向网络发送检查会导致访问链路变得拥塞。因此，网络运营商需要确保其ICE实现支持节奏控制。尽管这种节奏确实增加了呼叫设置时间，但它使ICE网络友好且更易于部署。

### 17.2.3 保活

STUN保活（以STUN绑定指示的形式）在数据会话间发送。但是，只有在没有实际数据流量的情况下才会发送。在具有连续介质且不使用语音活动检测（VAD）的部署中，或者使用VAD且短间隔（最大1秒）舒适噪声的部署中，从不使用保活，并且带宽使用率不会增加。当VAD在不使用舒适噪音的情况下使用时，静默期间将发送保活。这导致每15-20秒发送一个保活数据包，远远低于有语音时每20-30毫秒发送的数据包。因此，保活对容量规划没有任何实际影响。

## 17.3 ICE和ICE-Lite

部署通常利用ICE和ICE-lite混合实现相互操作。它们已被明确设计这样做。

但是，ICE-lite只能在有限的应用场景中部署。这些场景以及其中涉及的注意事项载于附录A。

## 17.4 故障排除和性能管理

ICE利用端到端连接检查，并将大部分处理放在终结点上。这给网络运营商带来了挑战—他们如何对ICE部署进行故障排除？他们怎么知道ICE的表现？

ICE具有内置功能来帮助解决这些问题。信号服务器（通常部署在网络运营商的数据中心）将看到传输ICE参数的候选项交换的内容。这些参数包括每个候选项的类型（主机、服务器反射或中继）及其相关地址。ICE处理完成后，将发生更新的候选项交换，发出所选地址（及其类型）的信号。发送更新信令完全是为了通知网络设备（如连接到信令的诊断工具）有关ICE处理的结果。

因此，通过信令服务器生成的日志，网络运营商可以观察每次呼叫使用的候选项类型以及ICE选择了哪些地址。这是有助于评估ICE性能的主要信息。

## 17.5 终结点配置

ICE依赖于将多个数据配置到终结点中。此配置数据包括计时器、TURN服务器的凭据以及STUN和TURN服务器的主机名。ICE本身不为此配置提供机制。相反，假定此信息附加到用于配置终结点中所有其他参数的任何机制。对于SIP呼叫，定义了配置框架[RFC6080]标准解决方案。

# 18 IAB的考虑

互联网架构委员会（IAB）研究了"单向自我地址修复"（UNSAF）问题，这是ICE代理试图通过协作协议反射机制[RFC3424]在NAT的另一端确定其地址的一般过程。ICE是执行此类功能的协议的实例。有趣的是，ICE的进程不是单向的，而是双向的，这种差异对IAB提出的问题具有重大影响。事实上，ICE可被视为双向自我地址修复（B-SAF）协议，而不是UNSAF协议。无论如何，IAB规定为此目的制定的任何协议都记录一组特定的注意事项。本节描述如何满足这些要求。

## 18.1 问题定义

从RFC 3424起，UNSAF的任何提案都需要提供：

* 精确界定一个具体、范围有限的问题，该问题将由UNSAF提案解决。短期修复不应一般化到解决其他问题。这种一般化导致对所谓的短期修复的长期依赖和使用—意味着将其称为"短期"不再准确。

ICE正在解决的具体问题包括：

* 为两个对等节点提供一种方法，以确定可用于通信的传输地址集。
* 为代理提供一种方法，以确定它希望与其通信的另一个对等节点可访问的地址。

## 18.2 退出策略

从RFC 3424起，UNSAF的任何提案都需要提供：

* 退出策略/过渡计划的说明。更好的短期修复是那些随着技术的发展，使用会越来越少。

ICE本身不容易被淘汰。但是，即使在全球连接的Internet中，它还是有用的，可以作为检测路由器故障导致暂时中断连接的方法。ICE还有助于防止某些与NAT无关的安全攻击。然而，ICE的作用是帮助逐步取消其他UNSAF机制。ICE有效地从这些机制中挑选，优先考虑那些更好的机制，并取消更坏机制的优先级。随着引入IPv6，NAT开始消散，服务器反射和中继候选项（两种UNSAF地址形式）根本不再使用，因为本机主机候选项存在优先级较高的连接。因此，服务器的使用越来越少，当服务器的使用达到零时，它们最终可以被删除。

事实上，ICE可以帮助从IPv4过渡到IPv6。当两个双栈主机与SIP（使用IPv6）通信时，它可用于确定是否使用IPv6或IPv4。它还允许具有6to4和本机v6连接的网络来确定在与对等节点通信时要使用的地址。

## 18.3 ICE引入的脆弱性

从RFC 3424起，UNSAF的任何提案都需要提供：

* 讨论可能使系统更加"脆弱"的具体问题。例如，涉及在多个网络层使用数据的方法会创建更多的依赖项，增加调试难题，并加大转换难度。

ICE实际上消除了现有的UNSAF机制中的脆弱性。特别是，经典的STUN（如RFC 3489所述）有几个脆性点。其中之一是发现过程，需要ICE代理尝试对它后面的NAT进行分类。此过程容易出错。对于ICE，根本不使用该发现过程。它不是单方面评估地址的有效性，而是通过测量与对等节点的连接动态确定其有效性。确定连接性的过程非常可靠。

