AnyChat for Linux Server

用户操作手册

（版本：V9.0）

广 州 佰 锐 网 络 科 技 有 限 公 司

GuangZhou BaiRui Network Technology Co.,Ltd.

http://www.bairuitech.com http://www.anychat.cn

2021 年 6 月

目 录

一、系统概述

非常感谢您使用佰锐科技的产品，我们将为您提供最好的服务。

本文档可能包含技术上不准确的地方或排版错误。本手册的内容将做定期的更新，恕不另行通知；更新的内容将会在本手册的新版本中加入。我们随时会改进或更新本手册中描述的产品或程序。

本文档是 Linux 平台上服务器程序的用户手册，包含部署、参数配置、常见问题等部分。

1.1 系统介绍

AnyChat 音视频互动开发平台（SDK）是一套跨平台的即时通讯解决方案，基于先进的 H.264 视频编码标准、AAC 音频编码标准与 P2P 技术，整合了佰锐科技在音视频编码、多媒体通讯领域领先的开发技术和丰富的产品经验而设计的高质量、宽适应性、分布式、模块化的网络音视频互动平台。

AnyChat 音视频互动开发平台（SDK）包含了音视频处理模块（采集、编解码）、流媒体管理模块（丢包重传、抖动平滑、动态缓冲）、流媒体播放模块（多路混音、音视频同步）以及 P2P 网络模块（NAT 穿透、UPnP 支持）等多个子模块，封装了底层的硬件操作（音视频采集、播放）、封装了流媒体处理（编解码、网络传输）等非常专业和复杂的技术，为上层应用提供简单的 API 控制接口，可以在极短的开发周期，以及极少的人力资源投入下为客户的现有平台增加音视频即时通讯、多方会议的功能。

AnyChat SDK 分为客户端 SDK 和服务器 SDK 两大部分，其中客户端 SDK 用于实现语音、视频的交互以及其它客户端相关的功能，而服务器 SDK 主要实现业务层逻辑控制，以及与第三方平台的互联等。Linux 版本的客户端 SDK 和服务器 SDK 均支持 C++、Java 等开发语言。

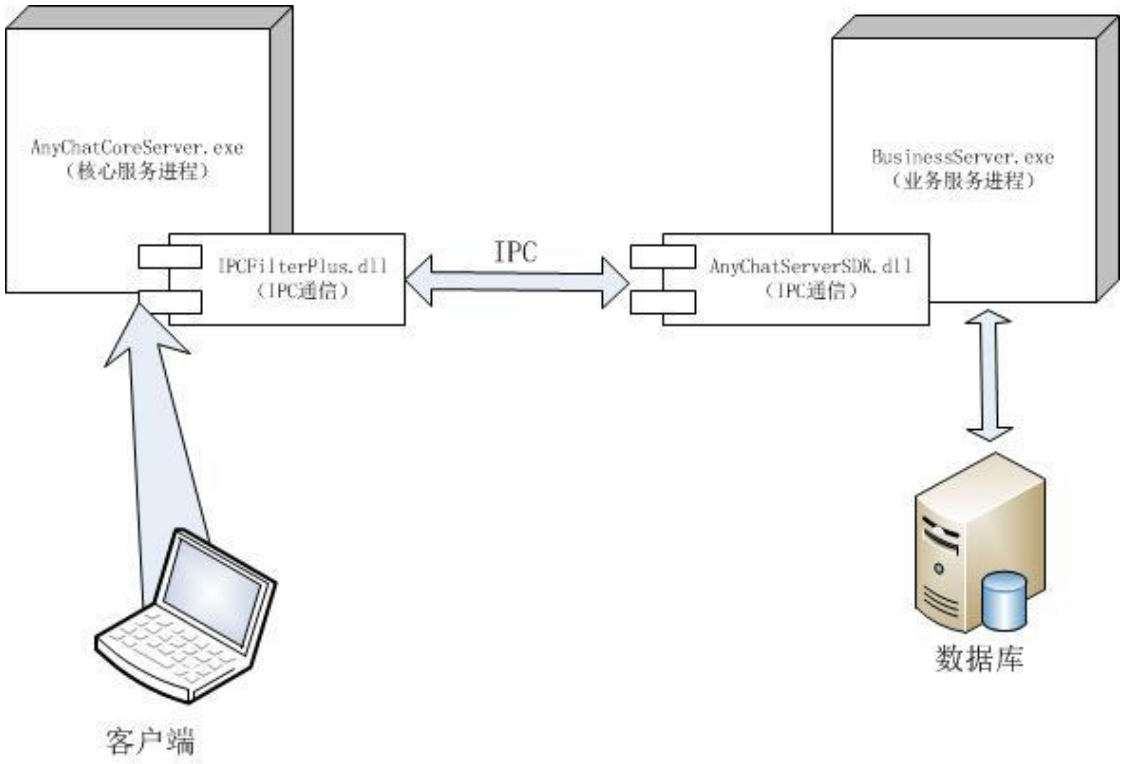
1.2 服务器架构

AnyChat 平台的服务器分两类：核心服务器、业务服务器。

其中核心服务器是指 AnyChatCoreServer.exe 进程，主要负责与客户端的交互、连接管理、房间管理、P2P 穿透过程中的协调以及当 P2P 不通时的流媒体数据的转发等任务；

业务服务器是一个单独的进程，采用 AnyChat Server SDK 提供的 API 接口开发，与核心服务器之间通过 IPC 的方式进行通信，支持分布式部署。业务服务器主要负责业务逻辑的处理，包括用户身份验证、业务流程的控制等，SDK 包中提供了简单的业务服务器示例程序，源代码在 SDK 包的 src 目录中。

有关核心服务器与业务服务器之间的相互关系，请见下图：



AnyChat Server SDK应用模式

1.3 环境要求

目前已经测试且能正常运行 AnyChat 服务器的 Linux 系统基本上涵盖了当前主流的 32bit、64bit Linux 操作系统，已测试版本清单如下：

Red Hat Enterprise Linux：

Red Hat Enterprise Linux Server release 5.4 (Tikanga)　64bit

Red Hat Enterprise Linux Server release 5.6 (Tikanga) 64bit

Red Hat Enterprise Linux Server release 5.7 (Tikanga) 64bit

Red Hat Enterprise Linux Server release 6.0 (Santiago) 32bit

Ubuntu：

Ubuntu 10.04.4 LTS 32bit

Ubuntu 11.10 (oneiric) 32bit

Ubuntu 12.04.2 LTS (precise) 64bit

Ubuntu 14.04 LTS (32bit,64bit)

