# 1.P2P原理及框架

## 1.1 P2P原理

P2P(Peer to Peer)即对等网络，简单理解成处于P2P网络中的成员通过相互间直接连接达到资源和服务的共享。相比于传统的C/S框架（如图1），避免了所用的客户端都只向服务器请求数据，在P2P模式(如图2)下，客户端即充当客户端又充当服务器角色，把自己从服务器请求获取的数据分享给其它客户端，将服务器从重负荷、高带宽的重担中解救出来，把一部分压力分散到各个客户端。

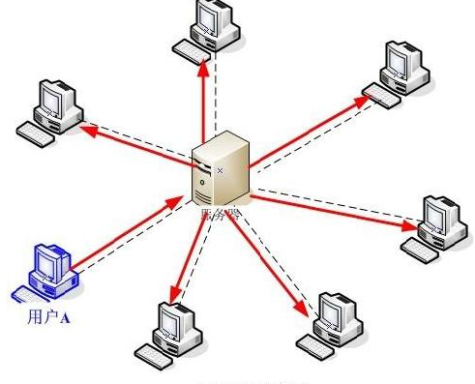
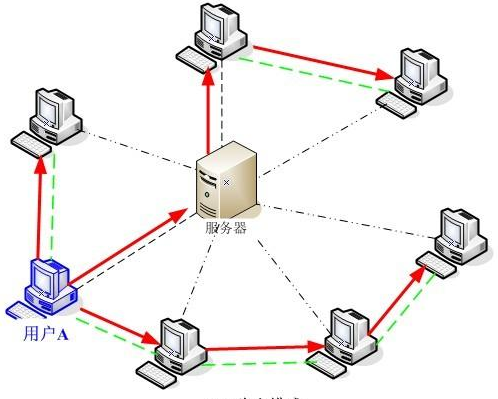
 

图1.传统C/S模式 图2.P2P模式

## 1.2 P2P框架

本P2P框架如图3所示，分为四部分，Tracker Server，Stun server和vodpeer，下面简单介绍：

1）Tracker Server负责管理peer节点；

2）Stun server负责打洞穿透的服务器，实际包括TccSVR和Stun(下面没有特殊说明时所说的Stun server即包括两者)。TccSVR负责局域网的互发现，Stun负责跨局域网穿透。实际部署时，TccSVR和Stun实际部署时同时部署在两台服务器上，使用两台服务器的目的是探测nat类型，辅助洞；

3）vodpeer，即所说的peer端，实际运行在手机端或盒子端的程序（JNI）；

4）web，peer从webserver读取配置文件进行初始化。

图3 P2P框架图

① peer登陆tracker，向tracker上报自己的信息；

② peer登陆stun1服务器，探测nat类型，stun负责辅助打洞

③ stun2辅助stun1辅助vodpeer探测nat类型，辅助打洞；

④ stun1与stun2互相发现对方，辅助peer探测nat类型；

⑤ peer之间打洞，通信分享数据；

⑥ vodpeer从webserver读取配置文件；

# 2.P2P代码编译

## 2.1 编译环境

编译环境要示使用Linux 64系统，安装gcc及相关库，安装android ndk。为方便编译采用编写编译环境脚本"p2penv.rc"，只需进行简单的路径修改，就可以方便的进行编译，具体见脚本。

下载ndk安装包“android-ndk-r9c-linux-x86\_64.tar.bz2”并解压，根据自己服务器的实际情配置编译环境脚本"p2penv.rc"，示例如下：

ANDROID\_NDK\_PATH=~/Work/P2P/android-ndk-r9c

P2P\_MAIN\_PATH=~/Work/P2P/P2P-Source

ANDROID\_NDK\_PATH指向解压后的ndk路径，P2P\_MAIN\_PATH指向解压后的P2P代码（这里把P2P源码包解压到”P2P-Source/vkkp2p“目录）；

## 2.2.编译过程

编译时先执行行脚本（执行一次即可，请先看2.4的注意事项），”$source p2penv.rc“

1) 编译Tracker Server

$makets

2) 编译TccSVR server

$maketcc

3) 编译Stun Server

$makestun

4) 编译vodpeer

$makevodpeer

## 2.3 编译结果

编译后的结果保存在"p2pout"目录，使用"$gout"即可进入编译后生成物的目录。

## 2.4 注意事项

1) P2P源码编译stun时有错误，原因是Makefile链接有问题，修改" vkkp2p/comm/linux/uacstun "目录下的Makefile如下：

$(CPP) $(LIBS) -o $@ $^ => $(CPP) -o $@ $^ $(LIBS)

2)编译过程中若出现部分软件没有安装的话，比如make或者g++，ubuntu请使用”sudo apt-get install xxx“安装；

3)为让peer端从webserver读取配置文件，修改源码如下：

**"vkkp2p/vkkp2p/android/vodpeer/jni/Android.mk"**修改如下：

LOCAL\_CPPFLAGS += -Wall -Werror -DNDEBUG -DLINUX32 -D\_FILE\_OFFSET\_BITS=64 -D\_ANDROID -DSM\_DBG

**" vkkp2p/vkkp2p/src/vodpee/Main.cpp"**修改如下：

#ifdef SM\_DBG

sprintf(buf, "%s", argv[i+1]);

#else

sprintf(buf, "http://media.tvata.com/services2/%s/config.php", argv[i+1] );