在经典STUN和任何其他单方机制中，另一个脆弱性是它绝对依赖于额外的服务器。ICE使用服务器分配单边地址，但它允许代理在可能的情况下直接连接。因此，在某些情况下，STUN服务器的失败仍允许使用ICE进行呼叫。

经典 STUN 中另一个脆弱性是，它假定STUN服务器位于公共互联网上。有趣的是，对于ICE，这是没有必要的。在各种地址领域可以有许多STUN服务器。ICE将发现提供可用地址的一个地址。

在经典STUN中，最棘手的脆弱性是它不能在所有网络拓扑中工作。如果每个代理和STUN服务器之间存在共享NAT，则传统STUN可能无法正常工作。使用ICE则不存在该限制。

经典STUN还引入了一些安全性问题。幸运的是，ICE也减轻了这些问题。

因此，ICE可以修复经典STUN中引入的脆弱性，并且不会向系统引入任何额外的脆弱性。

这些改进的缺点是ICE会增加会话建立时间。

## 18.4 长期解决方案的要求

从RFC 3424起，UNSAF的任何提案都需要提供以下内容：

* 确定对长期、健全的技术解决方案的要求; 有助于找到正确的长期解决方案。

我们从RFC 3489中得出的结论保持不变。然而，我们认为ICE实际上有帮助，因为我们相信它可以是长期解决方案的一部分。

## 18.5 现有NAPT盒的问题

从RFC 3424起，UNSAF的任何提案都需要提供：

* 讨论上述实际问题对现有已部署的NA[P]T的影响以及用户体验报告。

目前市面上部署了许多NAT设备，试图提供"通用的"ALG功能。这些通用ALG 以文本或二进制形式在数据包中搜索IP地址，如果它们与绑定匹配，则重写它们。这干扰了经典的STUN。但是，STUN[RFC5389]的更新使用一种编码对通用ALG隐藏这些二进制地址。

现有的NAPT设备具有不确定性，并且对于基于UDP的绑定通常具有较短的过期时间。这要求实现定期发送保活以维护这些绑定。ICE使用默认值15 s，这是一个非常保守的估计值。随着时间的推移，随着NAT设备符合[RFC4787]的行为特性，这种最小保活将变得确定和众所周知，ICE计时器可以进行调整。有一种方法来发现和控制最小保活间隔将好得多。

# 19 安全考虑

## 19.1 IP地址隐私

探测候选项程序会向任何网络侦听攻击者显示客户端及其对等节点的源地址，交换候选项的过程会向任何能够看到协商的攻击者显示这些地址。某些地址（如通过VPN用户的本地接口采集的服务器反射地址）可能是敏感信息。如果无法缓解这些潜在攻击，ICE使用可以定义限制向协商和/或探测过程显示哪些地址的机制。各个实现还可以有专门的规则来控制显示哪些地址。例如，[WebRTC-IP-HANDLING]提供了通过ICE向WebRTC应用程序披露IP地址隐私方面的其他信息。建议在出现此类问题的ICE实现提供编程或用户界面，以控制用于生成候选的网络接口。

根据对等节点提供的候选项类型以及针对这些候选项执行的连接检查结果，对等节点可以确定本地网络的特征，例如，对等节点明显显示不同的时间。在限制范围内，对等节点可以探测本地网络。

ICE系统中可能存在几种类型的攻击。这些小节考虑这些攻击及其对策。

## 19.2 对连接检查的攻击

攻击者可能会尝试中断STUN连接检查。如果所有这些攻击欺骗了ICE代理，使其认为连接检查的结果有误。根据攻击类型，攻击者需要具有不同的功能。在某些情况下，攻击者需要位于连接检查的路径上。在其他情况下，攻击者不需要在路径上，只要能够生成STUN连接检查。虽然对连接检查的攻击通常由网络实体执行，但如果攻击者能够控制终结点，则有可能触发连接检查攻击。攻击者可能尝试并导致的虚假结论包括：

* 假无效：攻击者可以愚弄一对代理，使其认为候选对无效，而实际情况并非如此。这可用于使代理首选其他候选项（例如攻击者注入的候选项），或通过强制所有候选项失败来中断呼叫。
* 错误有效：攻击者可以愚弄一对代理，使其认为候选对有效，而实际情况并非如此。这可能会导致代理继续会话，但无法接收任何数据。
* 错误的对等反射候选项：攻击者可能导致代理意外发现新的对等反射候选项。这可用于将数据流重定向到DoS目标或攻击者，用于窃听或其他目的。
* 虚假候选项上的"错误有效"：攻击者已使代理确信，存在一个候选项具有实际上未路由到该代理的地址（例如，通过注入错误的对等反射候选项或假服务器反射候选项）。然后，攻击者发起攻击，迫使代理相信该候选项有效。如果攻击者可能在一个虚假的候选项上导致错误的对等反射候选项或错误有效，则可以发起[RFC5389]中描述的任何攻击。

要强制错误无效的结果，攻击者必须等待发送自其中一个代理的连接检查。 此时，攻击者需要注入一个假的响应与不可恢复的错误响应（如400），或丢弃响应，使其永远不会到达代理。但是，由于候选项实际上有效，因此原始请求可能会到达对等代理并产生成功的响应。攻击者需要通过DoS攻击、第2层网络中断或其他技术强制丢弃此数据包或其响应。如果不执行此操作，成功响应也会到达发起方，提醒其可能的攻击。攻击者产生虚假响应的能力通过STUN短期凭据机制得到缓解。为了处理此响应，攻击者需要口令。如果候选项交换信令受到保护，攻击者没有口令，其响应将被丢弃。