Ubuntu 16.04 LTS (32bit,64bit)

Debian：

Debian GNU/linux 7 (wheezy) 32bit

Debian GNU/Linux 6.0.6 (squeeze) 64bit

Debian 7（wheezy）

Debian 8（jessie）

Debian 9（stretch）

Fedora：

Fedora release 15 (Lovelock) 32bit

CentOS：

CentOS release 5.7 (Final) 64bit

CentOS release 5.8 (Final) 64bit

CentOS release 6.3 (Final) 64bit

CentOS release 6.4 (Final) 32bit

CentOS release 6.4 (Final) 64bit

CentOS release 6.8 (Final) 32bit

CentOS release 6.8 (Final) 64bit

CentOS release 7.2 (Final) 64bit

CentOS release 7.3 (Final) 64bit

更详细的测试记录可参考在线论坛文档：AnyChat for Linux Server 测试记录，由于 32bit 和 64bit 的程序不能通用，所以在下载时请注意选择与操作系统对应的 AnyChat for linux SDK 包。

二、服务器的部署

下面以 Ubuntu 11.10 32bit 平台为例介绍 AnyChat 服务器在 Linux 平台的部署流程。

2.1 运行核心服务器

在终端使用“cd”命令转到服务器所在【./bin/server】目录，执行“./runanychatcoreserver.sh”命令即可开启服务器。如图 2-1 所示：

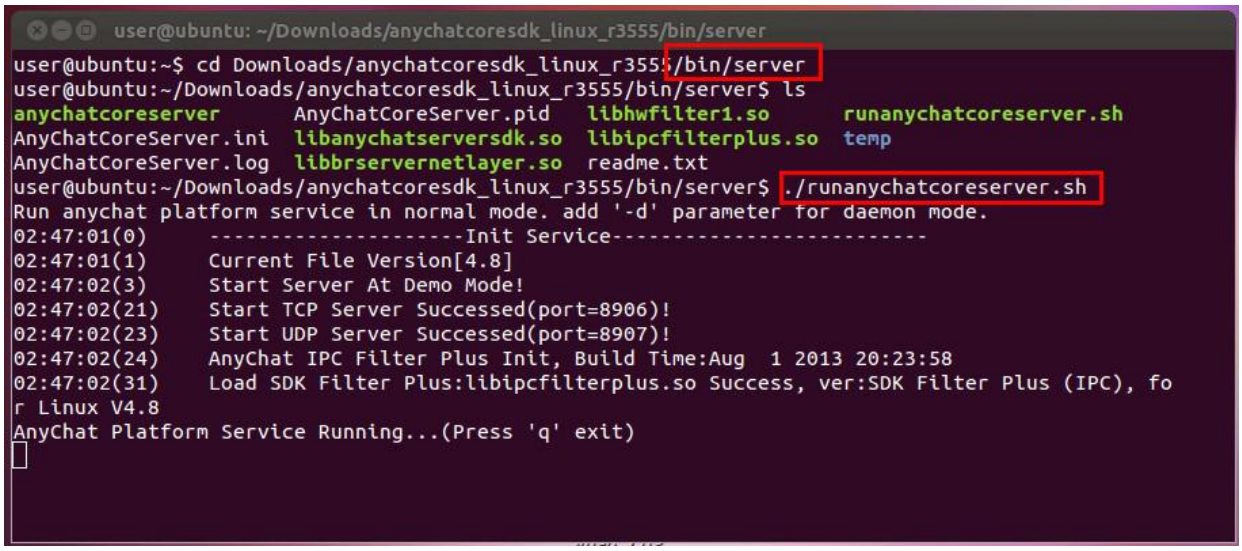


图 2-1 开启服务器

终端界面上将会输出相关的日志信息，显示核心服务器已正常启动，并提示“按 q 键退出程序”；

若希望核心服务器在后台运行，可使用“./anychatcoreserver -d”，如图 2-2 所示：

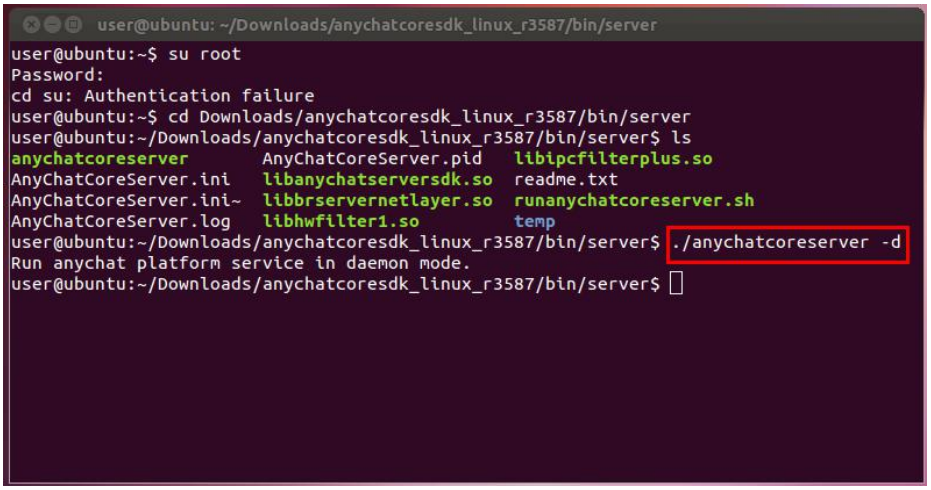


图 2-2 服务器后台运行

2.2 核心服务器守护进程

核心服务器成功运行之后，会在进程列表中出现两个同名进程：anychatcoreserver，其中进程号较小者为守护进程，进程号较大者为工作进程，当工程进程异常崩溃之后，守护进程会自动重启工作进程，保障可以继续对外提供服务。

在结束进程时，“kill 守护进程号”，工作进程会自动结束。

AnyChat for Linux SDK 包含了一个守护进程脚本（位于 SDK 包\bin\server\anychatdeamon.sh），该守护进程脚本会自动监控核心服务器（anychatcoreserver）的工作状态：

