

eMule 协议说明书

1 简介

1.1 目的

eMule 是一个如今非常流行的基于eDonkey协议的文件共享程序。 本文描述了eMule的网络行为并且解释了一些我们理解这个协议所需要的基本学术用语。 并且本文也是eMule网络协议的一个详尽的说明书，内容包括传送的消息的结构， 本文中所有的信息是基于一个开源的eMule的客户端[2]。下文的目的是让读者有一个整体的概念去阅读和理解这些文档，一个关于eMule的巨大信息[3]。

1.2 概述

eMule网络是由数百台eMule服务器和数百万的eMule客户端组成的[1]。为了进行网络通信，客户端将连接到一个服务器。只要这个客户端还在这个系统当中，这个服务器连接就一直存在。这些服务器集中执行着索引服务(就像在Napster中一样)并且不与其它的服务器进行交流。每一个eMule客户端都在其本地文件系统中预先配置了一个服务器列表和共享文件列表文件。客户端只使用TCP来连接服务端来加入网络，获得想要的信息和可以连接的客户。同时eMule的客户端也使用成百的TCP连接到其它的客户端来上传和下载文件。每一个eMule客户端为它的每一个共享文件维持一个上传队列。下载者刚加入队列的时候在队列的底部并且逐渐的前进，直到他们到达顶部才开始下载他们需要的文件。一个客户端可以从其它的几个客户端那里同时下载，取得他们所拥有的不同的文件片段。客户端同时也上传那些还没完全下载的文件。 最重要的是eMule对eDonkey 进行了扩充，允许客户端之间交换服务器信息，包括其它的客户端和文件信息。注意客户端与服务端的通信都是以TCP协议为基础的。服务器使用本身的数据库来存储用户和文件信息。一个eMule服务器不保存任何文件，它充当了文件位置索引表的功能。服务器的另外一个值得争议功能是透过防火墙连接两个客户端，并且不能够引入连接。桥连接函数急剧的增加了服务器的负载。eMule在客户端和服务端都使用UDP协议来提高客户的接受能力。客户端接收和发送UDP数据的能力不取决于客户的正确操作并且即使防火墙阻止发送和接受UDP数据也可以无差错的工作。

1.2.1 客户到服务器的连接

在启动客户端之前先使用TCP协议连接到一个eMule服务器。服务器提供给客户端一个客户端ID (1.3章节)这个ID在整个客户端服务端连接的生命周期都有效 (注意:拥有high ID的客户端在IP地址改变之前将从所有的服务端获得同等级的ID)。连接确立之后客户将自己的共享文件列表发送给服务器，服务器将这些列表存在它本身的数据库中。

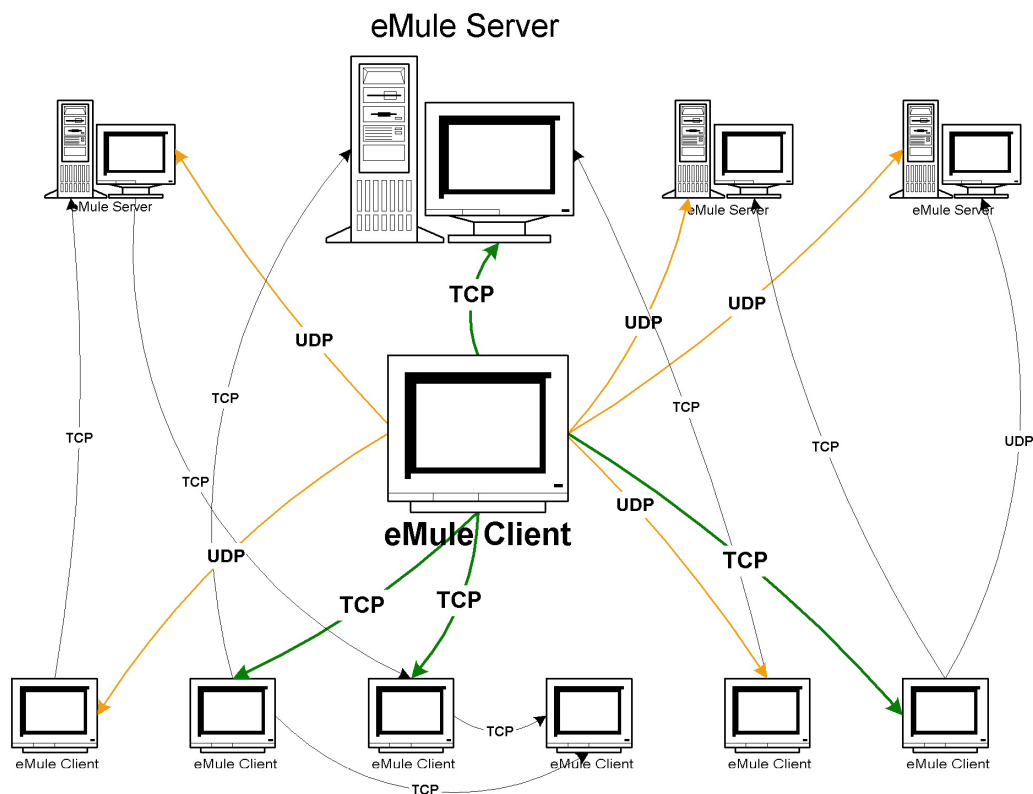


Figure 1.1: eMule high level network diagram

通常这些数据库包含了成百上千的共享文件清单和有效的客户端。eMule客户端同时也下载它想获得的文件的文件列表。第二章详细的描述了eMule的客户端和服务端的TCP信息交换。在连接确立之后，eMule服务器将发送给这个客户端一份客户端列表，这个列表中的客户端中有下载者想下载的文件（这些客户端叫做“源”）。从此刻开始，eMule客户端像下面的1.2.2章节中描述的一样开始建立和他客户端的连接。

注意客户端/服务端的TCP连接在整个客户端会话过程中持续开放。在初次握手协议之后的就主要是用户活动：客户端为了获得新的搜索结果不时地发送搜索请求，一个搜索通常紧跟着是一个对于特殊文件资源的查询，回复查询的清单是由能下载到该文件的资源(IP和端口)组成。UDP是用来与服务器端之间的信息交换而不是服务器端与它相连的客户端之间交流。UDP消息的目的是增加文件搜索，增加资源搜索，最终保持活跃性(确保客户端中的服务器列表中的服务器都是有效的)。您可以在第3章节中找到更多关于客户端-服务端UDP信息交换的详细资料。

1.2.2 客户端和客户端的连接

一个eMule客户端为了下载文件而连接到其他的eMule客户端(一个资源)。文件被分割为几个部分每个部分又是由更多的片段组成。一个客户端可以从许多(不同的)客户端下载同一个文件的不同的片断。当两个客户端连接时他们交换接受能力信息，然后商议开始下载(或者上传,取决于需求)。每一个客户端拥有一个下载队列来保存等待下载文件的客户端。当一个eMule客户端清空下载队列，通常是由于要开始一个下载(例外,例如：这个请求着被禁止)。当下载队列不清空时将导致队列中请求客户的增加。在给定的时间内即便是每个客户用最小的带宽2.4kb/s也只能服务极少数的客户。如果在15分钟的下载过程队列中有比正在下载的用户更高级的用户，该用

户将抢占下载中的用户进行下载，这样做来防止长期等待。当一个下载中的客户端到达下载队列的最顶端,上传客户端就会发起一个连接来传输下载端需要的部分。一个eMule客户端或许同时许许多多其它的客户端的等待队列中，去下载一个文件的同一部分。当等待中的客户端实际上已经下载完成了该部分(从他们其中的一个)它并不通知其他剩下的所有客户端可以从等待队列中将其清除，她仅仅是在他到达等待队列的顶端是时拒绝那些客户端的上传请求。

eMule采用了一个荣誉系统(看1.4章节)来鼓励上传，采用RSA公开密钥加密算法来保护eMule荣誉系统的安全性。客户的连接使用的是eDonkey协议中没定义的一组消息，这些消息被叫做扩展协议。这个扩展协议用来实现荣誉系统，用来说明信息交换(例如更新服务器和资源列表)和提高传送和接受压缩数据块的性能。eMule客户端仅在他要开始下载文件时使用UDP去检查他的上传队列中的每一个客户端的状态。

1.3 客户ID

客户端在与服务端进行握手连接时，服务端给客户端一个4字节的标示符作为客户端 ID。一个客户端的有效ID仅在客户-服务器的TCP连接中获得，有时该客户端可能是High ID 但是直到它的IP地址改变之前对于所有的服务器他将赋予同一个ID。客户端ID被分为low IDs和high IDs。当客户端不能接受连入连接请求时，eMule服务端将把它们标示为low ID。拥有low ID的用户将被限制使用eMule网络并且可能导致服务器拒绝客户端的连接。像下面描述的一样，一个high ID的计算以这个客户端的IP地址为基础。本章从eMule协议的角度上来描述客户ID的任务和意义[3]。一个被给与high ID的客户端允许其他的客户端自由的连接到他主机的eMule的TCP端口(默认的端口号是4662)。拥有high ID的客户端可以无限制的使用eMule网络。当服务端不能够与客户端的eMule端口建立TCP连接，该客户端奖杯给予low ID。只主要是由于客户端在他们的机器上设立的防火墙阻止引入的连接。由于下面的情况客户端同样也有可能获得low ID:

- 当客户端是通过NAT或者代理服务器连接的;
- 当服务端过于繁忙(导致服务端的重接计时器到时间);

High ID是通过下面的方法计算的:

假设主机的IP是 $X.Y.Z.W$ 这个ID将是 $X + 2^8 * Y + 2^{16} * Z + 2^{24} * W$ ('big-endian' 表示法)。一个low ID总是小于16777216 (0x1000000)，我找不到过于他如何计算的线索，注意不同的服务器中你拥有的low ID是不同的。一个low ID没有允许其他客户端客户连入的公共IP，所以所有的信息必须通过eMule服务器。这增加了服务器的计算负担并且直接导致了服务器不愿意接受low ID客户。同样这也意味着不同服务器之间的low ID用户之间不能够建立连接，因为eMule不支持服务器之间的请求通道。为了支持low ID，用户复查机制被引入了。通过这个机制一个high ID客户可以邀请(通过eMule服务器)low ID用户和它建立连接来交换文件。

1.4 用户 ID

eMule支持一个荣誉系统来鼓励使用者共享文件。用户上传给其他用户的文件越多,他就能获得更多的荣誉同时它们在等待队列中前进的就越快[3]。用户ID是由连接随机数创造的一个128位(16字节)的GUID，第6字节和第15

字节不是随即产生的，它们的值分别是14和111。当客户端与特殊的服务端绘画取得有效的客户ID时用户ID(也叫做用户哈希数)是唯一的并且通过会话识别客户(用户ID识别工作站)。用户ID在荣誉系统中扮演了重要角色，这也为'hackers'模仿其他的用户利用它们的荣誉度获得特权提供了动机。eMule支持一种加密方案来阻止欺骗和用户扮演。执行一个简单的依靠RSA公开/私有密钥加密算法的挑战相应交换。