虚假的ICMP硬错误（类型3，代码2-4）也可用于创建虚假的无效。如果ICE代理对这些ICMP错误实现了响应，则攻击者可以生成ICMP消息，该消息将传递到发送连接检查的代理。代理对ICMP错误消息的验证是其唯一的防御。对于类型3代码4，外部IP标头不提供任何验证，除非连接检查使用DF=0发送。对于由主机发布的代码2或3，地址应为远程代理的任何主机、反射式或中继候选IP地址。ICMP消息包括触发错误的消息的IP标头和UDP标头。这些字段也需要验证。IP目标和UDP目标端口需要与目标候选地址和端口或候选项的基本地址匹配。源IP地址和端口可以是发送连接检查的代理的相同Base地址的任何候选地址。因此，任何攻击者都有权访问候选项交换，他们将获得必要的信息。因此，验证是一种弱防御，并且对于没有源地址验证的网络节点的偏离路径攻击者，也可以发送欺骗性ICMP攻击。

强制伪造有效结果的工作方式类似。攻击者需要等待来自每个代理的绑定请求，并注入假的成功响应。同样，由于STUN的短期凭据机制，为了攻击者注入有效的成功响应，攻击者需要口令。或者，攻击者可以将有效的成功响应（例如，使用隧道机制）路由到代理（通常被网络丢弃或拒绝）。

强制虚假的对等反射候选结果可以通过假请求或响应或重播来完成。我们首先考虑虚假请求和响应情况。它要求攻击者向具有源IP地址和端口的一个代理发送绑定请求，以便获取假候选项。此外，攻击者需要等待来自其他代理的绑定请求，并生成包含假候选项的XOR-MAPPED-ADDRESS属性的虚假响应。与此处描述的其他攻击一样，STUN消息完整性机制和安全候选项交换可缓解此攻击。

使用数据包重播强制错误的对等反射候选结果是不同的。攻击者等待，直到其中一个代理发送检查。它拦截此请求，并将其重播到具有假源IP地址的其他代理。它还需要通过启动DoS攻击导致丢弃数据包或使用第2层机制强制丢弃数据包来防止原始请求到达远程代理。重播的数据包在其他代理处接收并接受，因为完整性检查通过（完整性检查不能也不涵盖源IP地址和端口）。然后，它响应。此响应将包含与假候选项的XOR-MAPPED-ADDRESS地址，并发送给该假候选项。然后，攻击者需要接收它并将其中继到发起者。

然后，另一个代理将对该假候选项启动连接检查。此验证需要通过。这要求攻击者强制对假候选项进行虚假有效。使用STUN的完整性机制和候选项交换来阻止注入虚假请求或响应以达到此目标。因此，此攻击只能通过重播进行。为此，攻击者需要截取针对此假候选项的检查并将其重播到其他代理。然后，它需要拦截响应并重播该响应。

除非假候选项标识攻击者，否则很难发起此攻击。这是因为它需要攻击者拦截和重播由两个不同的主机发送的数据包。如果两个代理都位于不同的网络上（例如，跨公共Internet），则此攻击可能难以协调，因为它需要同时针对网络不同部分上的两个不同终结点。

如果攻击者本身由假候选项标识，则攻击更易于协调。但是，如果数据路径是安全的（例如，使用安全实时传输协议（SRTP）[RFC3711]），攻击者将无法处理数据包，只能丢弃它们，从而有效地禁用数据流。但是，此攻击要求代理中断数据包，以阻止连接检查到达目标。在这种情况下，如果目标是破坏数据流，则使用相同的机制中断数据流比攻击ICE要容易得多。

## 19.3 攻击服务器反射地址采集

ICE终结点使用STUN绑定请求从STUN服务器采集服务器反射候选项。这些请求不会以任何方式进行身份验证。因此，攻击者可以使用许多技术向客户端提供错误的服务器反射候选项：

* 攻击者可能会危害DNS，导致DNS查询返回恶意的STUN服务器地址。 该服务器可以为客户端提供假服务器反射候选项。DNS安全性可缓解此攻击，但不需要DNSSEC来解决它。
* 能够观察STUN消息（如共享网段上的攻击者，如Wi-Fi）的攻击者可以注入有效的虚假响应，并且客户端将接受这些响应。
* 攻击者可能会危及STUN服务器，并导致它发送具有错误映射地址的响应。

这些攻击所获得的虚假映射地址将在ICE会话的建立中用作服务器反射候选项。要实际将此候选项用于数据，攻击者还需要攻击连接检查，特别是强制对假候选项进行虚假有效检查。如果虚假地址标识第四方（既不是发起方、响应方也不是攻击者），则很难发起此攻击，因为它需要攻击会话中每个ICE代理生成的检查，如果虚假地址标识攻击者本身，则SRTP会阻止它。

如果攻击者选择不攻击连接检查，则其最坏情况是阻止使用服务器反射候选项。但是，如果对等代理至少有一个受攻击的代理可访问的候选项，则STUN连接检查本身将提供可用于交换数据的对等反射候选项。对等反射候选项通常比服务器反射型候选者更受欢迎。因此，仅对STUN地址采集的攻击通常对会话没有任何影响。