1、当核心服务器不在进程列表中时，会自动启动核心服务器；

2、当核心服务器工作状态异常时，会自动结束进程并重新启动新的进程；

通过守护进程脚本可以保障核心服务器 7x24 小时不间断连续运行，正式上线运营的平台建议使用该脚本。配置方法如下：

sudo gedit /etc/init.d/rc.local

在 rc.local 文件的最后加入脚本路径

/home/bairuitech/anychatcoresdk_linux_x64/bin/server/anychatdeamon.sh &

保存修改，重启计算机即可，需要注意脚本最后需要加一个“&”，表示让脚本在后台运行。

如果需要停止核心服务器，请先结束该守护进程脚本，然后再结束核心服务器进程：

sudo killall -9 anychatdeamon.sh

sudo killall -9 anychatcoreserver

2.3 关闭核心服务器

在服务器成功运行后，在运行服务器的终端上面按‘q’键并回车，即可关闭服务器。如图 2-3 所示：

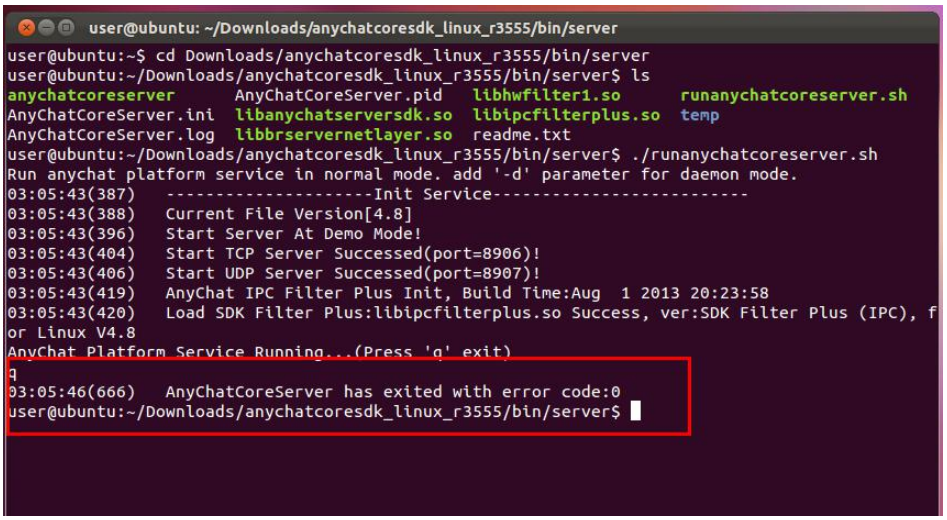


图 2-3 关闭核心服务器

核心服务器关闭成功后，终端界面上将会输出相关的日志信息，显示服务器已正常关闭。

如果核心服务器是后台打开，如图 2-2 所示那样，则需要先查出服务器的进程号再将其杀掉关闭。如图 2-4 所示：先用“ps -ef|grep anychatcoreserver”命令查出服务器进程号，再用“kill”命令杀掉服务器进程即可。

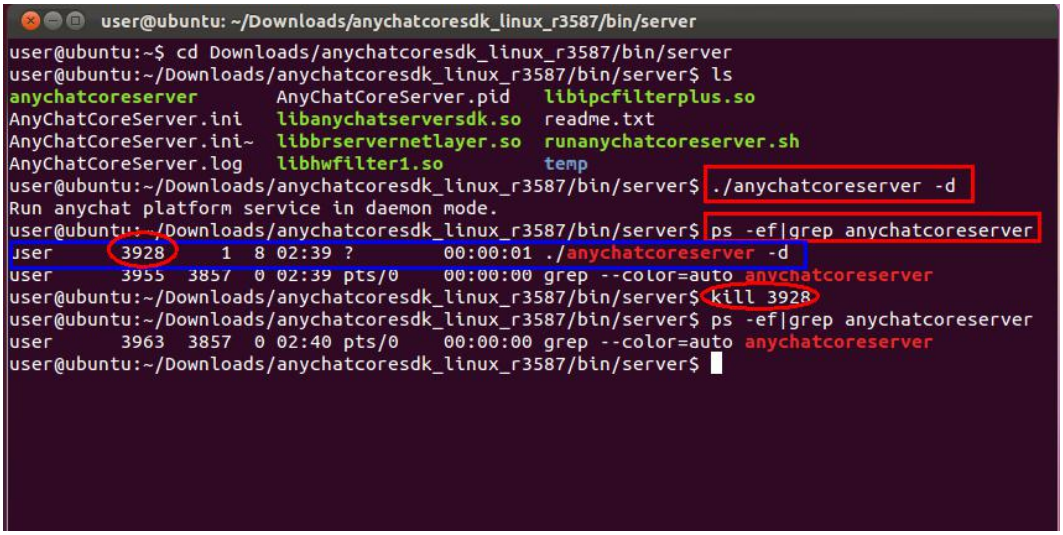


图 2-4 关闭后台服务器

2.4 运行业务服务器

业务服务器是采用 AnyChat Server SDK 开发的应用服务器，独立进程模式，在标准 SDK 包中提供了 C++和 Java 语言实现的简单示例程序，源代码在 SDK 包的 src 目录下，下面以 Java 语言实现的业务服务器为例说明运行流程。

首先需要准备好 Java 环境，然后新打开一个终端，使用“cd”命令转到业务服务器所在【./bin/serversdk】目录，执行“./runbusinessserver.sh”命令即可开启业务服务器，如图 2-5 所示。

开启业务服务器成功时会自动打开一个 AnyChat Server for Java 的示例，此时 JAVA 示例下半部界面会提示“connect core server successed”，表示与核心服务器建立连接成功。关闭 JAVA 程序就可退出业务服务器程序。