#endif

# 3.P2P服务器部署

## 3.1 外网服务器

使用两台外网服务器即可，一台用作Tracker Server和stun server（下面称外网服务器1），另一台用作stun server（外网服务器2）。为了使peer端灵活的读取配置文件，需要配置webserver，见3.4。使用SSH客户端登陆外网服务器进行部署。

### 3.1.1 服务器配置

两台linux系统（Ubuntu 14.04即可，与版本无关）的服务器，服务器要求具有公网IP，64bit系统，内存及CPU核数没有具体要求，尽量配置好些，实际服务器信息如下：

1）外网服务器1用作Tracker和stun server

账户： tvfan

密码：

ip: 103.244.165.201

ssh端口号: 22

2） 外网服务器2用作stun server

账户： tvfan

密码：

ip: 103.244.165.202

ssh端口号: 22

### 3.1.2 部署前准备

1) 将编译生成物”trackerbin“和”stunbin“目录copy到外网服务器1，将”stunbin“目录copy到服务器2，再根据下面情况配置部署。

2) 服务器1开通端口4000,4001,4002,4004,4005，服务器2开通4006,4007端口的防火墙

$sudo ufw allow xxxx

## 3.2 Tracker Server部署

1 配置文件"tracker\_config.ini"如下：

[public]

tracker\_id=0

tracker\_ip=103.244.165.201 //tracker服务器的ip

tracker\_port=4001 //tracker的port，应该是TCP

http\_port=4002 //tracker的httpsever的port

[search]

rsp\_server\_num=5

rsp\_center\_num=2

rsp\_super\_num=2

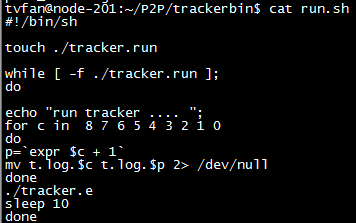
vip\_super\_minnum=0

2 运行服务器

$screen run.sh &

Note:使用 "screen ./tracker.e"可以让服务与当前的ssh的session脱离

run.sh脚本内容如下：



3.查新Trackerserver状态

1) 查询tracker状态接口:

http:// 103.244.165.201:4002/tracker/tracker\_info.do

2) 查询所有sn服务器信息接口:

http:// 103.244.165.201:4002/tracker/server\_info.do

3) 查询某个playlist节目的信息接口：

http:// 103.244.165.201:4002/tracker/playlist\_info.do?url=http://media.tvata.com/streams\_123/iptv/d/kkdh/playlist.m3u8

Note："http://media.tvata.com/streams\_123/iptv/d/kkdh/playlist.m3u8"是要查询的节目的链接

4) 查询所有共享文件信息接口:

http:// 103.244.165.201:4002/tracker/allfile\_info.do

## 3.3 Stun server部署

### 3.3.1 服务器1部署

服务器1(103.244.165.202)的配置文件“uacstun.ini”如下（与Tracker Server使用同一服务器)：

[stunsvr]

accept\_ip=0.0.0.0

accept\_port1=4004 //accept\_port1用作TCP连接性测试端口

accept\_port2=4005 //accept\_port2用作HTTP接口端口

stunB\_ip= 103.244.165.201

stunB\_port1=4006

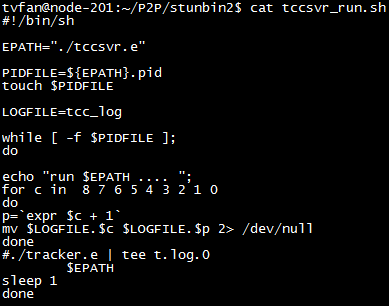
stunB\_port2=4007

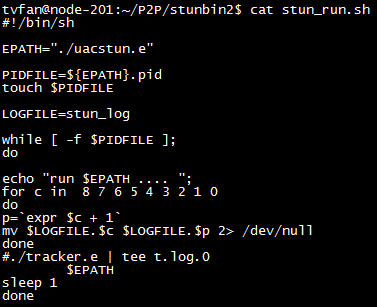
运行服务器

$screen ./ tccsvr\_run.sh &

$screen ./ stun\_run.sh &

Note：一个终端窗口只能运行一个screen指令





### 3.3.2 服务器2部署

服务器2 (103.244.165.201)的配置文件“uacstun.ini”如下：

[stunsvr]

accept\_ip=0.0.0.0

accept\_port1=4006 //accept\_port1用作TCP连接性测试端口

accept\_port2=4007 //accept\_port2用作HTTP接口端口

stunB\_ip=103.244.165.202

stunB\_port1=4004

stunB\_port2=4005

运行服务器

$screen ./tccsvr.e

$screen ./uacstun.e

**Note**:服务器1与服务器2的stun\_B的ip及port互指对方

## 3.4 webServer部署

为方便peer端读取配置文件，根据前面配置的Tracker及Stun server情况，对应修改vodpper配置文件"peerconfig.ini"的Tracker和Stun的配置项，修改使用静态文件的格式：http://103.244.165.201:4000/p2p/webvodpeer.ini

webserver网络配置文件位置:

IP：103.244.165.201

配置文件位置：/server/tomcat/tomcat\_p2p/webapps/ROOT/p2p

## 3.5 手机端的部署

手机端只需使用编译后的“vkkp2p\_android.e”替换之前的，并将P2PManager.java中“-zone”后的参数修改为“http:// 103.244.165.201:4000/p2p/webvodpeer.ini”，重新生成jar包，导出apk即可。