## 19.4 对中继候选项采集的攻击

攻击者可能会试图中断中继候选项的采集，迫使客户端相信其具有错误的中继候选项。使用长期凭据对与TURN服务器的交互进行身份验证。因此，注入虚假响应或请求将不起作用。此外，与绑定请求不同，分配请求不容易受到具有修改的源IP地址和端口的重播攻击，因为源IP地址和端口不用于向客户端提供其中继的候选项请求。

即使攻击者使客户端相信错误的中继候选项，连接检查也会导致仅成功时才使用此类候选项。因此，攻击者需要对上述错误候选项发动一个无效的假攻击，这是一个很难协调的攻击。

## 19.5 内部攻击

除了攻击者是第三方试图插入假候选项信息或STUN消息的攻击之外，当攻击者是ICE交换中经过身份验证的有效参与者时，ICE可能会进行攻击。

### 19.5.1 STUN放大攻击

STUN放大攻击类似于"语音锤"攻击，攻击者导致其他代理将语音数据包定向到攻击目标。但是，STUN连接检查不是定向到目标的语音数据包，而是定向到目标。攻击者发送了大量候选项，例如50名。响应代理收到候选项信息并启动检查，这些检查针对目标，因此永远不会生成响应。对于WebRTC，用户可能甚至不知道此攻击正在进行，因为它可能由用户获取的恶意JavaScript代码在后台触发。响应方每Ta ms（例如，Ta=50ms）启动新的连接检查。但是，由于候选项数量众多，重新传输计时器的值很大。因此，数据包将每Ta ms发送一个，然后以增加的间隔发送。因此，STUN不会以比发送数据更快的速度发送数据包，并且STUN数据包仅短暂保留，直到ICE会话失败。尽管如此，这也是一个放大攻击。

消除放大攻击是不可能的，但影响可以通过多种启发式方法减少。ICE代理应将其执行的连接检查总数限制为100。此外，代理会限制接受的候选项数量。

通常，希望避免此类攻击的协议会强制发起者在发送下一条消息之前等待响应。但是，就ICE而言，这是不可能的。因为无法区分以下两种情况：

* 没有响应，因为发起者正被用来对毫无防备的目标发起DoS攻击，而该攻击无法响应。
* 没有响应，因为IP地址和端口无法到达。

在第二种情况下，下次将发送另一检查，而在前一种情况下，不会发送进一步检查。

# 20 IANA的考虑

原始ICE规范注册了四个STUN属性和一个新的STUN错误响应。此处复制STUN属性和错误响应。此外，此规范注册了新的ICE选项。

## 20.1 STUN属性

IANA已注册了四个STUN属性：

0x0024 PRIORITY，优先级

0x0025 USE-CANDIDATE，使用-候选项

0x8029 ICE-CONTROLLED，ICE-受控

0x802A ICE-CONTROLLING，ICE-控制

## 20.2 STUN错误响应

IANA已注册以下STUN错误响应代码：

* 487角色冲突：客户端断言ICE角色（控制或受控）与服务器的角色冲突。

## 20.3 ICE选项

IANA按照[RFC6336]中定义的程序，在"交互式连接建立 （ICE）"注册表的"ICE 选项"子注册表中注册了以下ICE选项。

ICE Option name:

ice2

Contact:

Name: IESG

Email: iesg@ietf.org

Change Controller:

IESG

Description:

ICE选项指示使用ICE选项的ICE代理根据RFC 8445实现。

Reference:

RFC 8445

# 21 相对RFC 5245的变化

此更新的ICE规范的目的是：

* 澄清RFC 5245中的程序。
* 由于发现RFC 5245存在缺陷，以及基于RFC 5245实施和部署ICE应用程序的社区反馈，因此进行技术更改。
* 通过删除SIP和SDP过程，使过程独立于信令协议。特定于信令协议的过程将在单独的使用文档中定义。[ICE-SIP-SDP]定义了SIP和SDP的ICE使用情况。

进行了以下技术更改：

* 删除了积极提名。
* 修改了计算候选对状态和计划连接检查的过程。
* 修改了Ta和RTO的计算过程。
* 删除了活动清单和冻结清单定义。
* 添加了"ice2"ICE选项。
* 修改了IPv6注意事项。
* 删除了使用no-op保活，与非ICE对等节点保活。

# 22 参考文献

## 22.1 规范性参考

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <https://www.rfc-editor.org/info/rfc4941>.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, DOI 10.17487/RFC5389, October 2008, <https://www.rfc-editor.org/info/rfc5389>.

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, DOI 10.17487/RFC5766, April 2010, <https://www.rfc-editor.org/info/rfc5766>.

[RFC6336] Westerlund, M. and C. Perkins, "IANA Registry for Interactive Connectivity Establishment (ICE) Options", RFC 6336, DOI 10.17487/RFC6336, July 2011, <https://www.rfc-editor.org/info/rfc6336>.

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <https://www.rfc-editor.org/info/rfc6724>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

## 22.2 信息参考

[ICE-SIP-SDP] Petit-Huguenin, M., Nandakumar, S., and A. Keranen, "Session Description Protocol (SDP) Offer/Answer procedures for Interactive Connectivity Establishment (ICE)", Work in Progress, draft-ietf-mmusic-ice-sip-sdp-21, June 2018.

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/info/rfc1918>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <https://www.rfc-editor.org/info/rfc2475>.