当业务服务器与核心服务器连接成功时，用户就可在客户端进行业务操作。如业务服务器与核心服务器连接不成功时，客户端是不能连线不能进行业务操作。

实际应用时，客户需要根据自身的实际需求单独开发业务服务器。

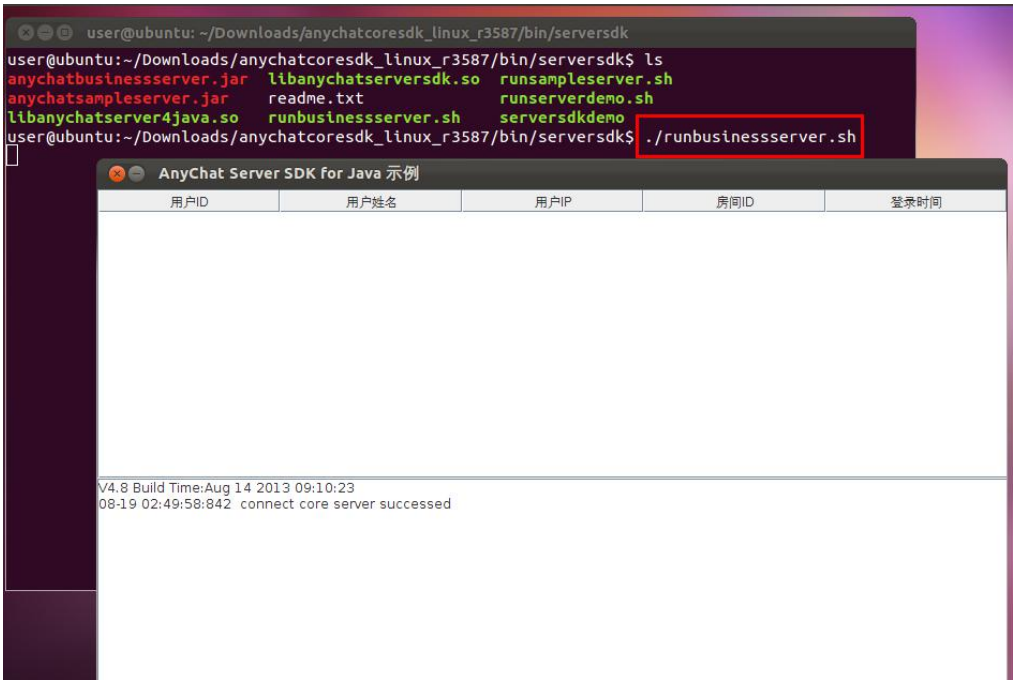


图 2-5 运行业务服务器(Java)

2.5 关闭业务服务器

只需关闭 JAVA 示例便可关闭业务服务器。

注：运行业务服务器，“./runbusinessserver.sh”就是一个简单的业务服务器示例，采用 Java 语言开发，源代码在 SDK 包的【src\server\AnyChatBussinessServer】目录下，可供给用户学习或测试用，在实际应用过程中，需要独立开发自己业务相关的业务服务器；

2.6 配置开机自启动

Linux 是通过脚本来启动服务或应用程序的，通常在开机过程中均会自动执行“rc.local”脚本，所以我们可以将核心服务器、业务服务器的运行指令添加到该脚本文件中即可实现开机自启动。

为了让操作流程清晰易懂，在此将 SDK 包目录下的“bin”文件夹移出到“home”路径下。（此步可不操作，不会影响配置流程，但要注意配置时路径的填写），如图 2-6 所示：

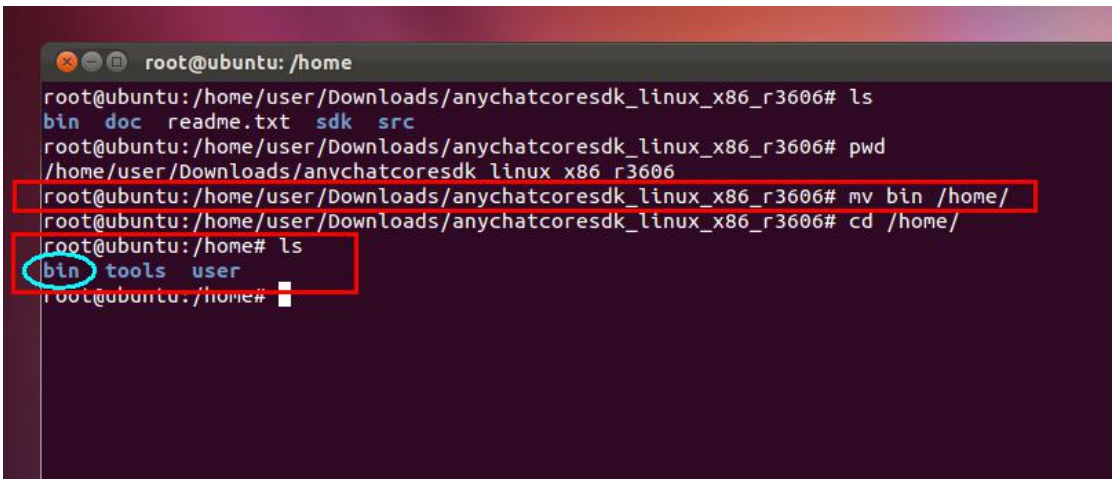


图 2-6

2.6.1 添加自启动脚本

编辑 “/etc/init.d/”目录下的“rc.local”文件（CentOS，RHEL，Fedora 的 rc.local 是在/etc/rc.d/目录下。Ubuntu，Debian 的 rc.local 是在/etc/init.d/目录下），添加如下脚本并保存退出编辑文件，如图 2-7 所示：

```
#anychatcoreserver
export LD_LIBRARY_PATH=/home/bin/server:$LD_LIBRARY_PATH
/home/bin/server/anychatcoreserver -d  > /dev/null &

#anychatsampleserver
#JAVA_PATH
export LD_LIBRARY_PATH=/home/bin/serversdk:$LD_LIBRARY_PATH
export PATH=/home/tools/jdk1.7.0_25/bin:$PATH
java -Dfile.encoding=UTF-8 -jar /home/bin/serversdk/anychatsampleserver.jar > /dev/null &
```

(注意：如果没有移动 SDK 包里的 bin 文件夹到“/home”目录下，请修改上面的路径指到 bin。这里的 Java 环境为编写文档用，用户需配置自己的 Java 环境)