1.5 文件 ID

文件ID即使用网络中文件的识别也用在文件的正确性检测和修复上面。注意eMule不是依靠文件名来唯一的识别和记录一个文件，一个文件通过对其进行哈希算法而得出来的全球唯一的ID来标示。有两种文件ID-第一种主要用来产生唯一的文件ID，第二中用来进行正确性检测和修复[3]。

1.5.1 文件哈希

文件通过根据其内容计算出来的唯一的的一个128位GUID哈希数来标示出来。这个GUID是根据文件内容适用MD4运算规则[4]计算出来的。当计算文件ID时该文件被分割为许多份每一分大小为9.28MB。每一部分独立的算出一个GUID,然后所有的哈希数载结合起来组成一个文件ID。当一个下载客户下载文件的一部分时特就会计算该部份文件的哈希数并且将它与源文件的哈希数进行比较，如果这部份文件有错误，客户端将尝试着修复错误处，具体做法是每次替代错误处一定位数(每次180kb)的数据直到计算出来的哈希数是正确的。

1.5.2 根哈希

根哈希是通过SHA1运算法则来计算文件的每一部分，每一部分的大小是180kb。它提供了一个极高的可靠性和错误修复能力，详细的细节在eMule的官方网页上。

1.6 eMule 协议的扩展

尽管eMule完全的兼容了eDonkey，但是他还实现了几个扩展来允许两个客户端之间向他们的使用提供附加的功能。这些扩展主要用于客户端与客户端之间的信息交换尤其是安全领域和UDP协议的利用。在本文当中所有eMule扩展部分的流程图制定消息都是灰色的。

1.7 软界限和硬界限

服务器的配置包含两种界限，这取决于使用中用户的数目-软界限和硬界限。硬界限比软界限提供更大的平衡。当使用中的用户的数目达到软界限时服务器停止接受新的low ID用户的连接，当使用者数目达到硬界限的时候服务器就满了并且不再接受任何连接。

2 客户服务器TCP信息

每一个客户端使用TCP连接连接到唯一一个指定的服务器。这个服务器将分配给这个客户端一个ID，这个ID将在该客户与其他服务器会话的过程中唯一的标示自己(一个high ID用户总是依据其IP地址进行分配的)。为了使一个eMule图形用户界面客户端运作起来，首先要和服务器建立一个连接。客户端既不能同时连接多个客户端，也不能在没有用户的干涉下，自动改变客户端的连接。

2.1 连接的建立

在建立与服务器的连接时客户端尝试着向许多平行服务器建立连接，成功登陆序列之外的将全部被放弃。

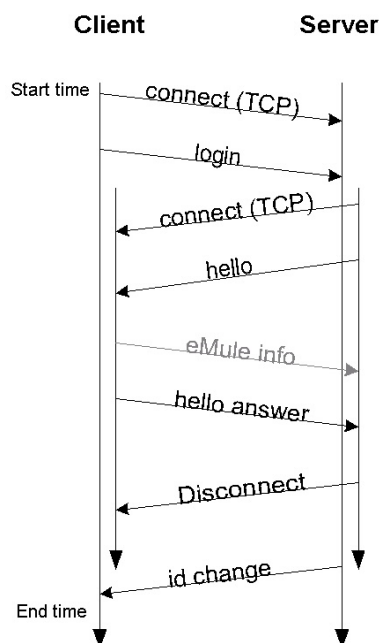


图 2.1: High ID 登录序列

有许多可能成功建立连接的情况：

1. High ID连接 - 服务器分配给连入的客户端一个 high ID
2. Low ID连接 - 服务器分配给连入客户端一个 low ID
3. 拒绝会话 - 服务器拒绝该客户端

当然还有少数情况例如：服务器崩溃了或者服务器是不可到达的。图2.1描述了建立一个high ID连接的消息顺序。在这种情况下客户端先和服务器建立一个TCP连接然后向服务器发送登入请求消息。服务器使用另外一个TCP连接连接客户端，这个连接扮演了一个client-to-client的握手过程，通过这个连接服务器来确定该客户端是否有接受其他客户端连入连接的能力。完全完成客户端握手之后服务器关闭第二个连接并且向客户端发送一个ID变换消息然后结束客户-服务器的握手会话。你或许注意到了eMule-info消息是灰色的。这是因为这个消息是eMule扩展协议(1.6章节)的一部分。

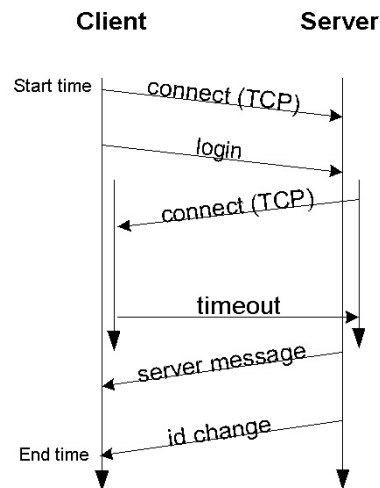


图 2.2: Low ID登录序列

图2.2描述了建立一个low ID连接的消息顺序。这种情况发生在服务端尝试与客户端建立连接失败并分配给该客户端low ID。这样的服务器消息通常包含警告信息,例如:”Warning [server details] - You have a lowid. Please review your network config and/or your settings.” Low和high ID握手都是以ID变更消息作为结束的,这个消息分配了该客户端今后它与其他服务器会话时所使用的客户ID。

图2.3描述了拒绝连接的消息顺序。由于客户端拥有的是low ID或者当服务器达到了他们的硬界限时它们会拒绝客户端的连接。服务器消息将报还关于拒绝原因的简单描述。

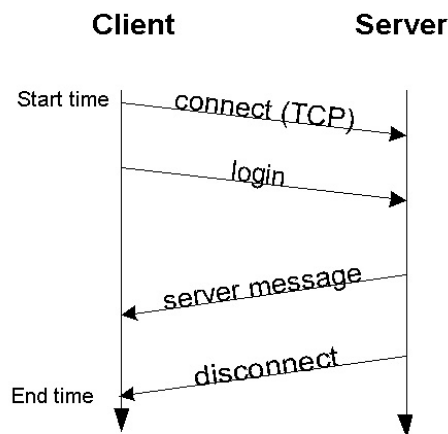


图2.3: 拒绝会话序列

2.2连接启动信息交换

在成功建立连接之后,客户与服务器交换几个设置消息。这些消息是为了更新与他们同层的peer的状态。开始客户端现象服务端提过一个共享文件信息列表(见6.2.4章节),然后他请求更新它的服务器列表。服务器发送有关自己的版本和状态信息(6.2.6和6.2.2章节),然后发送一个已知eMule服务器列表,并提供更多一些有关自我识别的细节。最后客户端请求资源(在服务器下在列表中能下载到该文件的客户端),并且服务端回复一连串消息,每一个文件提供一个客户端的下载列表,直到客户端下载完成了所有的资源列表。

图2.4将说明这个顺序。

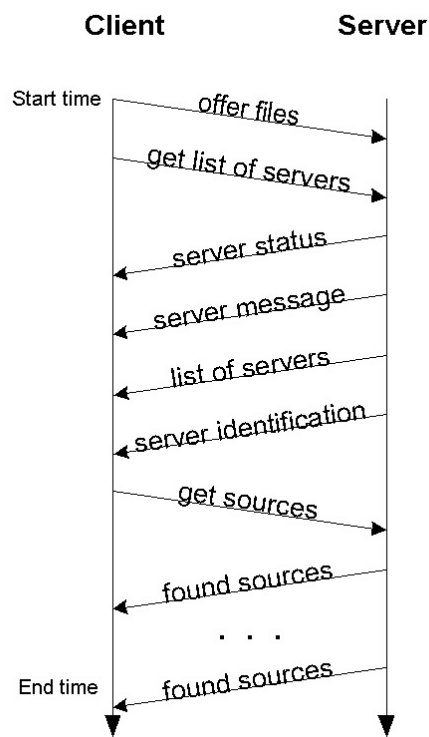


图2.4: 连接启动消息序列

2.3文件检索

文件检索是由用户开始的。操作很简单，向服务器发送搜索请求(见6.2.9章节)，并得到搜索结果(6.2.10章节)。当结果很多时，搜索结果将被压缩。下一步，用户将选择一个或者多个他要下载的文件，然后客户端请求选择的文件资源，并且服务器将为每一个请求的文件提供一个资源列表(6.2.12章节)。在建立资源答复消息之前，有一个可选的服务器状态信息。这个状态信息(6.2.6章节)包含了当前用户数和服务器所支持的文件。值得着重注意的是这里有一个额外的UDP消息序列是用来增强客户对端搜索列表种资源的定位能力的。可以从第3章节中寻找更多的消息。在确定资源都是最新的之后，eMule的客户端开始试图建立连接并且增加它们的资源列表。客户端的资源连接的顺序解释他们从服务器端所接收到的顺序。图2.5描述了文件检索的消息顺序。

eMule客户端按照他们增加到他们列表中的顺序去连接资源。eMule没有决定哪个资源先连接的优先权机制。eMule有一个复杂的机制来向客户端说明在他的下载列表中的哪一个资源是可以被请求的(注意eMule在两个客户端之间仅允许一条上传连接)。

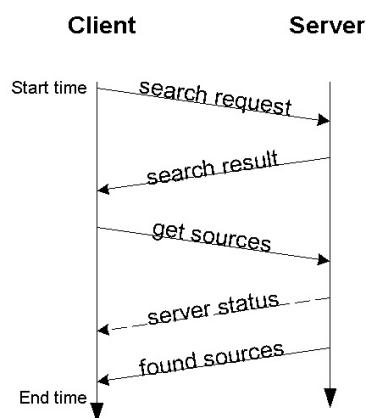


图 2.5: 文件检索消息序列

这个选择算法是基于我们的优先权机制，当没有指定优先权时磨人的规则是按照字母的顺序。让一个资源可以上传多个文件的详细描述在eMule的网页上。

2.4 复查机制

复查机制的引入是为了克服low ID用户不能够接受引入的连接因此不能与其它的客户分享它们的文件。这个机制很简单：比如A,B连个客户端连接到同一个eMule服务器，A需要一个B所拥有的文件但是B是一个low ID，A可以向服务器提交一个复查请求(见6.2.13章节)，请求服务器让B来主动请求A。与B已将建立一个开放的TCP连接的服务器将向B发送一个复查请求消息(6.2.14章节),向B提供A的IP地址和端口号。B就可以不借助服务器向A发送文件了。很明显的，只有拥有high ID的客户端可以向拥有low ID的客户端请求复查(一个low ID的客户端不能够接受连入连接请求)。图2.6将描述复查机制的消息交换。依靠服务器作为中介我们也可以进一步来实现两个low ID的用户之间的文件交换。但是由于大多数服务器的负载能力有限，他们不再支持这项功能。

