**前言**

API的发送方式均为socket

文档将分为两部分，第一部分为从client端发送到server端的API，第二部分为从server端发送到client端的API，发送时均打包成JSON格式

现制定API基本格式如下：

client -> server API:

|  |  |  |
| --- | --- | --- |
| 名称 | 是否必须 | 说明 |
| type | 是 | API类型 |
| data | 是 | API数据 |

server -> client API：

|  |  |  |
| --- | --- | --- |
| 名称 | 是否必须 | 说明 |
| type | 是 | API类型 |
| data | 是 | API数据 |
| msg | 否 | API附带的消息（如提示或者警告信息） |

需注意的三个基本点：

1. 什么时候需要发送该API
2. 发送给哪些socket对象，即发送给哪些用户
3. 什么时候需要去准备接受该API（网页端更需注意这点）

目录

[1. 客户端发送的API 3](#_Toc468123126)

[1.1 用户登录（type: login） 3](#_Toc468123127)

[1.2 加入房间（type: join\_room） 3](#_Toc468123128)

[1.3 文字消息（type: text\_message） 3](#_Toc468123129)

[1.4 开始游戏（type: start\_game） 3](#_Toc468123130)

[1.5 选择目标（type: choose\_target） 4](#_Toc468123131)

[1.6 是否选择该功能（type: Is\_function\_chosen） 4](#_Toc468123132)

[2. 服务端发送的API 4](#_Toc468123133)

[2.1 用户登录成功（type: login\_succ） 4](#_Toc468123134)

[2.2 文字消息（type: text\_message） 4](#_Toc468123135)

[2.3 可以开始游戏（type: game\_aviliable） 5](#_Toc468123136)

[2.4 发送身份牌（type: send\_id\_card） 5](#_Toc468123137)

[2.5 天黑或天亮（type: get\_dark） 5](#_Toc468123138)

[2.6 公布阶段开始（type: announce\_period\_started） 5](#_Toc468123139)

[2.7 用户被选择（type: user\_is\_chosen） 6](#_Toc468123140)

[2.8 用户被锁定（type: user\_is\_locked） 6](#_Toc468123141)

[2.9 用户是否好人（type: Is\_user\_goodman） 6](#_Toc468123142)

[2.10 用户被杀死（type: user\_dead） 6](#_Toc468123143)

[2.11 用户可以发言（type: start\_talking） 6](#_Toc468123144)

[2.12 用户开始投票（type: start\_voting） 7](#_Toc468123145)

[2.13 用户被票出（type: user\_out） 7](#_Toc468123146)

[2.14 游戏结束（type: game\_over） 7](#_Toc468123147)

[3. API调用流程 7](#_Toc468123148)

# 客户端发送的API

#### 用户登录（type: login）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| name | 是 | string | 用户昵称 |

#### 加入房间（type: join\_room）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| number | 是 | int | 房间号，1000到9999之间的数 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

#### 文字消息（type: text\_message）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| message | 是 | string | 聊天文字消息 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

#### 开始游戏（type: start\_game）

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Game\_started | 是 | Bool | 开始游戏 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

房主选择开始游戏

#### 选择目标（type: choose\_target）

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Target\_id | 是 | Int | 被选择目标的id |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

当前选定的用户

#### 是否选择该功能（type: Is\_function\_chosen）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Is\_chosen | 是 | Bool | 是否选择该功能 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

# 服务端发送的API

#### 用户登录成功（type: login\_succ）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| name | 是 | string | 用户昵称 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |
| Users | 是 | List | 目前房间内所有的用户信息 |

用户登录后需返回改API

#### 文字消息（type: text\_message）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| message | 是 | string | 聊天文字消息 |
| id | 是 | int | 用户id，标记消息是谁发送的 |
| name | 是 | string | 消息发送人的昵称 |

服务端用来分发聊天信息给用户

#### 可以开始游戏（type: game\_aviliable）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Is\_game\_aviliable | 是 | Bool | 可以开始游戏 |

给房主发送可以开始游戏的标志

#### 发送身份牌（type: send\_id\_card）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Id\_card\_type | 是 | int | 身份牌类型 |

给房间内用户发送身份牌

约定：

|  |  |
| --- | --- |
| 角色id | 身份 |
| 1 | 平民 |
| 2 | 狼人 |
| 3 | 预言家 |
| 4 | 女巫 |
| 5 | 守卫 |

#### 天黑或天亮（type: get\_dark）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Is\_sky\_dark | 是 | Bool | True为天黑，False为天亮 |