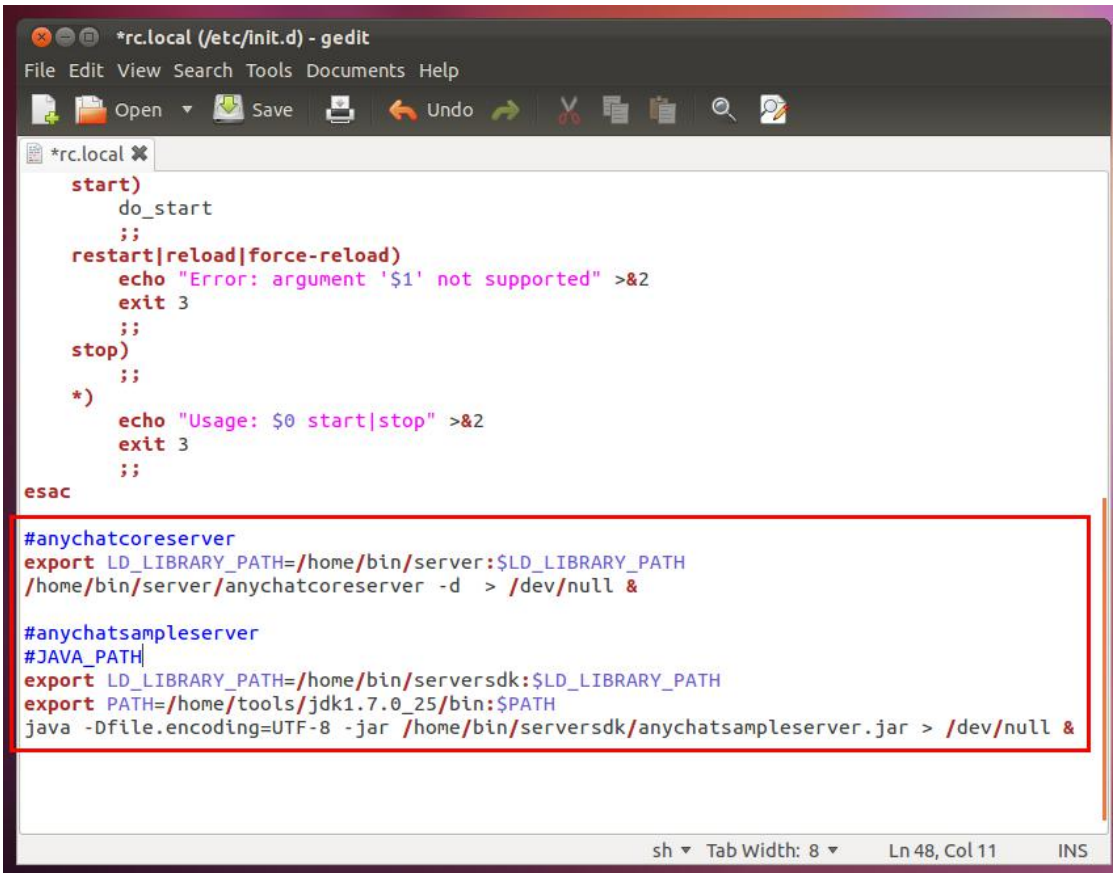


图 2-7

（设置 anychatcoreserver 和 anychatsampleserver 的库链接，JAVA 环境。指定开机启动项目路径）

2.6.2 查看开机启动

编辑并保存好“rc.local”文件之后，重启系统，查看进程：

分别查看服务器与业务服务器，如图 2-8 所示：

ps –ef | grep anychatcoreserver

ps –ef | grep anychatsampleserver

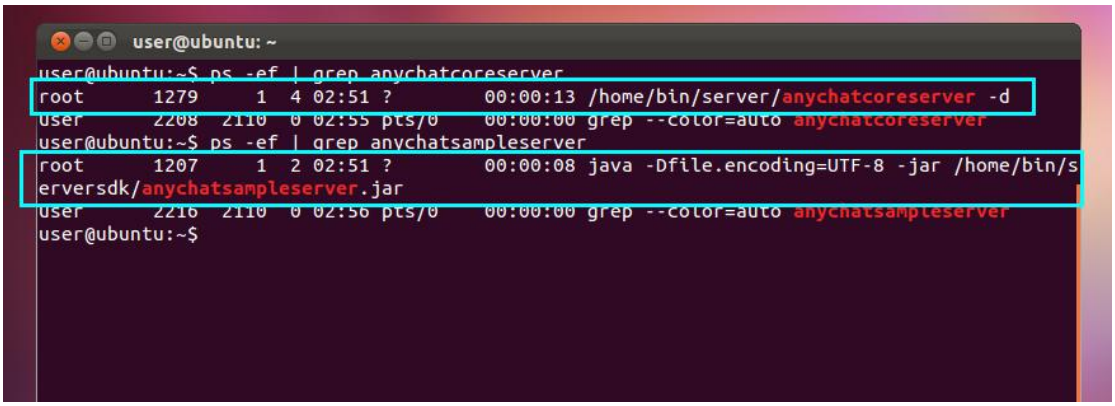


图 2-8

可以看到“anychatcoreserver”和“anychatsampleserver”服务已在运行。

如需关闭进程，使用命令“kill”杀掉指定进程号即可。如需关闭开机启动服务器则编辑“rc.local”文件，将之前所添加的内容删去或者注释就可。

有关设置开机自启动更详细的流程可参考论坛详细说明：Linux 开机启动 AnyChat 服务器配置流程

2.7 分布式部署

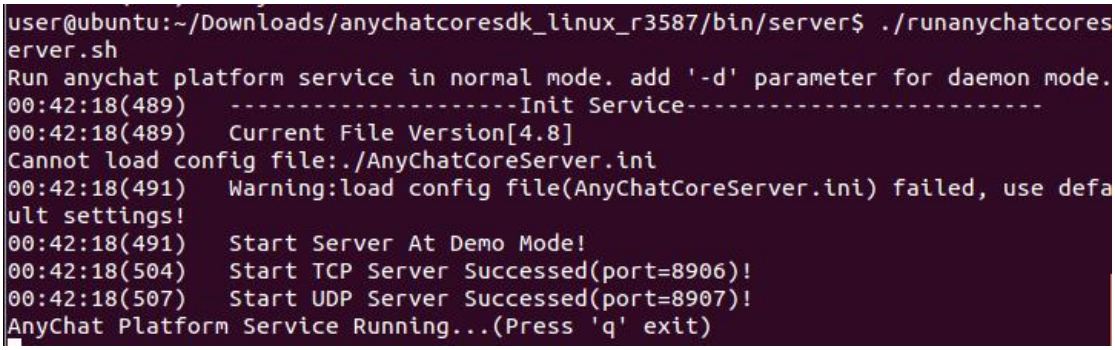
AnyChat 核心服务器与业务服务器可以分布式部署，分别运行于不同的服务器上，也可以运行于不同的操作系统上。默认配置下，核心服务器、业务服务器运行于同一台服务器上，若要实现分布式部署，需要修改配置文件（AnyChatServerSDK.ini），将默认的配置参数（127.0.0.1）修改为核心服务器的 IP 地址（修改完成后需要重启业务服务器），如下：

```
[Base Settings]
ServerIpAddr=192.168.1.25
```