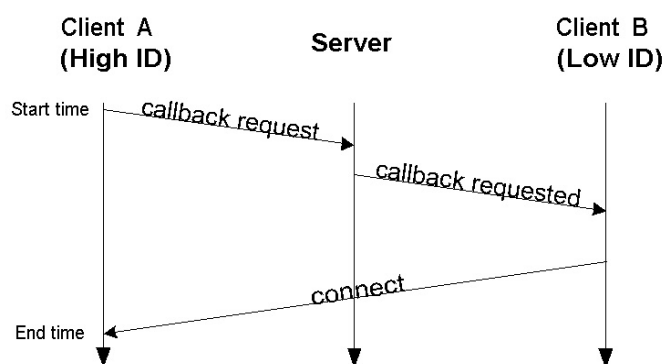


图 2.6:复查消息序列

3 客户端服务端UDP信息

eMule客户端和服务端采用不可靠的UDP服务来保持联系和增进搜索。产生的UDP封包(注意,封包不识字节)的可以达到eMule客户端所产生封包总数量的5%-这取决于客户端服务器列表中服务器的数量，客户端下载列表

中每一个要下载文件的资源数量和客户搜索查询的次数。有一个100毫秒为周期的计时器来触发UDP封包，并且有单独一个线程来处理UDP相关的事情，因此UDP封包的数目可以达到最多每秒10个。

3.1 服务器持续运行和状态信息

客户不时地改变他服务清单上的服务器状态。这种改变是通过使用UDP服务器状态请求(见6.3.3章节)和UDP服务器描述请求(见6.3.7章节)来完成的。这里描述的简单服务器持续运行计划每小时不会产生太多的数据包，这些数据包无论如何也不会超过0.2个/秒（每5秒一个数据包）。客户在检查服务器的时候首先会发送一个服务器状态请求信息并且在每两次的服务器描述尝试中需要像特征3.1中例证的请求。

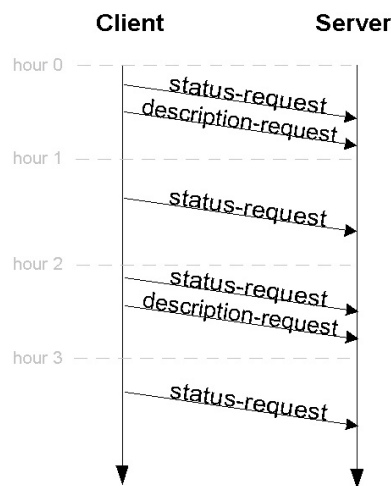


图3.1: UDP 周期循环

客户端发送的服务器状态请求包含了服务器回复中回应的随机数。在服务器回应的数字与客户端发送的口令不一致时，在回复当中的信息将被丢弃。每有一个状态请求数据包发送至服务器，客户端都会预先准备一个attempt-counter。任何来自服务器的信息(包括搜索结果等等)都将重置这个 attempt-counter。当它达到一个可控制的极限，服务器就被认为是死亡的并从客户端的服务器清单中清除。 服务端回执的几个数据项目：

- ✓ 服务器状态回执(6.3.4章节)包括用户当前的数字和服务器中的文件还有服务器软硬件的极限限制(1.7章节)。
- ✓ 服务器描述回执(6.3.8章节)包括服务器的名字和一个短的描述字符串。

特征3.2例证了在客户端和活动服务端中完整的 keep-alive序列中的信息流。

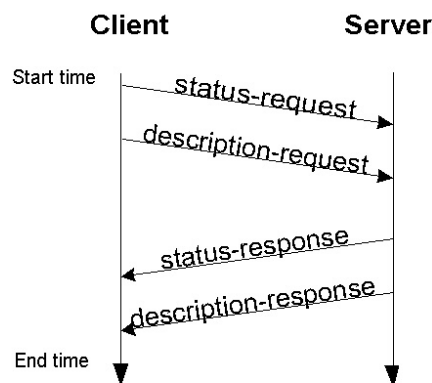


图3.2: UDP 保持运行序列

3.2 增强文件搜索

eMule可以使用UDP来增强它的文件搜索能力。UDP搜索请求的格式基本上和TCP搜索请求的格式一样。服务器只在有搜索结果时才做出回应。UDP搜索信息有两种不同的opcodes，我找不出它们之间的不同。UDP搜索分包被发送到客户端服务器列表中的服务器中去。更多的信息请看6.3.5 和 6.3.6章节。

3.3 增强文件资源搜索

当一个客户端的下载列表中的某一个文件的资源数少于一个配置好的下限(100)时，客户端将会周期性的向服务器列表中服务器发送UDP封包去获得该文件的更多的资源。每秒钟都要发送封包，因此这些封包占客户端产生的封包中相当可观的一部分。消息的格式(在6.3.1章节中描述)与TCP counter部分的格式很相似。注意与TCP资源搜索不懂得是UDP资源搜索与文件搜索无关，它只取决于一个给定文件所拥有的资源数。

4 客户端到客户端的TCP信息

客户端在完成他在服务端的注册和查询它所要的文件和资源后，它将试着与其它的客户端进行连接来下载文件。我们将为每一对[文件,客户]建立一个单独的TCP连接。当在一个确定的时间内(默认是40秒)没有活跃的socket或者peer主动断开连接时连接将被终止。为了提供满意的下载速度，在能够向下载客户端提供最小的允许速度(硬性规定是2.4kb/s)之前eMule不会让客户端开始下载。

4.1 初次握手

初次握手的时候双方向对方发送同样的消息。两个客户端之间交换彼此的标示，版本和接受能力信息。这里有两种消息-欢迎消息(6.4.1章节)和eMule信息消息(6.5.1章节)，第一个是eDonkey的一部分和eDonkey客户端该消息时一样的，第二个是客户端扩展协议的一部分只针对eMule。图4.1描述了两个eMule客户端之间的握手。这之中包含了一些额外的信息如：UDP消息交换，安全识别和资源交换能力。

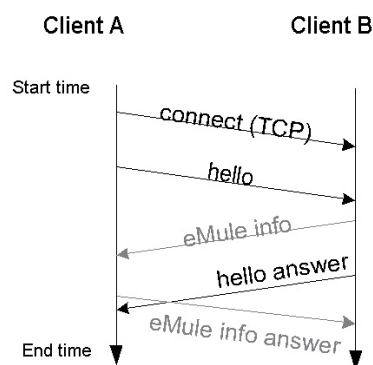


图4.1: eMule 客户端初次握手

4.2 安全用户识别

在1.4章节中我们简单的介绍了用户ID和用户扮演其他用户的可能性[3]。安全用户识别是eMule扩展协议的一部分。在初次握手完成后就会进行用户安全识别。如果要使用安全用户识别我们通过下面几个步骤来实现：

1. 在初次握手时，B提出它支持并且想使用安全用户识别。
2. A回复安全识别消息(6.5.8章节)该消息指明了A是否需要B的公钥并且包括一个已经由B确定的4字节的标记。
3. 如果A指出需要B的公钥那么B将它的公钥发送给A(6.5.9章节)。
4. B发送一个由标记产生的签名消息(6.5.10章节)额外还有一个双字节在B是low ID的情况下是A的IP地址，在B是high ID的情况下是B的ID号。图4.2将描述这个序列。

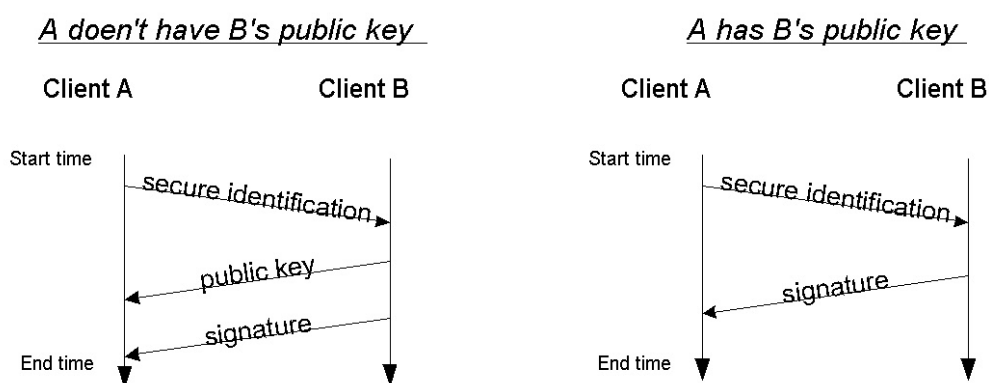


图4.2: 用户安全识别过程

4.2.1 荣誉系统

这一章节简单的描述了客户端的荣誉系统。这个荣誉系统的目的是鼓励用户们共享文件。在客户端向它的peer上传文件的时候，正在下载的客户端根据下载数据的总量来更新它的荣誉度。注意荣誉系统不是全球性的-荣誉度被存储在下载中的客户端的本地磁盘并且仅当上传客户端(获得荣誉值)想要从明确的客户端下载文件时才会被转化为数值。荣誉值取决于下面几种情况的最小值：

1. 上传总数*2/下载总数

当下载总数是0的时候表达式的值为10

2. $\sqrt{\text{上传总数}+2}$

当上传的总数小于1MB时表达式的值为1

上传/下载的数量以兆字节为单位。任何情况下荣誉度一定不能大于10或者小于1。

4.3文件请求

就像前面提及到的，每一对[客户端,文件]都建立一个单独的连接。连接建立之后，客户端立即发送一些关于他想下载的文件查询信息。图4.3描述了一个典型的成功过程。



图4.3: 文件请求

4.3.1 基本信息交换

四个消息组成了基本的信息交换：A在发送一个文件请求信息(6.4.18章节)之后立即发送一个请求文件ID信息(6.4.17章节)。B回复这个请求一个文件请求应答(6.4.15章节)和一个根据文件ID信息的文件状态信息(6.4.18章节)。我找不到任何理由将发送的信息分为四个消息，我可以通过简单的两个消息来实现(一个请求消息和一个回复消息)。扩展协议在资源请求(6.5.6章节)中增加了两个消息和一个资源应答(6.5.7章节)。这些扩展使用来将B的资源(在B正在下载文件的时候)传送给A。B没有必要在完全下载完成部分文件之后才将它传送给其他的客户，B可以传送给A他完全下载任何一部分即使只是文件的一个很小的的片断。。

4.3.2 没有找到文件时的情况

当A向B请求一个文件但是B的共享文件列表中没有这个文件。在接收到请求文件ID消息后B将忽略这个文件请求信息并回复一个文件无法找到消息(6.4.16章节)，就像图4.4中描述的一样。

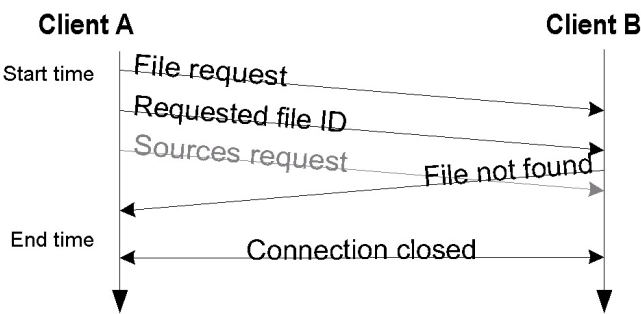


图 4.4: 文件请求失败-文件没有找到

4.3.3 争取到上传队列

优势B被请求上传文件但是他的上传队列这时并不为空,这就意味着有客户正在下载文件同时上传队列中还有等待中的客户。A与B就像图4.3中描述的那样建立一个完整的握手,但是当A向B请求下载文件的时候,B将A加入到他的上传队列中然后恢复一个包含A在B的下载队列中的位置的队列消息。图4.5将详细说明这个过程。