[RFC3102] Borella, M., Lo, J., Grabelsky, D., and G. Montenegro, "Realm Specific IP: Framework", RFC 3102, DOI 10.17487/RFC3102, October 2001, <https://www.rfc-editor.org/info/rfc3102>.

[RFC3103] Borella, M., Grabelsky, D., Lo, J., and K. Taniguchi, "Realm Specific IP: Protocol Specification", RFC 3103, DOI 10.17487/RFC3103, October 2001, <https://www.rfc-editor.org/info/rfc3103>.

[RFC3235] Senie, D., "Network Address Translator (NAT)-Friendly Application Design Guidelines", RFC 3235, DOI 10.17487/RFC3235, January 2002, <https://www.rfc-editor.org/info/rfc3235>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <https://www.rfc-editor.org/info/rfc3261>.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <https://www.rfc-editor.org/info/rfc3264>.

[RFC3303] Srisuresh, P., Kuthan, J., Rosenberg, J., Molitor, A., and A. Rayhan, "Middlebox communication architecture and framework", RFC 3303, DOI 10.17487/RFC3303, August 2002, <https://www.rfc-editor.org/info/rfc3303>.

[RFC3424] Daigle, L., Ed. and IAB, "IAB Considerations for UNilateral Self-Address Fixing (UNSAF) Across Network Address Translation", RFC 3424, DOI 10.17487/RFC3424, November 2002, <https://www.rfc-editor.org/info/rfc3424>.

[RFC3489] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, DOI 10.17487/RFC3489, March 2003, <https://www.rfc-editor.org/info/rfc3489>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3605] Huitema, C., "Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)", RFC 3605, DOI 10.17487/RFC3605, October 2003, <https://www.rfc-editor.org/info/rfc3605>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC3725] Rosenberg, J., Peterson, J., Schulzrinne, H., and G. Camarillo, "Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)", BCP 85, RFC 3725, DOI 10.17487/RFC3725, April 2004, <https://www.rfc-editor.org/info/rfc3725>.

[RFC3879] Huitema, C. and B. Carpenter, "Deprecating Site Local Addresses", RFC 3879, DOI 10.17487/RFC3879, September 2004, <https://www.rfc-editor.org/info/rfc3879>.

[RFC4038] Shin, M-K., Ed., Hong, Y-G., Hagino, J., Savola, P., and E. Castro, "Application Aspects of IPv6 Transition", RFC 4038, DOI 10.17487/RFC4038, March 2005, <https://www.rfc-editor.org/info/rfc4038>.

[RFC4091] Camarillo, G. and J. Rosenberg, "The Alternative Network Address Types (ANAT) Semantics for the Session Description Protocol (SDP) Grouping Framework", RFC 4091, DOI 10.17487/RFC4091, June 2005, <https://www.rfc-editor.org/info/rfc4091>.

[RFC4092] Camarillo, G. and J. Rosenberg, "Usage of the Session Description Protocol (SDP) Alternative Network Address Types (ANAT) Semantics in the Session Initiation Protocol (SIP)", RFC 4092, DOI 10.17487/RFC4092, June 2005, <https://www.rfc-editor.org/info/rfc4092>.

[RFC4103] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, DOI 10.17487/RFC4103, June 2005, <https://www.rfc-editor.org/info/rfc4103>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC4787] Audet, F., Ed. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January 2007, <https://www.rfc-editor.org/info/rfc4787>.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, DOI 10.17487/RFC5245, April 2010, <https://www.rfc-editor.org/info/rfc5245>.

[RFC5382] Guha, S., Ed., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, DOI 10.17487/RFC5382, October 2008, <https://www.rfc-editor.org/info/rfc5382>.

[RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, DOI 10.17487/RFC5761, April 2010, <https://www.rfc-editor.org/info/rfc5761>.

[RFC6080] Petrie, D. and S. Channabasappa, Ed., "A Framework for Session Initiation Protocol User Agent Profile Delivery", RFC 6080, DOI 10.17487/RFC6080, March 2011, <https://www.rfc-editor.org/info/rfc6080>.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <https://www.rfc-editor.org/info/rfc6146>.

[RFC6147] Bagnulo, M., Sullivan, A., Matthews, P., and I. van Beijnum, "DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, DOI 10.17487/RFC6147, April 2011, <https://www.rfc-editor.org/info/rfc6147>.

[RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP's Retransmission Timer", RFC 6298, DOI 10.17487/RFC6298, June 2011, <https://www.rfc-editor.org/info/rfc6298>.

[RFC6544] Rosenberg, J., Keranen, A., Lowekamp, B., and A. Roach, "TCP Candidates with Interactive Connectivity Establishment (ICE)", RFC 6544, DOI 10.17487/RFC6544, March 2012, <https://www.rfc-editor.org/info/rfc6544>.

[RFC6928] Chu, J., Dukkipati, N., Cheng, Y., and M. Mathis, "Increasing TCP's Initial Window", RFC 6928, DOI 10.17487/RFC6928, April 2013, <https://www.rfc-editor.org/info/rfc6928>.

[RFC7050] Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", RFC 7050, DOI 10.17487/RFC7050, November 2013, <https://www.rfc-editor.org/info/rfc7050>.

[RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", RFC 7721, DOI 10.17487/RFC7721, March 2016, <https://www.rfc-editor.org/info/rfc7721>.