注：分布式部署需要“企业版”服务器才能支持。

三、核心服务器的配置

核心服务器通过读取服务器程序所在目录下的 AnyChatCoreServer.ini 配置文件进行初始化配置。当系统第一次启动时，如果该配置文件不存在，则服务器所开启的终端界面将会报出错误，并且自动用默认设置。如图 3-1 所示：



3-1 读取文件.ini 失败

3.1 配置文件内容

```
[Base Settings]
ServerID=1
TCPListenPort=8906
UDPServicePort=8907
SDKAuthPass=
DemoMode=1

[P2P Settings]
RoutingPolicy=2
RoutingTTL=3
RoutingMTN=4

[Video Settings]
VideoWidth=320
VideoHeight=240
VideoFPS=15
VideoGOPSize=30
VideoBitrate=90000
VideoCodec=1
VideoQuality=3
```

```
[Audio Settings]
AudioSamplesPerSec=16000
AudioChannels=1
AudioBitsPerSample=16
AudioCodec=11
AudioBitrate=15850

[Function Settings]
CloseFrontLink=1
AutoAddRoom=1
SDKFilterPlus=libipcfilterplus.so
AllowGuestLogin=1
MaxUsersPerRoom=100
VideoCallStartRoomId=1
NetCheckTime=5000
NetTimeoutTime=20000
TempFilePath=

[Standby Server Settings]
SlaveMode=0
MasterServerIP=
MasterServerUDPPort=8907

[Debug Settings]
AutoUploadLogInfo=0
SaveLogRootDir=
OutputThreadStatus=0
OutputRecordTask=0
```

3.2 基础信息配置

ServerID: 是指服务器的编号，备用，默认为 1。

TCPListenPort: 指定服务器的 TCP 服务端口，默认为 8906。

UDPServicePort: 指定服务器的 UDP 服务端口，默认为 8907。

SDKAuthPass: SDK 连接服务器认证密码（字符串，不超过 64 字节），默认为空，未设置，如果设置了密码，则 SDK 必须在连接服务器之前，调用“BRAC_SetServerAuthPass”传入有效的密码，传入的密码将会被加密后传输到服务器验证，如果验证失败，则连接将被关闭，在实际使用时，建议设置一个密码，能有效的保护服务器，当启用双机热备模式时，如果设置了认证密码，则主

服务器、从服务器的认证密码必须相同。

DemoMode： 是否启动演示模式，1 为是，0 为否。（演示模式内置 2 个房间，房间号为 1、2），默认为 1。

3.3 网络传输配置

RoutingPolicy： P2P 数据路由传输策略：

1 所有数据传输均由服务器转发（禁止 P2P）

2 NAT 穿透成功的用户之间互传，穿透失败的用户由服务器转发，该策略适用于局域网使用；（默认）

3 P2P 网格传输，由服务器以房间为单位，根据当前房间各用户 NAT 穿透情况、网络带宽等情况，动态生成一份网格路由表，流媒体数据按路由表传输，该策略适用于互联网使用。（目前还处于测试阶段）

关于 P2P 数据传输策略的详细信息，可参考在线文档：

http://www.anychat.cn/faq/index.php?action=artikel&cat=1&id=180&artlang=zh

RoutingTTL： 流媒体数据经过网格传输时的最大生存期，每经过一个中间用户转发，生成期减一，为 0 将不再继续转发，该配置项当 RoutingPolicy=3 时有效，设置较大时，会较大的减轻服务器带宽压力，但会带来末端用户接收数据延迟时间增加的后果。

RoutingMTN： 流媒体数据经过网格传输时，中间节点最大包转发路数（通俗的讲，就是任意节点最多可服务的后续节点个数，如当设置为 4 时，表示任意用户可向其它 4 个用户转发数据），该配置项当 RoutingPolicy=3 时有效，设置较大时，会较大的减轻服务器带宽压力，但会占用用户较多的出口带宽（因为需要向其它用户转发数据），当用户本身网络状况较差时，会影响后续用户的接收效果。

3.4 视频参数配置

VideoWidth： 视频分辨率宽度

VideoHeight： 视频分辨率高度

VideoFPS： 视频的采样帧率：1…25，默认为 15。

VideoGOPSize： 视频关键帧间隔，默认：40。

VideoBitrate： 视频编码的目标码率：60000（单位：bps）。

VideoCodec： 视频编码器（H.264=1）。

VideoQuality： 1=较差质量；2=一般质量；3=中等质量（默认）；4=较好质量； 5=最好质量（视频质量需要与视频目标码率相配合，较高的视频码率使用较高的视频质量，较低的视频码率建议采用较低的视频质量参数，在局域网内可以采用“5 最好质量”，其它网络环境建议采用默认值）。

VideoPreset： 视频预设参数，取值范围为 1…5，默认为 3，主要用来控制 CPU 资源占用率和画面细节，值越小，编码时的 CPU 占用率越低，但会损失对画面细节的处理；值越大，编码时的 CPU 占用率越高，但对画面细节的处理会更细腻。

服务器所配置的视频参数为默认参数，当客户端通过 API 接口修改本地相关参数后，将以本地参数为准。

3.5 音频参数配置

AudioSamplesPerSec：音频采样频率：8000Hz； 11025Hz； 16000Hz；22050Hz； 44100Hz，默认 16000Hz。

AudioChannels： 音频通道数：1－2，默认 1。

AudioBitsPerSample：音频量化位数：8，16，24，默认 16。

AudioCodec=11： 音频编码器 ID

AudioBitrate=15850：音频编码的目标码率（单位 bps）

（注：音频参数与编码器有密切的关系，所设置参数不符合编码器的相关规范，将会导致客户端无法打开 Mic，有任何疑问，请与佰锐科技技术支持联系。）

s3.6 功能参数配置

CloseFrontLink：用户登录时，是否断开之前的连接。如果用户登录多次，则系统只会接受其中一个连接，为 1 时，最后一个连接被接受，前面的连接被断

开，为 0（默认）时，第一个连接被接受，后面的所有连接均返回错误代码。

SDKFilterPlus：服务器 SDK 增强插件文件名，默认为空，表示不加载 SDK 插件，如果此处配置了文件名，则服务器在启动时，将会自动加载该动态库。该项配置可以为独立的文件名（该文件必须在当前目录下），也可以为带完整路径的文件名。有关服务器 SDK 插件的详细信息可参考《AnyChat SDK Filter Plus 开发指南》。