4.3.4 上传队列管理

客户端对于每一个上传的文件都有一个上传优先权队列。队列中的每一个优先权是基于队列中等待的时间和一个优先权参数计算出来了。在队列头部的客户有有罪该的分数。这个得分的计算公式如下: $\text{得分} = (\text{优先级} * \text{在队列中等待的时间}) / 100$ 或者在下载客户端是好友时这个结果时1。初始的优先级数值是100,当又不被禁止的时候优先级数值为0(为了防止被禁止客户到达队列的顶端)。这个优先级数值将会通过下载用户的荣誉值

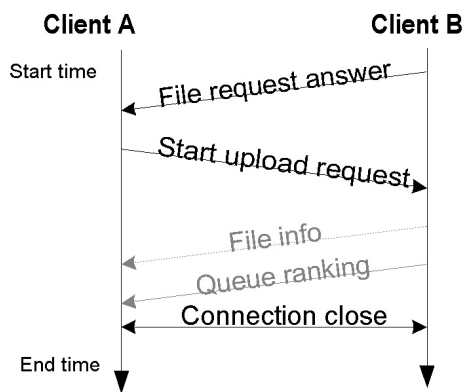


图4.5: 文件请求等待队列

(大小是1-10)和上传客户端设置的上传冲突设定得上传文件优先级(0.2 - 1.8)来修正。所有等待队列中得分最高的客户端进行文件的下载。在以下情况发生时客户将中止下载:

- 1. 上传客户端被用户关闭了。
- 2. 现在客户端完全下载完成了。
- 3. 现在中的客户端被逼他用的优先级更高的客户抢占了。

为了使刚开始进行下载的客户端不被马上被队列中优先级比它高的客户抢占, eMule规定将该开始下载的前15分钟里将下载客户的优先级设定为200。

4.3.5 到达上传队列的最顶端

当A到达B的上传队列的最顶端, B和A进行连接, 完成初次握手然后发送同意下在请求消息(6.4.11章节)。此时A可以发送一个请求部分消息来开始下载文件也可以发送一个取消传送消息(6.4.12章节)来中止(在已经从其他的资源取得了这部份文件时)。图4.6将详细描述这个过程。

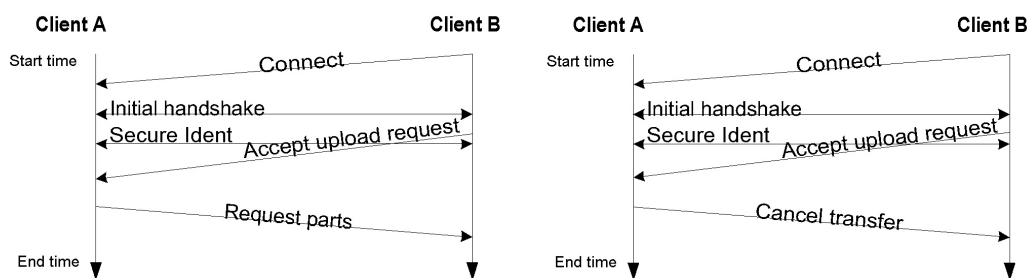


图4.6: 文件请求开始下载

4.4 数据传输

4.4.1 数据包

发送和接收文件是eMule网络活动的主要部分。由FTP传输我们可以推断出eMule在发送文件匹配数据传输中起到控制作用。一次发送的数据量可以在5000到15000字节之间(取决于压缩程度)。为了避免文件破碎，一个文件被分割为很多部分来传输，每一部分放在一个独立的TCP封包当中。在eMule 0.30e中每一部分最大为1300字节(注意这个数字仅取决于TCP的负载能力)。换句话说控制消息的TCP封包有时包含其他的信息，但是数据信息却是以封包为单位的。第一个封包包含了发送文件的信息报头(6.4.3章节)。剩下的封包中仅包含了数据。有时被发送的文件比1300字节略大一点，这部分内容也放在第一个封包中(被包含在报头当中)。图4.7将详细说明文件部分消息。

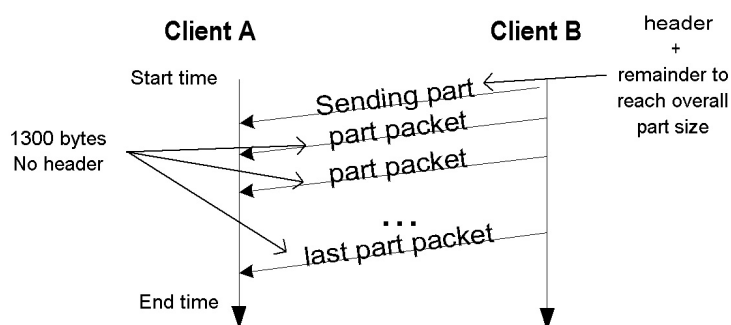


图4.7: 文件部分消息细节

4.4.2 数据传输序列

在文件请求恢复后将立即进行文件传输。正在下载的客户A发送一个开始上传请求(6.4.10章节)然后被回复一个街上上传请求消息(6.4.11章节)。之后A立即还是接收文件(6.4.4章节)，B开始发送文件(6.4.3章节)。注意一个文件部分请求可以到达3部分因此所以每一个文件部分请求应答也到用到3个发送部分序列。当两个客户端都支持扩展协议的时候文件数据将会被压缩(6.5.3章节)。扩展协议也支持一个可选择的文件信息消息(6.5.5章节)，这个消息在接受到同一上传请求消息之后立即发送。图4.8将会详细描述数据传输序列。

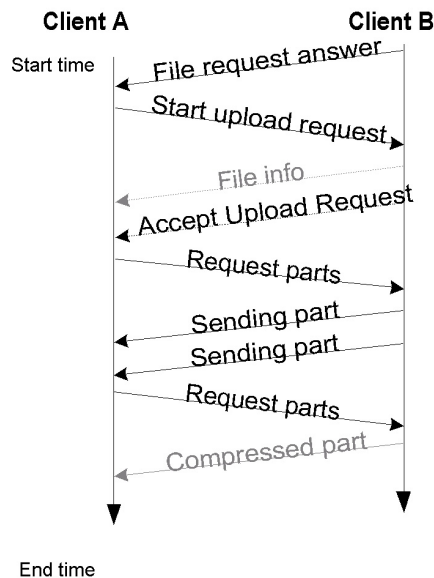


图 4.8: 数据部分交换

4.4.3 选择哪一部分要被下载

eMule有选择性的选择一个文件各部分的下载顺序来达到最大的网络吞吐量和共享性。每一个文件被分割为很多部分，每一部分大小为28MB，每一部分又被分割为很多大小为180k的块。文件块的下载顺序由正在下载的客户端的文件块请求消息(6.4.4章节)决定。下载中的客户可以在任何一个给定的时间内下载文件的一个部分和从这个资源下载这个文件部分的所有块。

下面是下载优先的规则:

1. 文件块(可用的)出现的频率，稀少的文件块必须被尽快的下载来成为新的可获得的资源。
2. 用来预览文件块(第一和最后一块)，预览或者检查文件(比如电影，mp3)
3. 请求状态(正在下载中的)，试着向每一个资源请求另外一块文件块。请求被发送到所有的资源。
4. 完成(最短的先完成)。在开始另外一个下载之前要先完成所有为完成的块。

文件出现的频率被分为三个等级：非常细又，稀有和普通。在每一个等级都有一个标准来计算每一个部分的等级。低等级的部分被先下在。下面的标详细的说明了文件的级别信息：

- 0-9999 – 请求的和未被请求得非常稀有的部分
- 10000-19999 – 未被请求的稀有部分和预览部分
- 20000-29999 – 未被请求的完成最多的部分
- 30000-39999 – 请求非常多的和预览部分
- 40000-49999 – 未完成的普通部分

这个优先法则通常选择第一个最稀有的部分。然而接近完成的部分被下载的文件块也将被选择，这回从一些不同的资源下在。

4.5 浏览共享文件和文件夹

eMule有两个消息来浏览peer客户的共享文件和文件夹。第一个是浏览共享文件消息(6.4.21 章节)，他将在建立完第一次握手之后立即被发送。总是有一个共享文件应答消息(6.4.22 章节)来回复这个消息。当应答客户端想隐藏他的共享文件时应答消息文件中包含 0个文件(而不是否认这条消息)。4.9将详细地描述这个过程。

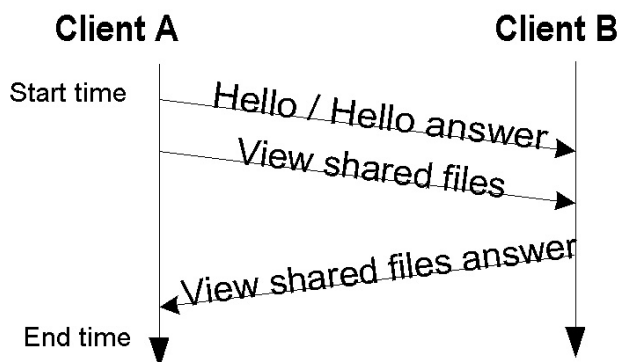


图4.9: 浏览共享文件

第二个消息是请求共享文件夹列表(6.4.23 章节)，然后再为每一个共享文件夹发送一个共享文件夹消息(6.4.24 章节)请求信息(6.4.25 章节)。每一个这种消息将会收到一个文件夹信息(章节6.4.26)作应答。图4.10 将详细说明这个过程。