[RFC7825] Goldberg, J., Westerlund, M., and T. Zeng, "A Network Address Translator (NAT) Traversal Mechanism for Media Controlled by the Real-Time Streaming Protocol (RTSP)", RFC 7825, DOI 10.17487/RFC7825, December 2016, <https://www.rfc-editor.org/info/rfc7825>.

[RFC8421] Martinsen, P., Reddy, T., and P. Patil, "Interactive Connectivity Establishment (ICE) Multihomed and IPv4/IPv6 Dual-Stack Guidelines", RFC 8421, DOI 10.17487/RFC8421, July 2018, <https://www.rfc-editor.org/info/rfc8421>.

[WebRTC-IP-HANDLING] Uberti, J. and G. Shieh, "WebRTC IP Address Handling Requirements", Work in Progress, draft-ietf-rtcweb-ip-handling-09, June 2018.

# 附录A lite和完整实现

ICE允许两种类型的实现。完整实现支持会话中的控制和受控角色，还可以执行地址采集。相反，lite实现是极简的实现，除了响应STUN检查外，它仅支持会话中的受控角色。

由于ICE需要两个终结点来支持，以便为任一终结点带来好处，因此在网络中增量部署ICE更为复杂。许多会话都涉及一个终结点，它本身不在NAT后面，也不是担心NAT穿透的终结点。常见的情况是，有一个终结点需要NAT穿透（如VoIP硬电话或软电话）对这些设备之一发起呼叫。即使手机支持完整ICE实现，如果其他设备不支持ICE，ICE也无法使用。lite版实现允许为这些设备提供低成本的入口点。一旦他们支持lite实现，完整实现就可以连接到它们，并实现ICE的全部好处。

因此，lite实现仅适用于[始终]连接到公共Internet并具有公共IP地址的设备，以便接收来自任何代理的数据包。当lite实现在NAT之后时，ICE将不起作用。

ICE允许lite实现具有单个IPv4主机候选项和多个IPv6地址。在这种情况下，候选对由控制代理使用静态算法（如RFC 6724中的算法）选择，此规范建议该算法。但是，用于地址选择的静态机制总是容易出错，因为它们永远无法反映实际拓扑或提供连接的实际保证。他们总是启发式。因此，如果ICE代理实施ICE只是为了在其IPv4和IPv6地址之间进行选择，并且其IP地址均不在NAT后面，则仍建议使用完整ICE，以便提供可能最可靠的地址选择形式。

请务必注意，lite实现已添加到此规范中，为完整实现提供了一个基础。即使对于始终仅使用单个IPv4地址连接到公共Internet的设备，如果可实现，最好采用完整实现。完整实现还可以获得与NAT穿透无关的ICE的安全优势。最后，通常的情况是，一个设备发现自己今天具有公共地址，明天将放置在一个网络中，它将位于NAT之后。很难明确知道，在设备或产品的生命周期内，它是否将永远用于公共互联网。完整实现保证了通信始终发挥作用。

# 附录B 设计动机

ICE包含许多规范行为，这些规范行为本身可能很简单，但源自复杂或非显而易见的思维或用例，值得进一步讨论。由于这些设计动机对于实现目的不需要理解，因此在此处讨论这些动机。本附录是非规范的。

## B.1 STUN交易节奏

用于采集候选项和验证连接的STUN事务以大约每Ta毫秒一个新事务的速度完成。每个事务，反过来，有一个重新传输计时器RTO，这也是Ta的函数。为什么这些事务是节奏的，为什么使用这些公式？

发送这些STUN请求通常会在客户端和STUN服务器之间在NAT设备上创建绑定。经验表明，许多NAT设备创建新绑定的速率有上限。IETF ICE WG中关于此规范的讨论得出结论，每5毫秒一次得到很好的支持。这就是为什么Ta的下限为5毫秒。此外，这些数据包在网络上的传输消耗带宽，需要ICE代理的速率限制。基于早期草稿版本的[RFC5245]的部署往往超速率限制访问链路，总体执行不佳，此外还会对网络产生负面影响。因此，节奏可确保NAT设备不会过载，并且流量保持在合理的速率。

"合理"速率的定义是，一旦数据开始流动，STUN不得使用比RTP本身使用更多的带宽。Ta的公式旨在使STUN数据包每Ta秒发送一次，使得在所有数据流中，消耗与RTP数据包相同的带宽量。当然，STUN可能重新传输，并且希望保持节奏。因此，RTO的设置使第一个事务上的第一次重新传输与上次事务上的第一个STUN请求发生一样。如图：



在此图中，将发送三个事务（例如，在候选项采集中，有三个主机候选项/STUN服务器对），即事务A、B和C。将重新传输计时器设置为在第一个事务（数据包A2）上发送第一个重新传输的时间为3Ta。