AllowGuestLogin：是否允许游客（空密码用户）登录系统，默认为 1 表示允许，服务器将自动为该用户分配 userid，如果不希望空密码用户连接服务器，可设置为 0 禁止游客登录。（注：若服务器工作于演示模式，则该配置项将被忽略）

MaxUsersPerRoom：单个房间最大用户数，更加为 100，表示支持 100 个用户同时在一个房间中。房间内用户数越大，所占用的内存资源也就越大。

VideoCallStartRoomId：视频呼叫起始房间号，默认为 1，表示从 1 开始往后顺序分配，如果需要预留一些房间给其它功能使用，避免和视频呼叫冲突，则可以修改该参数。若设置为 0，则表示从 1 开始顺序分配，一直累加，不会分配已经使用的房间号。

NetCheckTime：网络连接检测周期，单位：毫秒，默认为 5000，表示每 5 秒发送一次心跳包来检测网络状态，保持网络连接。

NetTimeoutTime：网络连接超时时间，单位：毫秒，默认为 20000，表示 20 秒没有收到心跳包即说明网络连接被中断，将触发客户端的 OnLinkClose 事件。

TempFilePath：临时文件保存路径，默认为空，客户端上传的文件保存在核心服务器当前目录的 temp 子目录下。可配置为完整的保存路径，配置路径不存在时会自动创建。

3.7 双机热备参数配置

[Standby Server Settings]

SlaveMode=0 是否为从服务器模式，0（默认）表示为主服务器模式，1 表示为从服务器模式

MasterServerIP 主服务器 IP 地址

MasterServerUDPPort 主服务器 UDP 通信端口，默认为：8907

更多信息可参考：AnyChat 服务器双机热备解决方案

3.8 功能调试参数配置

[Debug Settings]

AutoUploadLogInfo=0 是否开启服务器自动收集客户端日志功能，0（默认）表示关闭， 1 表示打开；

SaveLogRootDir= 保存客户端日志的根目录，若为空，则默认保存在服务器应用程序的“clientlog”子目录下；

OutputThreadStatus=0 是否输出核心服务器的线程状态数据，0（默认）表示关闭， 1 表示打开，通常在需要对核心服务器进行状态监测时可以打开；

OutputRecordTask=0 是否输出服务器录像任务调试信息，0（默认）表示关闭， 1 表示打开，通常在需要对中心录像功能进行调试时可以打开；

更多信息可参考：实现服务器集中收集客户端日志信息功能

四、服务器的授权

如果您从佰锐科技获取到了授权证书（AnyChat.cer），请将授权证书拷贝到核心服务器目录下，不能更改文件名，同时修改核心服务器配置文件（AnyChatCoreServer.ini）将 DemoMode 设置为 0，让服务器程序工作在授权模式下，然后重启核心服务器程序即可。

AnyChat 服务器支持多种授权绑定方式，目前常用的绑定方式有：

4.1 绑定硬件特征码

授权证书绑定一台特定的服务器，服务器程序只能部署在该服务器上。使用该方式时，佰锐科技会提供一个获取硬件特征码的工具软件，将获取的特征码提交给佰锐科技，佰锐科技便会依据所获取的特征码生成授权证书文件。

当服务器部分硬件损坏后（如更换硬盘、主板等），佰锐科技将提供授权证书更新服务，保障系统的正常使用。如果服务器完全损坏，则佰锐科技不提供授权证书更新服务。

该方式适合于不方便使用绑定域名，或是绑定 UKey 的场合，如有些政府平台对网络的安全性要求较高，禁用了服务器的所有 USB 端口，无法使用 UKey，而且是内部网络，不能进行域名的解析，使用绑定硬件特征码的方式就比较合适。

4.2 绑定域名

授权证书绑定一个特定的域名（URL 地址），服务器程序可以运行在该域名所解析的 IP 地址对应的服务器上。通常会绑定一个二级域名，如：video.anychat.cn，当需要更换服务器时，只需要重新解析绑定的域名到新服务器的 IP 地址上即可，用户可自行完成，不需要佰锐科技的参与。

如果是互联网类型的应用，且服务器是部署在互联网上，如视频游戏平台、视频聊天室、远程教育平台等，我们建议采用该绑定方式。

4.3 绑定 UKey

授权证书绑定一个 UKey（俗称加密狗，USB 接口），只要这个 UKey 插入任意一台计算机上，就可以正常运行服务器程序，当需要更换服务器时，只需要把 UKey 从原来的计算机上拔下来，插入到新的计算机上即可。

该方式适合服务器不固定，可能会随时进行调整，且方便对服务器进行管理的场合，通常工程项目类应用可采用该绑定方式。

若采用 UKey 授权方式，请将 UKey 插入计算机的 USB 接口上，然后进入 SDK 包的：【bin\server】下，并用 root 权限执行脚本：./ installukey.sh，然后重启计算机，便可在普通用户权限下识别 UKey 设备。

五、常见问题与解决方案

Q：如何开启 AnyChat 服务器，开启程序在哪儿？

A：AnyChat 服务器程序随 SDK 包一同发布，在 SDK 包的：bin\Server 目录下。

Q：如何关闭 AnyChat 服务器程序？

A：有三种方式来关闭 AnyChat 服务器程序：

1、在服务器开启的终端上按‘q’键即可关闭服务器。

2、直接关闭服务器运行所在的终端。

3、如果服务器为后台运行，则使用命令查询服务器进程，用“kill”命令杀掉服务器对应的进程号即可。

Q：如何关闭业务服务器程序？

A：有三种方式来关闭业务服务器：

1、在业务服务器开启的终端上按‘q’键即可关闭服务器。

2、直接关闭服务器运行所在的终端或示例。

3、查询业务服务器对应的进程号，用‘kill’命令杀掉业务服务器进程即可。（注：杀掉业务服器后，新连接用户不可进入房间，但已连接用户不会受到影响，可继续使用）

Q：如何卸载 AnyChat 服务器程序？

A：AnyChat 服务器程序是绿色程序，不需要安装，不会向操作系统释放额外的文件，也不会写入额外的信息到系统配置文件中，所以不需要使用 AnyChat 服务器时，直接把 AnyChat 服务器程序所在目录全部删除即可。

