

逻辑主文档

所有包的格式以及相关规定

以下称客户端为client，服务器端为server

游戏正式开始前（房间阶段）

client向server传递包格式规定

以下roomnumber表示所在房间号，请注意每个包都必须带有roomnumber

- join roomnumber

申请加入roomnumber号房间，此处roomnumber为用户自行键入

- leave roomnumber

离开roomnumber房间

- build roomnumber

xx xx xx xx xx xx xx

房主（收到Owner包之人）向服务器确认这局游戏的基础设置。每个xx为一个非负整数，依次表示总人数，狼人数目，预言家数目，女巫数目，守卫数目，猎人数目，警长数目（0/1）

- up roomnumber

举手（准备好了）

- down roomnumber

放下手（从准备状态变成未准备状态，请client保证先前是ready状态）

- start roomnumber

房主决定开始游戏

- worldchat roomnumber

xxxxxxxxxx@username

xxxxxxxxxx为用户发言内容，username为此用户ID，这是世界频道发言，所有登录的用户均可收到

- roomchat roomnumber

xxxxxxxxxx@username

格式同上，只有本房间用户可以收到

server向client传递包格式规定

- Success

- 表示你成功加入该房间
- Full
 - 房间已满，不能加入
- Owner
 - 收到此包的你即为房主，可以指定游戏设置（人数等）
- OwnerNew
 - 原房主跑路了，你现在是房主，不可以重新设置规则，但你负责按开始按钮
- Startbutton 0/1
 - 此包只发给现任房主，参数为0表示还不能开始游戏，为1表示可以开始游戏（激活开始按钮）
- Status


```
xx xx xx xx xx xx xx
username1 0/1
username2 0/1
...
```

 - 此包按节奏对房间内所有人广播，第二行xx即为房间游戏设置，格式同client向server的build包中指定的格式。
 - 以下n行（n表示房间内现有玩家个数）每行格式为username 0/1, username为玩家ID, 0/1表示是否举手
- Gamestarted
 - 对房间内所有人广播，表示游戏正式开始。
- worldchat和roomchat
 - 为client向server的发言包的原样转发，请client按上述规定做相应显示

游戏正式开始后(Game阶段)

此时client界面需要显示的包括Status包传递的信息（多少人活着）等，此外关键需要控制开闭的就是两个地方。

一个是一个投票按钮，功能为指定一人（向server发送一个vote包）（此按钮为一次性，每次收到Set包指示打开后只能选择一次），这个选择按钮的含义会有很多变化，具体在游戏流程中解释。

第二个是聊天框，开闭严格受server传来的Set包控制，对应为向server发送chat包（room/wolf）。

server向client传递包格式规定

- Status


```
username role 1/0 1/0
username 1/0 1/0
...
```

报告当前游戏状态（收到此包刷新一次游戏状态显示），第二行为自己的username，role为自己的游戏角色，两个1/0依次为是否活着，是否是警长
游戏角色代号为villager,werewolf,prophet,witch,defender,hunter
第三行到最后各行为一个用户的状态报告（也会包括自己的，顺序固定），1/0含义相同

- Set

0/1 0/1 channel //能否投票，能否说话，什么频道（Room/Wolf）

设置client允许的行为，第二行两个1/0依次为投票按钮是否打开，聊天框是否能发送信息
channel为一个字符串（Room/Wolf），表示当前发言在什么频道（Room频道全房间皆可见，Wolf只有狼人伙伴可见）

- Day xx

表示进入了第xx个白天

- Night xx

表示进入了第xx个夜晚

- Die

@username
@username
...

通知大家谁死了，以下每行一个死掉的玩家的ID，若没人死则只有Die一行

- Guard

发给守卫，告诉守卫应该选择守护谁了

- Witch

@username

此包有两个作用，如果第二行有@+用户名，表示通知女巫username被杀死了，请反馈是否救他。如果没有第二行，则表示你可以选择毒一个人

- Prophet

发给预言家，表示你可以选择验一个人了

- Good/Bad

发给预言家，Good表示验的人是好身份，Bad表示是狼人

- Choosecap

开始警长竞选（第一日），此后会按顺序要求每个人说一句话

- Captain

@username

广播包，username成为了警长（被选举或者上任警长死后传递了警徽），如果警徽被撕了，则@后面会留空

- Order

发给警长，表示你可以选择一个发言顺序

- Vote

开始白天的投票，此后会有Set包

- Equal

@username1

@username2

...