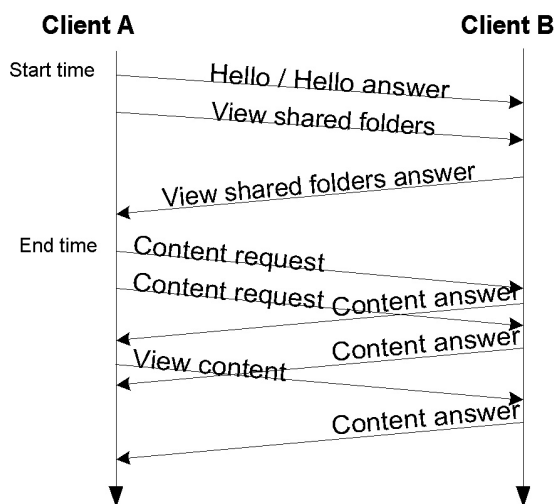


图 4.10: 浏览共享文件和共享文件夹

有时在接收客户端想阻止浏览共享文件/文件夹请求时,它将回复一个拒绝消息, 像图4.11中描述的那样。

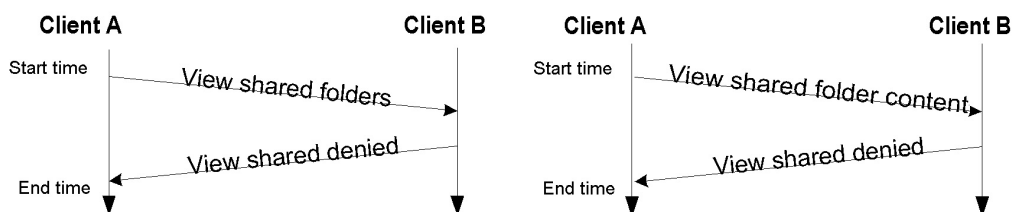


图 4.11: 浏览被拒绝

4.6 交换部分哈希组

通过发送一个哈希组请求来得到部分哈希，这个请求被哈希组回执回复，其中包括文件中每个部分中的哈希组。图4.12对其进行了说明。

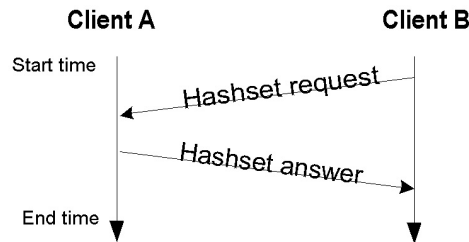


图4.12: 哈希组请求

4.7 得到一个文件的预览

客户端可以要求他们的peer来得到一个下载完毕文件的预览。预览是依靠应用程序的并且有不同的文件种类。EMULE 0.30E 只支持图像预览。信息交换在 图4.13中描述并仅包含2个信息： 预览请求(6.5.11 章节)和预览应答(6.5.12 章节)。

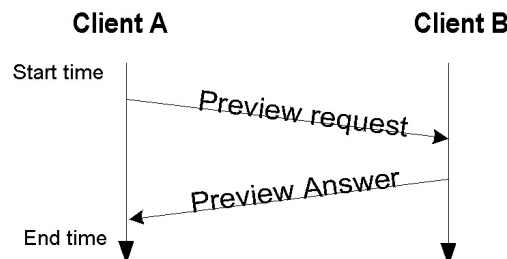


图4.13: 得到文件的预览

5 客户端到客户端的UDP信息

emule客户端不时地发送实用UDP协议的信息。在EMULE 0.30e UDP信息仅在询问客户端在它的peer下载序列中的位置时使用。简单的请求-回应计划时有一个re-ask文件信息(6.6.1 章节)初始化的。在图5.1有3种可能的向这种以上解释过的消息回复：

- 1.队列等级- 客户端在发送者队列种的等级头衔；
- 2.队列已满- 发送者队列已满；
- 3.没找到文件- 发送者没有它清单中的请求文件

re-ask文件信息大约每隔20分钟发送给每个客户端，这些客户端在他的下载队列中添加了发送者。

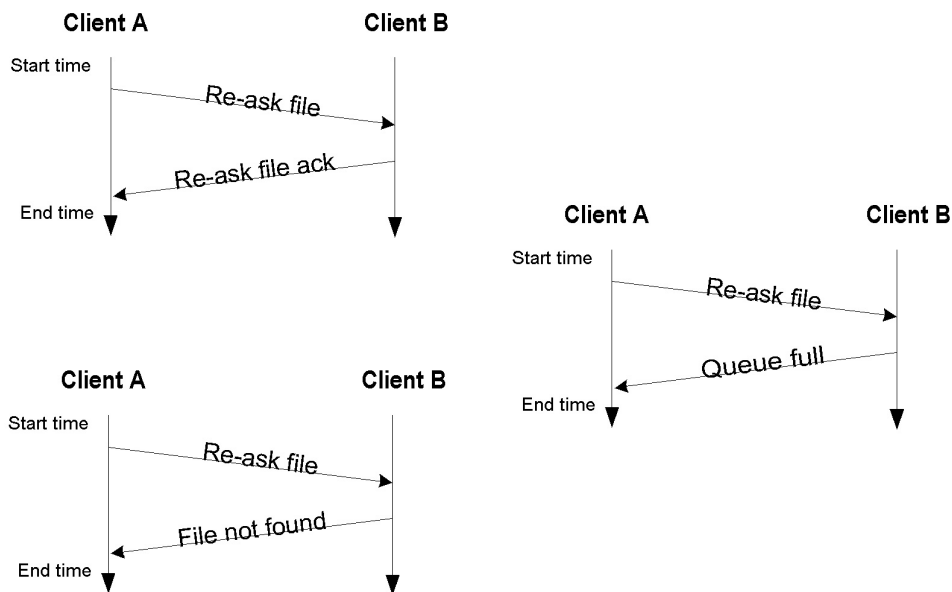


图5.1: re-ask文件信息

参考条目

- [1] D. Bickson 和 D. Malkhi. 研究独立的文件共享网络. Leibniz 研究中心的TR-2003-67研究报告,耶路撒冷的Hebrew 大学, 以色列, 2003.
- [2] eMule 在 source force. <http://sourceforge.net/projects/emule/>.
- [3] eMule 项目. <http://www.emule-project.net/>.
- [4] MD4 哈希函数说明书. <http://www.ietf.org/rfc/rfc1320.txt>.

6 附录A – 消息编码

6.1 常规消息编码说明

这部分主要描述了建立在TCP/UDP基础上的常规消息编码说明

6.1.1 Endianity

所有的消息都是以little-endian规则进行编码的不是传统网络上的以big-endian顺序进行编码。这可以很容易的解释为什么基于Microsoft Windows的客户端/服务器程序用在Intel处理器上。

6.1.2 消息头

所有的消息都包含有一个6字节的消息头，结构如下：

1. 协议- 一个1字节的标示符 - eDonkey是0xE3 ， eMule是0xC5
2. 大小- 消息大小为4字节 - 不包括标题和size字段的信息大小。例如有时像6.4.11章节中所表述的那样消息中不包含任何有效内容他的长度就为0。

3. 种类- 一个一位字节 – 一个唯一的消息 ID

6.1.3 消息标记

标记是一个TLV-like(种类，长度，内容)结构，用来传输可选择的数据。有好几种标记，他们将在本章中被一一列出。当读者想要一些有特别用途的特定标记时，可以以本章中精确定义的一些结构作为参考。一个标记拥有4部分，他们在消息中并不是连续的：

- 1. type- 1字节的整型
- 2. name –可以是下面中的任意一种
 - 可变长的string类型
 - 1字节的整型
- 3. value -可以是下面中的任意一种
 - 4字节的整型
 - 4字节的浮点型
 - 可变长的string类型
- 4. Special- 1字节的整型，特殊标记制定位

整型的我们叫做整型标记，对于浮点型和字符串型也一样。字符串标记有两种，一种是3位的整型另外一种是4位的浮点型。他们按照上面定下来的顺序进行编码比如第一个是type，然后是name，最后是value。Type占一个字节。Name占一个两个字节的结构及可以是String的 也可以是 Integer的。例如Integer名字是0x15将被编码为 0x01 0x00 0x15。定值域(比如整型和浮点型数字)按照规定的填写，string值按一样的长度编码，但是治还是不变的。注意name没有特别的意义只是为了将来更容易的描述消息。

6.2 客户服务器的TCP信息

这个章节描述了服务器与客户端之间使用 TCP port 传递的信息。

6.2.1 登入

登入信息是 TCP 连接完成后由客户端向服务端发出的首个信息。信息的长度的变化取决于如用户昵称的用户设置。为什么应该发送客户端 TCP 传输还不清楚，在 TCP 标题中也会发生。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x01	OP_LOGINREQUEST opcode
Client Hash	16		详细的信息兼 section 1.4
Client ID	4	0	发送第一次连接的 Client ID 通常为 0.详见 section 1.3

TCP port	2	4662	TCP port 是客户端使用的，是可以配置的
Tag Count	4	4	紧跟着信息的标签数
Name Tag	不定	N/A	用户的昵称（是可以在软件中配置的）。 Tag 是一个标签串并且标签的名字是值为 0x1 的整数。
Version Tag	8	0x3C	由客户端支持的 eDonkey 版本。 Tag 是一个整数标签并且标签的名字是一个值为 0x11 的整数。
Port tag	8	4662	TCP port 是由客户端使用的。 Tag 是一个整数标签并且标签的名字是一个值为 0x0F 的整数。
Flags Tag	8	0x01	Tag 是一个整数标签并且标签的名字是一个值为 0x20 的整数。

6.2.2服务器消息

服务器消息是在不同情况下由服务器发向客户端的长度可变的消息，首要的情况是客户端马上发送登录请求后。一个简单的服务器信息可能包括几个由新分隔符分开的一些信息。（‘\r’ ,’\n’,或者两者都有）。由 “server version” ,”warning”,”error”和”[emDynIP:”的信息对客户端来说有特殊的含义。 其它信息简单得显示给用户即可。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x01	OP_LOGINREQUEST opcode
Size	2	N/A	不包括到现在为止描述过的领域中的信息剩余的字节数
Messages	不定	N/A	由 new lines 分割的服务器信息清单

特别信息

- 1. version- 经常在一个成功的同步交换连接中返送
- 2. error-
- 3. warning- 经常在服务器拒绝连接或者客户端中有 low ID 的时候发送
- 4. emDynIP

6.2.3 ID 改变

服务器发送的ID改变消息，作为登入请求的应答并且意味着服务器允许连接。消息大小是14或16字节取决于发

送的TCP连接。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x40	OP_ IDCHANGE opcode
Client ID	4	NA	客户ID的详细内容请看1.3章节
TCP connection bitmap	4	0x00000001	通常只有 1bit 有意义(LSB), 将其设置为以表明服务器支持压缩

6.2.4 提供文件

这个消息是客户用来描述它所拥有的可以提供其他客户下载的文件。当用户有文件请求时，文件请求消息在连接被建立后立即发送。当客户的共享文件发生变更时也发送这条消息。这个消息的另外一个作用是保持激活，周期性的发送给服务端。当服务器支持压缩文件请求时将被压缩。当用来保持激活时(没有文件压缩消息大小小)这个消息不被压缩。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x15	OP_ OFFERFILES opcode
File Count	4	NA	消息中文件描述的数量。任何情况下不超过200。服务企业能设置这个限制。
Files	可变	NA	可选择文件列表，格式是下面描述的一个片断。

一个文件片段的格式

下面的表格描述了一个文件片断。一个文件请求消息中有几个连续的文件片断。

名称	字节数	默认值	注释
Hash Code	16	NA	文件内容的哈希值(详细的 TBD)。这个哈希值用来唯一的识别一个文件。避免不同客户的文件名不同。
Client ID	4	NA	当用户是 high ID 时是 high ID 值 否则是 0
Client Port	2	0x15	客户的 TCP 端口当是 low id 时是 0