Q：如何确认 AnyChat 服务器启动成功，运行正常？

A：AnyChat 服务器程序启动之后，会在当前目录下的日志文件（AnyChatCoreServer.log）中输出相关的运行状态信息，如果日志中没有启动失

败的提示，则说明服务器已正常运行，更多信息可参考：如何排查 AnyChat 故障信息？

Q：在启动 AnyChat 服务时，出现如下图所示的提示，启动服务器有错误信息，如图 5-1 所示：是什么原因？

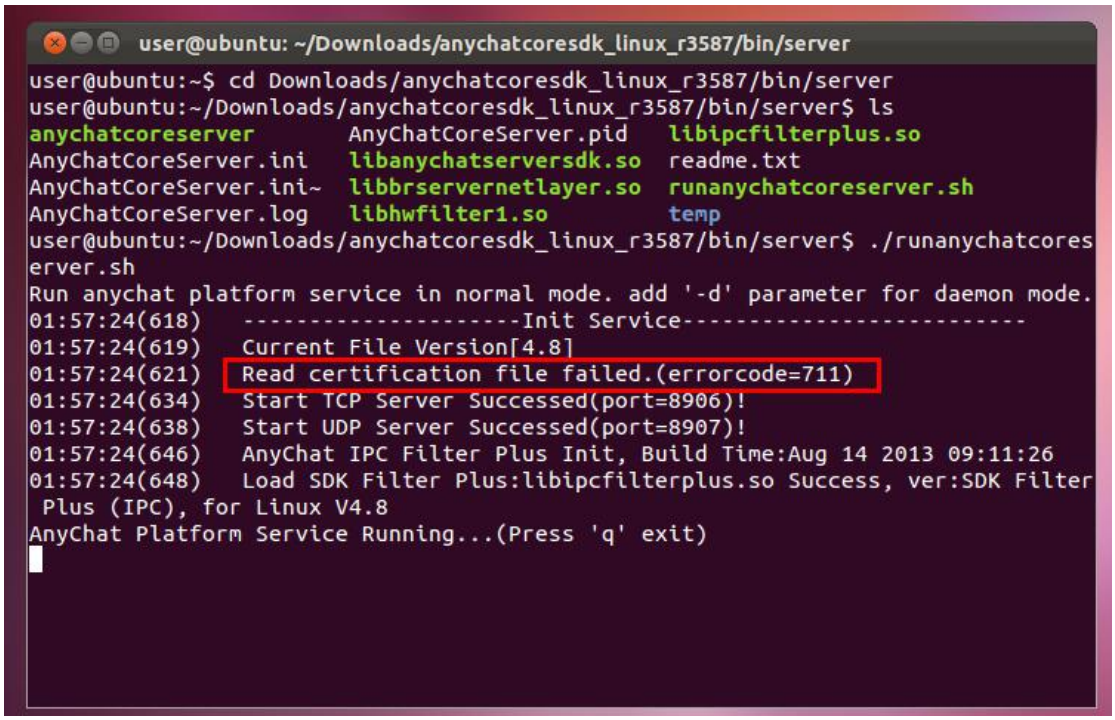


图 5-1 服务器开启错误信息

A：出现这种情况通常有两种可能性：

1、授权失败，在使用官方 Demo 版本进行学习或者测试的时候，打开服务器所在目录【./bin/server】，找到并打开“AnyChatCoreServer.ini”文件，将[Base Settings]下的属性 DemoMode 值改为 1 即可；

2、授权失败，如果授权信息不匹配，则服务器不会正常启动，可查看 AnyChatCoreServer.log 日志，里面会有对应的错误代码输出，出现该问题时，可将 AnyChatCoreServer.log 发给佰锐科技技术支持人员，请求技术支持。

Q：服务器可以成功打开，但业务服务器打开不成功是什么原因？

A：产生这种情况的可能性有：

1、Java 版本过低，升级 java 版本。

2、GCC 版本过低，升级 GCC 版本。

如果还是不能解决问题，请将问题反映给佰锐科技技术支持人员，请求技术支持。或将问题提交到官方论坛：AnyChat 论坛 Linux 专区

Q：在有些企业内网，或是政府专网中无法连接部署在互联网上的服务器，是什么原因？

A：在部分网络环境下，为了安全，对外只开了 80、25、110 等少数几个端口，只允许内部计算机上网、收发邮件，过滤其它端口的网络连接，这时 AnyChat 客户端就可能与外网的服务器连接不上了。

AnyChat 服务器默认是工作在 8906（TCP）、8907（UDP）两个端口上（可以在服务器的.ini 配置文件里面修改），如果内部网络有防火墙，就需要在防火墙里面设置一下，把这两个端口打开。

如果是连接不上服务器，通常是 TCP 端口关闭了；

Q：能连接上服务器，可以进入房间，也可以发文字消息，但是互相看不到视频是什么原因？

A：通常是 UDP：8907 端口被关闭，或是在防火墙上没有打开导致流媒体通信受阻，而网络连接、文字消息是走 TCP 通道，所以不会有影响。

可打开客户端日志文件（BRAnyChatCore.log），看里面是否有如下的记录：Warning: The UDP communication with the server abnormal!，如果有则说明与服务器的 UDP 通信存在故障。

另外还可以使用我们提供的专用工具软件来检测服务器的 UDP 通信状态，请参考：http://bbs.anychat.cn/forum.php?mod=viewthread&tid=9

Q：可以连接上服务器并进入房间，互相看视频时很卡，不流畅，可能是什么原因？

A：请检查客户端以及服务器的网络带宽是否足够，以及服务器的网络状况是否稳定，音视频数据流是采用 UDP 通信协议，所以我们提供了一个网络质量评估工具可实时检测服务器的网络状况，连续发送 UDP 数据包，客户端通过接收服

务器返回的 UDP 数据包情况来评估网络质量，详情可参考：网络质量评估工具工作原理与使用指南 。

六、技术支持

在您使用 AnyChat SDK 的过程中，遇到任何困难，请与我们联系，我们将热忱为您提供帮助。

您可以通过如下方式与我们取得联系：

1、在线论坛：http://bbs.anychat.cn/

2、知识中心：http://www.anychat.cn/faq/

3、官方网站：http://www.anychat.cn

4、电子邮件：service@bairuitech.com

5、24 小时客服电话：+86 （020） 85276986、38109065、38103410