这几个人平票，将开始下一轮投票，只能从这几个人里面选（请client实现这个逻辑）

- Youdie

表示你死了

- Tellsth

你有机会留下一句遗言

- Carry

你是猎人且你死了，可以选择带走一个人，如果没有带人的机会则不会收到这个包

- Hunter

@username1

@username2

一个广播的包，含义为username1是猎人而且死了，选择了带走username2，如果猎人放弃带走人则不会广播此包

- Pass

你是警长且你死了，可以选择传递警徽，如果没有机会则不会收到这个包

- Captaindie

广播通知，警长死了

- GameEnded

广播，游戏结束了

- YouWin/YouLose

你赢了/输了

- Endstatus

username role 0/1 0/1

结束时的状态，每个人的ID，角色，活着/死了，警长/非警长

client向server传递包格式规定

- roomchat/wolfchat roomnumber

格式同上，房间/狼人频道发言，同样会被原样转发到client，游戏开始后不能用世界频道

- vote roomnumber
@username

投票/选择某人的包，username为选定的人的ID，这个包功能很多，可以是白天投票，可以是狼人决定杀人，女巫决定救/杀人，守卫决定守人，预言家验人，甚至可以是警长定方向，具体见下面的流程
如果@后面留空，表示这个人弃票/放弃带走人/撕毁警徽，

游戏流程简述

描述一遍各包交换流程，注意除了join包和自由聊天的包以外，client的每个包都是响应server的某个包而发出的，不要没有收到指令包就主动发包。

->表示client向server

<-表示server向client

以下各包采用简写，只保留包头或者省略换行，格式严格按上面的来。

游戏开始前

->join包

<-Success/Full包

如果是建房者：

```
<-Owner包  
->build包
```

如果你变成了房主

```
<-OwnerNew
```

->up/down

<-Status(广播)

<-Startbutton 0/1(只向房主)

->start

<-Gamestarted

正式游戏阶段

所有规定时间的过程（用括号标出），请client数够足够的时间后，如果用户没有输入，就按应返回的格式返回空包

夜晚

<-Night包，广播

<-Status包

<- Set 0 0

对守卫(10s)：请client判断不能连续守一个人

```
<- Guard包
<- Set 1 0
-> vote包, 指定选谁 (@后面留空表示弃权)
<- Set 0 0
```

狼人杀人阶段(30s):期间随时可以投票, 投了不能改, 随时可以在狼人频道交谈, 以下均对活着的狼人, 死了的只能收到wolfchat, 啥也不能干

```
<- Set 1 1 Wolf
-> vote (每个狼人都要返回)
-> wolfchat(随便发)
<- Set 0 0 Room
```

女巫阶段1(10s):选择救不救人, client做判断只有初夜可以自救,只有女巫活着且解药没用才会如下操作

```
<- Witch @diedusername
<- Set 1 0
-> vote @ 如果救人就在@后面带上@diedusername,不救人就在@后面留空
<- Set 0 0
```

女巫阶段2(10s):选择毒谁, 只有女巫活着且毒药没用才会如下操作

```
<- Witch
<- Set 1 0
-> vote @ 如果毒人就带上username, 否则@后面留空
<- Set 0 0
```

预言家阶段(10s):

```
<-Prophet
<-Set 1 0
->vote @username 同上
<-Set 0 0
<- Good/Bad
```

白天

```
<- Day
<- Set 0 0
<- Die 昨晚谁死了
```

如果你是猎人, 昨晚死了, 不是被毒的 (10s):

```
<-Youdie
<-Carry
<-Set 1 0    client保证不能选择带走已死的人（包括上次Status中死的以及Die中报告的）
->vote @ （留空不带人）
<-Set 0 0
如果带人走了
    <- Hunter（群发）
```

如果你是警长，昨晚死了(10s):

```
<- Captaindie(群发)
<- Pass
<- Set 1 0
-> vote @    撕毁警徽则@后留空
<- Set 0 0
如果传递了警徽
    ->Captain 群发
```

如果是初夜死掉或者猎人带走的人(10s):

```
<- Tellsth
<- Set 0 1 Room
-> roomchat
<- Set 0 0
```

<- Status

如果第一日
对活着的玩家

```
<- Choosecap
依次竞选演讲（不竞选就发言说不竞选）（10s每人）：
    <- Set 0 1 Room
    -> roomchat
    <- Set 0 0
投票选警长阶段
<- Set 1 0 Room
-> vote @
<- Set 0 0 Room
<- Captain（群发）
```

对警长(如果有):

```
<- Order
<- Set 1 0 Room
-> vote @1（顺时针） 或者 vote @0（逆时针）
<- Set 0 0
```

按顺序依次(15s):

```
<- Set 0 1 Room  
-> roomchat  
<- Set 0 0
```

对警长（如果有）(20s):

```
<- Set 0 1 Room  
-> roomchat  
<- Set 0 0
```

<- Vote 群发

<- Set 1 0

-> vote @... (10s)

<- Set 0 0

如果平票:

```
<- Equal 群发  
<- Vote  
<- Set 1 0  
-> vote @... (10s)  
<- Set 0 0
```

<- Die

跟上面一样结算死亡

<- Status