Tag Count	4	NA	下面的标记的数量
File Name Tag	可变	NA	文件名，这是一个字符串标记，标记是一个整型值 0x1
File Size Tag	8	NA	文件大小，单位是字节。这是一个整型标记，标记是一个整型值 0x2
File Type Tag	不定	NA	文件类型(可选择的)，下面中的一种：“Audio”，“Video”，“Image”，“Pro”或“Doc”这是一个字符串标记，标记是一个整型值 0x3
File Format Tag	不定	NA	文件的格式(可选择的)。例如-“zip”，“exe”。这是一个字符串标记，标记是一个整型值 0x4
Media Length Tag	不定	NA	如果是 mp3 文件，歌曲的长度(可选)。这是一个字符串标记，标记是一个字符串值“length”
Media Bitrate Tag	TBD	NA	如果是 mp3 文件 编码的比特率(可选)。这是一个字符串标记，标记是一个字符串值“ bitrate”
Media Codec Tag	不定	NA	如果文件是个电影，编码器的类型(可选，不发送)。这是一个字符串标记，标记是一个字符串值“ codec”

6.2.5 获得服务器列表

这个消息在初次握手完成后客户立即发送给服务器。当客户通过向它正连接着的客户寻求服务器列表时发送这个消息。消息大小是6字节。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x14	OP_SERVERSTATUS opcode

6.2.6 服务器状态

服务器发送给客户的。消息包括服务器的当前连接数和文件数。这个消息被客户储存并显示。消息的大小是14字节。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x34	OP_SERVERSTATUS opcode
User Count	4	NA	服务器当前登陆用户数
File Count	4	NA	服务器纪录的文件数

6.2.7 服务器列表

服务器发送给客户的。这个消息包含了客户服务器列表重要增加的服务器的消息。消息大小不定(取决于传送服务器的个数)。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x32	OP_SERVERLIST opcode
Entry Count	1	NA	消息中描述的服务器的个数
Server entries	(Entry Count)*6	NA	服务器描述，每一个描述 6 个子节 4 字节 IP 地址 2 字节端口

6.2.8 服务器认证

服务器发送给客户的。包含一个服务器哈希(TBD)，服务器IP地址和TCP端口(当通过代理连接时有用)并且也有服务器的信息。消息大小不定。

名称	字节数	默认值	注释
Protocol	1	0xE3	
Size	4		不包括标题和 size 字段的信息大小
Type	1	0x41	OP_SERVERIDENT opcode
Hash	16	NA	服务器的 GUID(好像是用来 debug 的)
Server IP	4	NA	服务器 IP 地址
Server Port	4	NA	服务器监听的 TCP 端口
Tag Count	4	NA	消息末尾的标记数
Server Name Tag	不定	NA	服务器的名字。是一个字符串标记，标记名是一个整型值是0x1

Server Description Tag	不定	NA	服务器的描述。是一个字符串标记，标记名是一个整型值是0xB
------------------------	----	----	-------------------------------

6.2.9 搜索请求

客户发送给服务器的。消息使用用户的一个搜索字符串搜索一个文件。消息大小是可变的。搜索字符串包括布尔运算符'AND','OR','NOT'。用户可以详细设置文件的类型大小也可以设置开始位置(例如：展示给我至少5个其他客户端的结果)。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x16	OP_SEARCHRESULT opcode
Parsed search string	不定	NA	下面所描述的搜索字符串结构
Result list	不定	NA	搜索结果列表
Parsed search string	不定	NA	解析搜索字符串，格式如下
File Type Constraint	不定	NA	可选的。一个字符串约束。字符串值是"Audio", "Video", "Pro" or "Image"三者之一。类型域分别对应0x1 0x0 0x3
Min Size Constraint	不定	NA	可选择的，一个整型约束。以兆字节计算的文件大小。类型域有4位：0x1 0x1 0x0 0x2
Max Size constraint	不定	NA	可选择的，一个整型约束。以兆字节计算的文件大小。类型域有4位：0x2 0x1 0x0 0x2
Availability Constraint	不定	NA	可选择的，一个整型约束。搜索文件客户数量的最小上限，类型域有4位：0x1 0x10x0 0x15
Filename Extension	不定	NA	可选择的，一个整型约束。类型域有3位：0x1 0x0 0x3

constrain			
-----------	--	--	--

解析搜索字符串格式

解析字符串编码是通过二差树和'AND','OR','NOT'布尔运算符以及字符串操作数。二差树是按照先序编码进行的。操作是编码两位字节值是

The tree is encoded in pre-order . The operators0x0, 0x100, 0x200分别代表了'AND','OR' 和'NOT'。字符串按照TLV格式进行编码是一个以字节的值和一个两字节的长度。注意当字符串是以各自的时候它代表字符串操作(没有运算符)。以后的eMule 编码版本中指通过单独的字符串'AND'来编码搜索表达式，由空格代替'AND'。

由'AND'运算符将连续的单词分开来组成一个句子。

可选的约束格式

约束数列条目。每一个条目有'AND'描述符开始(2-byte0x00)紧跟着是编码约束。因此一个完整的搜索行格式是：<'search-string' AND constraint1 AND constraint2 etc>就想下面图例中描述的一样。编码被分成三部分。

- 1. 类型-一个字节的描述，表明是字符串(0x2)还是整型约束(0x3)。
- 2. 值-一个定长字符串编码或者一个4字节的整型值
- 3. 类型-一个3或4字节的约束类型描述(看上面的主要表)

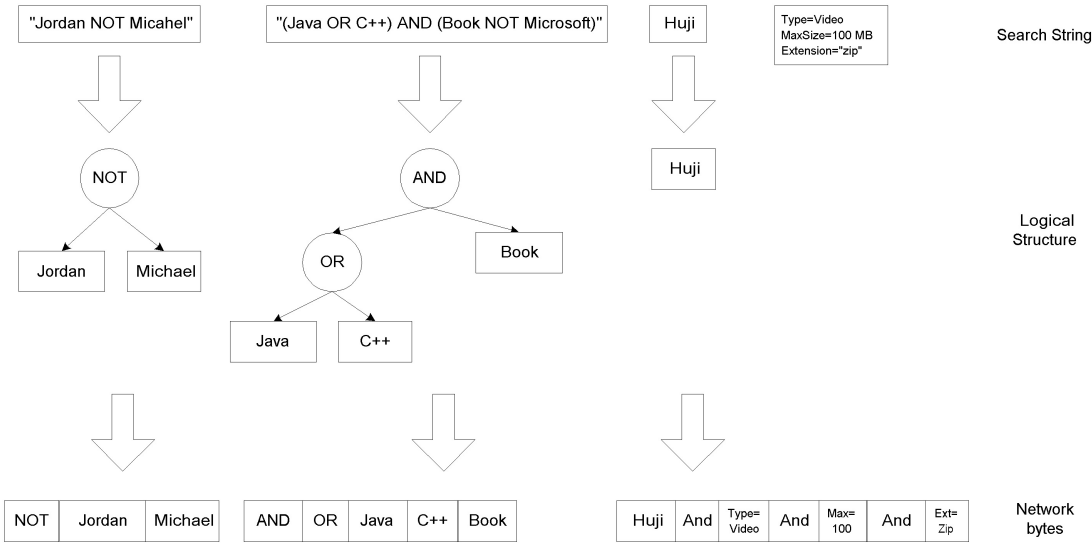


图 6.1: 查找字符串编码实例

6.2.10 搜索结果

服务器发送给客户的搜索请求应答消息。这个消息通常是压缩的。大小不固定。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x16	OP_SEARCHRESULT opcode

Result Count	4	NA	这个消息中搜索结果的个数
Result list	不定	NA	搜索结果列表

搜索结果列表项目的格式

下面的表格描述了一个搜索结果列表项目的格式。每一个搜索结果包含了一个唯一表示这个文件的哈西数和两外一个又有该文件的客户。还有几个描述文件属性的标记。这些标记将在下面描述：

名称	占用字节数	默认值	说明
File Hash	16	NA	一个哈西值，唯一的标示了一个文件
Client ID	4	NA	一个拥有该文件的peer的客户ID
Client Port	2	0x16	拥有文的客户端得端口号
Tag Count	4	NA	后面的描述标记的个数
Tag list	不定	NA	描述标记列表

注意大多数标记的位置是可以变动的，并不是固定的。标记编码的规则在本章开始处已经详细描述了。

名称	标记名称	标记类型	说明
File name	Integer 0x01	String	
File size	Integer 0x02	Integer	
File type	Integer 0x03	String	
File format	Integer 0x04	String	
Sources	Integer 0x15	Integer	这个文件可获得的资源数
Artist	String "Artist"	String	
Album	String "Album"	String	
Title	String "Title"	String	
Length	String "length"	Integer	
Bitrate	String "bitrate"	Intege	
Codec	String "codec"	Integer	

表6.1: 搜索结果标记列表

6.2.11 获得资源

客户向服务器发送一个请求来获得文件资源(其他的客户)。消息的大小是22字节。

名称	占用字节数	默认值	说明
----	-------	-----	----

Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x19	OP_ GETSOURCES opcode
File hash	16	NA	请求的文件哈希数

6.2.12 建立资源

服务器发送给客户端一个包含客户端所请求的文件资源(其他的客户)的消息。消息大小是不定的。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x42	OP_ FOUNDSOURCES opcode
File hash	16	NA	请求的文件哈希数
Sources Count	1	NA	这个消息中的资源数
List of sources	不定	NA	资源列表

资源列表项目的格式

下面的表格描述了资源列表项目的格式。每一个资源包括客户所拥有要下载文件的细节。

名称	占用字节数	默认值	说明
Client ID	4	NA	拥有文件的eMule peer得Client ID
Client Port	2	NA	拥有文件的客户的端口

6.2.13 复查请求

客户端向服务器发送一个请求，要哦求另外一个客户来应答-例如连接请求的客户。

这个消息是由一个想与low ID客户连接得high ID客户发送的(见2. 4章节)。消息的长度是10个字节。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x35	OP_ CALLBACKREQUEST opcode
Client ID	4	NA	想要被回叫的客户得ID

6.2.14 复查请求

从服务器发送到客户端的消息，标志着另外一个库护短迎接接收到消息正要和它取得连接。这个消息被发送当接受客户拥有的是low ID时(见2.4章节)。消息的长度是12字节。接收到消息的客户端试着连接复查请求封包中的IP地址和端口。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x35	OP_ CALLBACKREQUESTED opcode
Client IP	4	NA	客户将自己的IP传给它想连接的客户
Client TCP Port	2	NA	想让连接客户监听的端口

6.2.15 复查失败

服务器发送个客户端一个消息通知客户端的复查请求失败，消息的大小是6个字节。接收到这个消息的客户纪录然后丢弃它。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x36	OP_ CALLBACK FAILED opcode

6.2.16 拒绝消息

服务器发送给客户端的消息用来拒绝客户短发送来的最后一条消息。消息的长度是6字节。接收到这个消息的客户纪录然后丢弃它。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x05	OP_ SEARCHRESULT opcode

6.3 客户服务器的UDP消息

本章描述了服务器和客户端之间使用UDP传输的消息。UDP的消息报头不包含UDP消息的大小，但是大小可以从UDP的L3级报头推断出来。客户端回乡服务器列表中的服务器周期性的发送消息都是固定大小的消息。