第一次后，后续重新传输的频率甚至比Ta毫秒的频率还要低，因为STUN在其重新传输时使用指数回退。

这种全局最小节奏间隔为5 ms的机制通常不适用于传输协议，但基于以下推理适用于ICE。

* 从以下通常适用于传输协议的规则开始：
  1. 令MaxBytes为启动时允许网络对外发送的最大字节数，按照[RFC6928]第2节中的定义，该值应为14600。
  2. 令HTO为事务超时，如果RTT已知，则该超时为2\*RTT，否则应为500毫秒。这基于[RFC5389]中STUN消息的RTO和TCP初始RTO，在[RFC6298]中为1秒。
  3. 令MinPacing为事务之间的最小节奏间隔，即5毫秒（见上文）。
* 代理通常不知道ICE事务的RTT（特别是连接检查），这意味着HTO几乎总是为500毫秒。
* 5 ms的MinPacS和500 ms的HTO最多提供100个数据包/HTO，对于小于120字节的典型ICE检查，意味着网络中最多12000个对外发送字节，小于规则1定义的最大字节。
* 因此，对于ICE，规则集减少到仅MinPacing规则，相当于全局Ta值。

## B.2 具有多个Base的候选项

第5.1.3节谈到取消具有相同传输地址和Base的候选项。但是，具有相同传输地址但不同Base的候选项不是冗余的。ICE代理何时可以具有两个相同IP地址和端口但Base不同的候选项？考虑图11的拓扑：



图 11：具有不同基的相同候选项

在这种情况下，启动代理是多宿主的。它在网络C上有一个IP地址10.0.1.100，该网络是一个net 10私有网络。响应代理位于同一网络上。启动代理还连接到网络A，该网络为192.168/16，IP地址为192.168.1.100。此网络上有一个NAT，接到网络B，这是另一个net 10私有网络，但它未连接到网络C。网络B上有一个STUN服务器。

启动代理在网络C（10.0.1.100：2498）上的IP地址获取主机候选项，并在网络A上的IP地址（192.168.1.1.100：3344）上获取主机候选项。 它从192.168.1.100：3344对其配置的STUN服务器执行STUN查询。此查询通过NAT，NAT恰好分配绑定10.0.1.100：2498。STUN服务器在STUN绑定响应中反映了这一点。现在，启动代理获得了一个服务器反射候选项，其传输地址与主机候选项相同（10.0.1.100：2498）。但是，服务器反射候选项的Base为192.168.1.100：3344，并且主机候选项的Base为10.0.1.100：2498。

## B.3 相关地址和相关端口属性的目的

候选项属性包含ICE本身根本不使用的两个值—相关地址和相关端口。为什么需要它们？

列入它们有两个动机。第一是诊断。了解不同类型的候选项之间的关系非常有用。通过包括它们，ICE代理可以知道哪个中继候选项与哪个反射候选项相关联，而哪个候选项又与特定的主机候选项相关联。当检查一个候选项成功但对其他候选项不成功时，这提供了对网络中发生的情况的有用诊断。

第二个原因与非路径服务质量（QoS）机制有关。当ICE在PacketCable 2.0 等环境中使用时，代理除了执行正常的SIP操作外，还将检查SIP消息中的SDP并为数据流提取IP地址和端口。然后，他们可以通过策略服务器与网络中的访问路由器进行交互，为数据流建立有保障的QoS。此QoS基于5元组对RTP流量进行分类，然后提供保证速率，或适当地标记其DSCP来提供。当存在家用NAT并且为数据选择中继候选项时，此中继候选项将成为实际TURN服务器上的传输地址。该地址对数据包分类以进行QoS处理的访问路由器中的实际传输地址一无所知。 相反，需要面向TURN服务器的服务器反射候选项。通过在SDP中携带翻译，代理可以使用该传输地址向访问路由器请求QoS。

## B.4 STUN用户名的重要性

ICE要求使用基于STUN短期凭据功能的消息完整性。实际短期凭据在候选项交换中通过交换用户名片段而形成。建立这一机制的必要性不仅仅是安全；它实际上将正确操作ICE的需要摆在首位。

考虑ICE代理L、R和Z。L和R在私有企业网1中，使用10.0.0.0/8。Z属于私有企业网2，该企业网也使用10.0.0.0/8。事实证明，R和Z都具有IP地址10.0.1.1。L将候选项发送到Z。Z响应L及其主机候选项。在这种情况下，这些候选项为10.0.1.1：8866和10.0.1.1：8877。事实证明，R同时处于会话中，并且也使用10.0.1.1：8866和10.0.1.1：8877作为主机候选项。这意味着R准备接受这些端口上的STUN消息，就像Z一样。L将向10.0.1.1：8866发送STUN请求，将另一个请求发送到10.0.1.1：8877。但是，这些不会按预期转到Z。相反，他们去R！如果R只是回复了它们，L会相信它与Z有连接，而实际上它与完全不同的用户R具有连接。为了解决此问题，可以使用STUN短期凭据机制。用户名片段足够随机；因此，R使用与Z相同值的可能性极小。因此，R将拒绝STUN请求，因为凭据无效。从本质上讲，STUN用户名片段提供某种形式的动态主机标识，绑定到作为候选项交换的一部分建立特定会话。

IP地址不唯一的一个不幸后果是，在上面的示例中，R甚至可能不是ICE代理。它可以是任何主机，并且STUN数据包定向到的端口可以是该主机上的任何临时端口。如果应用程序在此套接字上侦听数据包，并且它不准备为正在使用的任何协议处理格式错误的数据包，则该应用程序的操作可能会受到影响。幸运的是，由于交换的端口是短暂的，通常来自动态或注册范围，因此端口不用于在主机R上运行服务器的可能性很大，而是某些协议的代理端。由于此范围内端口使用的暂时性，这降低了命中分配端口的可能性。但是，确实有可能存在问题，网络部署人员需要为此做好准备。请注意，这不是ICE特有的问题。流散数据包可以随时到达任何类型的协议，尤其是公共Internet上的协议到达端口。因此，这只是重申Internet应用程序的一般设计准则—为任何端口上的未知数据包做好准备。

## B.5 候选对优先级公式

候选对的优先级具有如下奇怪的形式：

为什么会这样？当候选对基于此值排序时，生成的排序具有极大/极小属性。这意味着首先根据两个优先级的最小值的减小值对这些对进行排序。对于最小优先级相同的对，使用最大优先级对它们进行排序。如果最大值和最小优先级相同，则控制代理的优先级将用作表达式最后一部分中的配合破坏者。使用因子，因为单个候选项的优先级始终小于，因此对优先级是两个组件优先级的"串联"。这将创建极大/极小排序。极大/极小确保对于特定ICE代理，在尝试所有高优先级候选项之前，绝不会使用优先级较低的候选项。

## B.6 为什么需要保活？

一旦数据开始在候选对上流动，仍有必要在会话期间保持绑定在中间NAT上处于活动状态。通常，数据包本身（例如RTP）满足此目标。然而，某些场景值得进一步讨论。首先，在某些RTP用法（如SIP）中，数据流可以"置于保留状态"。 这是通过使用SDP"仅发送"或"非活动"属性实现的，如RFC 3264中定义的。RFC 3264指示实现在这些情况下停止数据传输。但是，这样做可能会导致NAT绑定超时，并且数据将无法关闭保留状态。

其次，某些RTP有效负载格式（如文本对话的有效负载格式[RFC4103]）可能会发送数据包的频率很少，以至于间隔超过NAT绑定超时。

第三，如果使用静默抑制，长时间的沉默可能会导致数据传输停止足够长的时间，使NAT绑定超时。

由于这些原因，不能依赖数据包本身。ICE利用STUN绑定指示定义一个简单的定期保持活动状态。这使得其带宽要求高度可预测，因此可适应QoS预期。

## B.7 为什么更偏好对等反射候选项？

第 5.1.2 节介绍了根据候选项的类型和本地偏好计算候选项优先级的过程。该部分要求对等反射候选项的类型偏好始终高于服务器反射候选项。为什么？原因与第19节中的安全注意事项有关。攻击者更容易使ICE代理使用错误的服务器反射候选项，而不是虚假的对等反射候选项。因此，ICE更偏好对等反射候选项，以此克服针对绑定请求采集地址的攻击。

## B.8 为什么绑定指示用于保活？

数据保活在第11节作了说明。当两个终结点都支持ICE时，这些保活使用STUN。 但是，保活使用指示，而不是使用绑定请求事务（生成响应）。为什么？

主要原因与网络QoS有关。一旦数据开始流动，网络元素将假定数据流具有相当常规的结构，以固定间隔使用定期数据包，并有可能抖动。如果ICE代理正在发送数据包，然后收到绑定请求，则需要生成响应数据包及正常数据包。这将增加携带数据包的5元组的实际带宽要求，并在这些数据包的传递中引入抖动。分析表明，在某些第2层访问网络中，这是一个问题，这些网络对数据使用相当严格的数据包调度程序。

此外，使用绑定指示允许禁用完整性，这可能导致更好的性能。对于大型终结点很有用，例如公共交换电话网络（PSTN）网关和会话边界控制器（SBC）。

## B.9 选择候选项类型偏好

选择类型和位置偏好的一个标准是使用数据中介，如TURN服务器、隧道服务（如VPN服务器）或NAT。对于数据中介，如果数据发送到该候选项，它将在接收之前首先传输数据中介。涉及数据中介的候选项类型之一是中继候选项。另一种类型是从VPN接口获取的主机候选项。当数据通过数据中介传输时，它可以对传输和接收之间的延迟产生正或负面的影响。它可能增加，也可能不增加数据包丢失，因为可能采取额外的路由器跃点。它可能会增加服务的成本，因为数据将从提供程序运行的数据中介中路由，然后直接返回。如果这些关切很重要，则需要仔细选择转接候选项的类型偏好。

选择偏好的另一个标准是IP地址族。ICE适用于IPv4和IPv6。它提供了一种过渡机制，允许双栈主机比IPv6首选连接，但在v6网络断开连接时可回退到IPv4。实现应遵循[RFC8421]的准则，以避免存在断开的路径时连接检查阶段延迟过长。

选择偏好的另一个标准是拓扑感知。这对利用中间人的候选项是有好处的。在这些情况下，如果ICE代理预先配置或动态发现有关中介与自身拓扑邻近性的知识，则可以使用该知识为从更密切的中介获得的候选项分配更高的位置偏好。

选择首选项的另一个标准可能是安全性或隐私。如果用户是远程办公人员，需要连接到公司网络和本地家庭网络，则用户可能希望其语音流量通过VPN或类似隧道进行路由，以便在企业内部通信时将其保留在公司网络上，但在与企业外部用户通信时可以使用本地网络。在这种情况下，VPN地址的位置偏好将高于任何其他地址。

# 附录C 连接检查带宽消耗

下表显示了IPv4和IPv6使用不同的Ta值（以ms为单位）和不同的ufrag大小（以字节为单位）执行连接检查所需的带宽。

结果由尤辛·尤伯蒂（谷歌）于2016年4月11日提供。



图 12：连接检查带宽