6.3.1 获取资源

客户发送个服务器来请求一个文件的资源(拥有这个文件的其他客户)。如果资源数量太少的话这个消息每秒都会发送。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x9A	OP_ GLOBGETSOURCES opcode
File ID List	NA	NA	一个文件ID(哈希数)列表(每一个长16字节)这些ID无空隙的一个接一个的连接着。

6.3.2 找到资源

服务器发送给客户端的UDP请求资源应答消息。只有当为请求的文件找到资源时才发送这个消息。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x9A	OP_ GLOBFOUND SOURCES opcode
File source list	NA	NA	如同下面描述的资源的一个列表

资源列表项目的格式

这个列表包含了一个文件的资源

名称	占用字节数	默认值	说明
File ID	16	NA	要寻找组员的文件的ID
Sources Count	1	NA	报告的资源数
List of sources	NA	NA	和6.2.11章节中的TCP请求资源消息中的资源列表格式一样

6.3.3 状态请求

每几秒就要向服务器发送一个状态请求消息。这个消息包含了一个可以被服务器回应的4个字节的随即标示。这个消息长6字节。是UDP持续消息(见3.1章节)的一部分。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x96	OP_ GLOBSERVSTATREQ opcode
Challenge	4	NA	一个发送给服务器的无符号整型，用来

			验证应答的(在客户端相应的变量被叫做'time')
--	--	--	---------------------------

6.3.4 状态应答

服务器发送的客户的UDP状态请求消息的应答。包含几个服务器状态项目，注意大多数项目是可选择的。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x97	OP_ GLOBSERVSTATRES opcode
Challenge	4	NA	一个发送给服务器的无符号整型，可能是客户发送的标记的回复或者一个不同的数字。后面这种情况停止封包处理。
User Count	4	NA	可选择的。登入服务器的用户号
Files count	4	NA	可选择的。服务器数据库的文件数
Soft files limit	4	NA	可选择的。无符号整型，服务器软上限
Hard files limit	4	NA	可选择的。无符号整型，服务器硬上限
UDP flags	4	NA	可选择的。服务器的UDP标记。有下面两种标记：0x01表明服务器支持获得资源消息。0x02表明服务器支持扩展获得资源消息。

6.3.5 搜索请求

当服务器采用UDP来进行搜索时客户将这个消息发送给服务器列表中的服务器。这个小有两个可选择的 opcodes(0x98 或 0x92), 后面指出了服务器的最新(最好)版本。客户端通过服务器发送的UDP状态应答消息(6.3.4 章节)中的UDP标记来决定使用那种opcode。

为了发送增强的opcode服务器必须开启0x02 bit。UDP标记也可以从一个配置文件中加载。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x98 or 0x92	OP_ GLOBSEARCHREQ opcode 或者 OP_ GLOBSEARCHREQ2 opcode

Search request parameters	可变的	NA	与6.2.9章节中客户端发送给服务器的TCP信息章节中的搜索请求消息一样
---------------------------	-----	----	--------------------------------------

6.3.6 搜索应答

服务器发送个客户的一个搜索应答消息。回复两种含有不同的opcodes的消息都是使用这条消息。尽管这个结果中不含有明确的数量，这个消息还是与TCP的搜索结果消息很相似的。请阅读本书中的TCP搜索结果消息的相关章节去了解更过本消息中的可变块的内容(6.2.10章节)。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x99	OP_GLOBSEARCHRES opcode
Result list	NA	NA	搜索结果列表(没有数量)，如同6.2.10章节中描述的一样

6.3.7 服务器状态请求

每几秒就会发送给服务器一次。不负载任何内容。UDP ping配置的一部分。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x A2	OP_SERVER_DESC_REQ opcode

6.3.8 服务器状态应答

服务器发送个客户端的服务器状态请求的应答。包括名称和服务器的描述。消息长都市不定的。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Type	1	0x A3	OP_SERVER_DESC_RES opcode
Name	NA	NA	服务器的名称，2-byte长的字符串编码格式是字符数组
Description	NA	NA	服务器的描述，2-byte长的字符串编码格式是字符数组

6.4 客户到客户的TCP消息

本章节描述了客户之间使用的TCP消息。客户之间的消息被分为eMule和eDonkey两种-注意下面描述的消息头字

段的名称。这里也有一个标记用来传输文件的。

6.4.1 Hello

这是两个e-mule客户进行握手的第一个消息。这个消息非常地向服务器登陆消息(见6.2.1章节)。两种消息都含有相同的类型编码(0x01)，他们是协议中唯一含有重叠编码的消息。两种消息都提供相同的数据甚至是使用相同的顺序。有两点主要的不同：客户的hello消息是有一个user hash字段开始的然而服务器登陆消息质就是由用户哈西值开始的，另外和服务器登陆消息不同的是客户hello消息以额外的服务ip和端口结束。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x01	OP_HELLO opcode
User Hash size	1	16	user hash 域的大小
User Hash	16		TBD
Client ID	4	0	TBD
TCP Port	2	4662	用户使用得TCP端口，可变
Tag Count	4	4	消息后面的标记数
Tag list	可变的	NA	一个详细说明客户状态的标记列表
Server IP	4	NA	给客户端连接的服务器IP地址
Server TCP Port	2	NA	服务器监听的TCP端口

在tag-list中会出现三种标记。通常不使用的端口标记。标记的编码规则如同本章开始时描述的那样。

名称	标记名称	标记类型	说明
Username	Integer, 0x01	String	
Version	Integer 0x11	String	
Port	Integer 0x0F	Integer	

6.4.2 Hello 应答

Hello消息的应答。除消息类型之外其他和hello消息相同。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	

Size	4		不包括标题和size字段的信息大小
Type	1	0x4C	OP_ HELLOANSWER opcode
Hello fields			和hello消息相同

6.4.3 发送文件部分

这个消息包含了要下载文件的一部分。不包括标题和size字段的信息大小被明确地指出来(在报头当中)。这个消息被分割为好几个数据包，每个数据包都比在eMule 0.30e的TCP MTU中定义的最大的大小1300字节小。??章节描述了发送数据部分的详细信息。也可以看6.5.3章节去了解发送压缩文件部分的详细信息。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x46	OP_ SENDINGPART opcode
File ID	16	NA	根据文件内容由哈西算法算出来的唯一的文件ID
Start Pos	4	NA	下载数据的开始位置
End Pos	4	NA	下载数据的结束位置
Data	NA	NA	要下载的数据，可能是被压缩的

6.4.4 请求文件部分

发送给peer客户请求的文件。这个消息可以最多请求3各部分来下载。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x47	OP_ REQUESTPARTS opcode
File ID	16	NA	根据文件内容由哈西算法算出来的唯一的文件ID
Part 1 Start offset	4	NA	第1部分的位置

Part 2 Start offset	4	NA	
Part 3 Start offset	4	NA	
Part 1 End offset	4	NA	第1部分的结束位置
Part 2 End offset	4	NA	
Part 3 End offset	4	NA	

6.4.5 下载结束

下载中的客户端来通知下载结束。客户端接收完成才会发出这个消息(释放资源)。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x49	OP_END_OF_DOWNLOAD opcode
File ID	16	NA	根据文件内容由哈希算法算出来的唯一的文件ID

6.4.6 改变客户 ID

当客户更改服务器(断开现在连接的服务器连接到一个新的服务器)等到新的客户ID时发送。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x4D	OP_CHANGE_CLIENT ID opcode
Client ID	4	0	新的客户ID
Server IP	4	NA	用户连接的服务器IP
Server TCP Port	2	NA	服务器监听的TCP端口

6.4.7 聊天消息

使用eMule提供的聊天服务。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	

Size	4		不包括标题和size字段的信息大小
Type	1	0x4E	OP_ MESSAGE opcode
Length	2	NA	消息的长度
Message	可变的	NA	实际内容

6.4.8 部分哈西组请求

请求每一个文件的每一部分的哈西。在1.5章节中我们一个揭示了有关唯一的文件ID。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x51	OP_ HASHSETREQUEST opcode
File ID	16	NA	请求文件的文件ID

6.4.9 部分哈西组应答

部分哈西组请求的应答包含了全局哈西(所有文件的)和文件每一部分的哈西。当处理完承受客户发送开始下载请求。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x52	OP_ HASHSETANSWER opcode
File Hash	16	NA	所有文件的哈西
Part Count	2	NA	文件的部分数
Part Hashes	可变的	NA	文件的每一部分的哈西，每一个不分大小是16字节

6.4.10 开始上传请求

一个开始上传请求消息。像在4.3章节中讨论的一样这个消息开始了一个文件下载过程。

名称	占用字节数	默认值	说明
----	-------	-----	----

Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x54	OP_ STARTUPLOADREQ opcode
File ID	16	NA	所请求文件的ID

6.4.11 接受上传请求

上传被接受的标志，上船客户端开始等待要上船的部分请求。这个消息只有最基本的eMule消息头。这个消息是发送协议的一部分。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x55	OP_ ACCEPTUPLOADREQ opcode

6.4.12 中止传输

请求停止文件传输。这个请求不包含任何内容。典型的应用是下载客户端达到了下载队列的顶端但是这个文件它已经下载完毕。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x56	OP_ ACCEPTUPLOADREQ opcode

6.4.13 更多部分请求

eMule 0.30e中没有使用.表示下载客户端下载完成所有商船客户端所上传的文件需要寻找更多部分。注意客户端没收到实际内容。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x57	OP_ OUTOFPARTREQS opcode

6.4.14 文件请求

客户向另一个客户请求一个文件。这个消息包含所请求文件的ID和一个描述那几部分已经下载完成的状态字。

状态字域

eMule允许客户从其他客户那里下载即使没有完成的文件部分。这个状态字用来区别被请求的文件不存在和被请求的文件部分被下载。如果文件不存在的话它的值是0。如果文件被部分下在了，开始的两字节整型标明了所下载的部分，最后一个字节标明了文件的那些部分被完全下载了(通过将标识位设定为1)。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x58	OP_ FILEREQUEST opcode
File ID	16	NA	唯一的文件ID
Part Status	3	NA	可选的。如果在eMule信息消息中的额外请求版本大于0，本章将解释文件的意义
Source count	2	NA	可选的。如果在eMule信息消息中的额外请求版本大于1。文件的当前资源数

6.4.15 文件请求应答

一个文件请求应答，作为文件请求消息的回复。这个消息只是可能的文件请求消息的应答之一。具体的细节请看4.3章节。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x59	OP_ FILEREQANSWER opcode
File ID	16	NA	唯一的文件ID
Name length	2	NA	文件名的长度
Filename	可变得	NA	文件名(长度在前面说明了)

6.4.16 文件没有找到

当文件或者部分文件没有找到时将用这个消息恢复文件请求消息。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	

Size	4		不包括标题和size字段的信息大小
Type	1	0x48	OP_ FILEREQANSNOFIL opcode
File ID	16	NA	不存在的文件ID
Part Status	3	NA	文件请求的解释(6.4.14章节)

6.4.17 被请求的文件ID

一个文件请求消息，这个消息只指明了文件ID。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x4E	OP_ SETREQFILEID opcode
File ID	16	NA	请求的文件ID

6.4.18 文件状态

如果客户有请求的文件，此条消息作为文件ID请求消息的回复。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x50	OP_ FILESTATUS opcode
File ID	16	NA	被报告状态的文件ID
Part Status	3	NA	在文件请求中说明(6.4.14章节)

6.4.19 改变slot

eMule 0.30e不发送。忽略所收到的消息。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x5B	OP_ CHANGE_SLOT opcode

6.4.20 队列位置

eMule 0.30e不发送。忽略所收到的消息。更多的详情请看6.5.4.章节。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x5C	OP_ QUEUERANK opcode

6.4.21浏览共享文件

请求一个共享文件列表。这个应答之发生在请求客户端没有被禁止并且没有配置这类的请求。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x4A	OP_ ASKSHAREDFILES opcode

6.4.22浏览共享文件应答

这个消息是浏览共享文件消息的应答。这个消息报换一个共享文件内容的列表。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x4B	OP_ ASKSHAREDFILESANSWER opcode
Result Count	4	NA	消息中文件名的个数
Result List	不定的	NA	和服务器搜索结果编码一样(见6.2.10 章节)

6.4.23 浏览共享文件夹

请求一个共享文件夹列表。这个应答之发生在请求客户端没有被禁止并且没有配置这类的请求。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x5D	OP_ ASKSHARED_DIRS opcode

6.4.24浏览共享文件夹应答

这个消息是浏览共享文件夹消息的应答。这个消息报换一个共享文件夹的列表

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x 5F	OP_ ASKSHARED_DIRSANS opcode
Folder count	4	NA	消息中文件夹的个数
Folder list	不定的	NA	一个文件夹信息描述，格式如下

Folder list 格式

folder list格式是一个登陆序列，每一个条目包括一个短整型(uint16)和一个字符串。

6.4.25 察看一个共享文件夹的内容

一个请求察看共享文件夹内容的消息。这个请求只在文件加倍共享时并且是一个新客户时才有。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x5E	OP_ ASKSHARED_FILESDIR opcode
Name length	2	NA	下一个文件夹名字的长度
Directory name	不定的	NA	共享文件夹名字

6.4.26察看一个共享文件夹的内容应答

察看一个共享文件夹的内容消息的应答(OP_ ASKSHARED_FILESDIR)。这个应答包括一个文件名列表(所有存在文件夹中的文件)。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x60	OP_ ASKSHARED_FILESDIRANS opcode
Name length	2	NA	下一个文件夹名字的长度
Directory name	不定的	NA	共享文件夹名字(下一个共享文件夹)
Files Count	4	NA	共享文件的个数

Files list	不定的	NA	一个名字列表(unit16-length，字符串)
------------	-----	----	---------------------------

6.4.27 拒绝察看共享文件夹

拒绝察看共享文件时发送。这个消息用来拒绝OP_ASKSHARED_DIRS和OP_ASKSHARED_FILES_DIR请求。

名称	占用字节数	默认值	说明
Protocol	1	0xE3	
Size	4		不包括标题和size字段的信息大小
Type	1	0x61	OP_ASKSHARED_DENIEDANS opcode

6.5 客户到客户的TCP扩展消息

下面的消息是eMule的一部分，eDonkey中没有，它们被叫做扩展协议消息。

6.5.1 eMule 信息

消息包含了关于eMule客户端的主要信息。eMule 0.30e的标记列表如下或许在以后的协议中会有改动。eMule信息应答消息会立即恢复这个信息(6.5.2章节)。这个消息是客户初次握手的一部分(4.1章节)。eMule信息包括几个标记表示eMule客户的接受能力，这些标记很重要，下面简要地说明一下。

UDP 版本标记

客户所提供的UDP协议的版本-例如支持哪种客户到客户的UDP消息。这个消息是一个整型标记，它的值是0x22。在0.30e版本时这个值是0x03。

资源交换标记

这个标记表明是否支持客户之间的资源交换，这个标记的值是资源交换支持的版本。这个消息是一个整型标记，它的值是0x23。在0.30e版本时这个值是0x02也可以是0x01。

注释支持标记

表示是否支持文件注释。这个消息是一个整型标记，它的值是0x24。在0.30e版本时这个值是0x01表明支持注释。

额外请求标记

这个标记表明客户有能力发送和接受额外的文件请求。这个消息是一个整型标记，它的值是0x25。在0.30e版本时这个值是0x02表明支持注释。

额外选择标记

在eMule0.30e中 这个标记有两个作用-低6位表明客户是否支持安全认证，下一位(第七位)表明是否支持文件预览。这个消息是一个整型标记，它的值是0x27。

名称	占用字	默认值	说明
----	-----	-----	----

	节数		
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x01	OP_EMULEINFO opcode
Client Version	1	NA	eMule客户的版本
Protocol Version	1	0x01	eMule协议的版本
Tag Count	4	7	标记的个数
Compression tag	8	NA	表明客户支持压缩，这个标记是一个整型标记，它的值是0x20。在0.30e版本时这个值是0x01表明支持注释。
UDP Version tag	8	NA	文中已说明
UDP Port tag	8	NA	这个客户监听的UDP端口，这个标记是一个整型标记，它的值是0x21。默认端口是4672
Source exchange tag	8	NA	文中已说明
Comments tag	8	NA	文中已说明
Extend requests tag	8	NA	文中已说明
Extended options tag	8	NA	文中已说明

6.5.2 eMule 信息应答

eMule 信息消息的应。包含相同的信息。这个消息是客户初次握手的一部分(4.1章节)。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x02	OP_EMULEINFOANSWER opcode
eMule Info fields			和eMule 信息消息相同

6.5.3 发送压缩了的文件

这个消息发送压缩的文件片断。片断或许被压缩了压缩的部分被放在好几个封包就如同普通文件一样(6.4.3章节)。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x40	OP_ COMPRESSEDPART opcode
Opcode File ID	16	NA	唯一的文件ID
Part start offset	4	NA	开始部分的便宜量
Compressed content size	4	NA	压缩内容的大小
Compressed content	不定的	NA	压缩内容

6.5.4 队列位置

回复一个peer客户在它的队列中的位置-在下在队列中等待的位置。这个消息通常在为一个明确的文件而加入队列时发送，紧跟着这条消息的是断开客户的TCP连接。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x60	OP_ COMPRESSEDPART opcode
Opcode Queue position	2	NA	客户在队列中的位置
Buffer	10	0	10个0 目的未知

6.5.5 文件信息

一个包含对文件详细描述的消息。是文件请求或文件上传请求消息的应答。这个消息不包含唯一的文件ID，这个消息建设在接受客户端对一个指定的文件已经建立了一个TCP连接。

名称	占用字节数	默认值	说明
----	-------	-----	----

	节数		
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x61	OP_ FILEDESC opcode
File rate	1	NA	1位表明文件的等级
File Comment length	2	NA	后面信息部分的长度
Comment	可变的	NA	信息内容

6.5.6 资源请求

请求资源(其他拥有文件的客户)。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x81	OP_ REQUESTSOURCES opcode
File ID	16	NA	被请求的文件的ID

6.5.7 资源应答

资源请求消息的应答，这个消息可能是压缩的(当未压缩时长度长于394字节)。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x82	OP_ ANSWERSOURCES opcode
File ID	16	NA	被请求的文件的ID
Source Count	2	30	在eMule 0.30e中总是0
List of sources	不定的	NA	一个可变的请求文件的资源的描述，格式如下。

资源列表项目格式

下面的表格描述了资源列表项目的格式。每一个资源包括了eMule客户拥有文件的详细说明和它现在正连接的服务器。

名称	占用字节数	默认值	说明
Client ID	4	NA	一个拥有文件的eMule peer得客户ID
Client Port	2	NA	拥有文件的客户的TCP监听端口
Server IP	4	NA	用户所连接的服务器的IP地址
Server Port	2	NA	服务器监听客户连接的TCP端口
Client hash	16	30	在eMule 0.30e中总是0
List of sources	不定的	NA	可选择的-当双方都支持资源交换看(OP_EMULEINFO消息)大于1

6.5.8 安全认证

在初次握手后(4.1章节)发出。仅当peer客户支持安全认证(在eMule信息消息中详细描述)。这个消息决定了发送客户是否拥有peer的公钥并且提供了一个随机值用来标记peer客户。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x87	OP_SECIDENTSTATE opcode
Operation	1	2	表明是否是一个公钥和一个标示符(2) 或者 只是一个标示符(1)
Challenge	4	NA	用来标示peer的随机字

6.5.9 公钥

当两边都支持加密算法时包含一个客户的公钥。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x85	OP_PUBLICKEY opcode

Public key length	1	76	公钥的长度
Public key	不定的	NA	通常长度是76个字节

6.5.10 标示符

客户使用4字节的标记表明使用了公钥。标示符包括一个客户发送的一个随机字和另外一个客户IP(当客户拥有的是low ID)或者当客户拥有的是high ID时是客户的ID。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x86	OP_ SIGNATURE opcode
Signature length	1	48	标示符的长度
Signature	不定的	NA	通常长度是48个字节

6.5.11 预览请求

为特别的图片文件请求图片预览。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x90	OP_ REQUESTPREVIEW opcode
File ID	16	NA	文件ID

6.5.12 预览应答

一个图片预览请求消息应答。包含请求文件的预览或如果不是图片文件就是请求文件的文件ID。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x91	OP_ PREVIEWANSWER opcode
File ID	16	NA	文件ID或者是0这时是图片文件

Frame count	1	NA	图片的frame数
Frames	不定的	NA	一个4字节长的图片frame编码

6.6 客户到客户的UDP消息

本章表述了服务器之间和客户之间传送的UDP消息。所有这些UDP消息都是eMule扩展的一部分。所以协议ID时eMule协议而不是eDonkey。

6.6.1 再次请求文件

再次向peer客户询问请求文件的状态。当请求客户不在被请求客户的队列当中并且想要周期性的询问自己是否能下载这个文件时发送这个消息。这个消息有三种可能的答复，下面将说明。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x90	OP_REASKFILEPING opcode
File ID	16	NA	再次请求的文件ID
Source count	2	NA	可选的，无符号整数，请求文件的当前资源数

6.6.2再次请求文件应答

当peer客户收到再次请求文件消息时所作出的应答，下载队列不满并且请求客户在队列里。

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x91	OP_REASKACK opcode
Rank	2	NA	请求客户在被请求客户队列中的位置

再次请求文件应答的一种情况：文件没有找到，消息不指出文件没有找到

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小

Type	1	0x92	OP_ FILENOTFOUND opcode
------	---	------	-------------------------

6.6.3 队列满

再次请求文件应答的一种情况，消息不指出对列已满

名称	占用字节数	默认值	说明
Protocol	1	0xC5	
Size	4		不包括标题和size字段的信息大小
Type	1	0x93	OP_ QUEUEFULLopcode