房间内全部人发送天黑了或天亮了的标志

#### 公布阶段开始（type: announce\_period\_started）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| period | 是 | String | 现在是这个阶段 |
| time | 是 | Int | 该阶段选择操作的时间，单位：秒 |
| me | 否 | Bool | 在遗言和发言阶段，如果该用户是发言人，则这个属性是true |

房间内全部人发送现在到了某阶段的标志，不同身份相应选择不同的操作

|  |  |
| --- | --- |
| 阶段 | period |
| 守卫 | guard |
| 狼人 | werewolf |
| 女巫救人 | witch\_rescue |
| 女巫毒人 | witch\_poison |
| 预言家查人 | prophet |
| 遗言 | last\_words |
| 发言 | discuss |
| 投票 | vote |

#### 用户被选择（type: user\_is\_chosen）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| id | 是 | List | 用户id |

当前用户被选择

#### 用户被锁定（type: user\_is\_locked）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| id | 是 | Int | 用户id |

当前用户被决定

#### 用户是否好人（type: Is\_user\_goodman）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Is\_user\_a\_goodman | 是 | Bool | 这个用户是否为好人 |

返回给预言家用户是否是好人

#### 用户被杀死（type: user\_dead）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Id1 | 是 | int | 用户id(-1为没死) |
| Id2 | 是 | Int | 用户id(-1为没死) |

这一轮的死亡名单

#### 用户被票出（type: user\_out）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Id | 是 | int | 用户id |

这一轮的被票出的用户id

#### 游戏结束（type: game\_over）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| Is\_game\_over | 是 | Bool | 游戏是否结束 |
| Id | 是 | List | 胜利用户的id |

# API调用流程

|  |  |  |  |
| --- | --- | --- | --- |
| 调用阶段 | API名称 | 调用端 | 解释 |
| 天黑 | 天黑或天亮 | 服务端 |  |
| 守卫守人 | 公布阶段开始 | 服务端 | 客户端根据自己的身份和现阶段选择相应的行为 |
| 守卫守人 | 选择目标 | 客户端 | 守卫选择要守护的目标 |
| 狼人杀人 | 公布阶段开始 | 服务端 | 客户端根据自己的身份和现阶段选择相应的行为 |
| 狼人杀人 | 选择目标 | 客户端 | 身份牌为狼人的用户选择相杀的人 |
| 狼人杀人 | 用户被选择 | 服务端 | 这些用户被狼人选择，实时显示给所有狼人看 |
| 狼人杀人 | 用户被锁定 | 服务端 | 时间终止后按算法决定最后谁被杀死，显示给所有狼人 |
| 女巫救人 | 公布阶段开始 | 服务端 | 客户端根据自己的身份和现阶段选择相应的行为 |
| 女巫救人 | 用户被锁定 | 服务端 | 显示狼人杀了谁给女巫 |
| 女巫救人 | 是否选择该功能 | 客户端 | 女巫是否救人 |
| 女巫毒人 | 公布阶段开始 | 服务端 | 客户端根据自己的身份和现阶段选择相应的行为 |
| 女巫毒人 | 是否选择该功能 | 客户端 | 女巫是否毒人 |
| 女巫毒人（若选择毒人） | 选择目标 | 客户端 | 女巫选择毒人目标 |
| 预言家查人 | 公布阶段开始 | 服务端 | 客户端根据自己的身份和现阶段选择相应的行为 |
| 预言家查人 | 选择目标 | 客户端 | 预言家选择要查身份的人 |
| 预言家查人 | 用户是否好人 | 服务端 | 显示给预言家刚刚查的人是好人还是坏人 |
| 天亮 | 天黑或天亮 | 服务端 |  |
| 发布被杀的人名单 | 用户被杀死 | 服务端 | 显示所有被杀的人 |
| 留遗言 | 公布阶段开始 | 服务端 | 给所有用户广播留遗言阶段开始，可以发言的用户me这个字段是true，表示可以发言；其他人是false。如果没有人可以留遗言，则没有这个阶段。 |
| 留遗言 | 文字消息 | 客户端 | 死人留遗言 |
| 留遗言 | 文字消息 | 服务端 | 服务端转发文字消息给所有用户 |
| 讨论 | 公布阶段开始 | 服务端 | 同留遗言 |
| 讨论 | 文字消息 | 客户端 |  |
| 讨论 | 文字消息 | 服务端 | 服务端转发文字消息给所有用户 |
| 投票 | 公布阶段开始 | 服务端 | 宣布开始投票 |
| 投票 | 选择目标 | 客户端 | 选择要票死的人 |
| 投票 | 用户被选择 | 服务端 | 广播当前投票情况 |
| 投票 | 用户被票出 | 服务端 | 公布投票结果 |
| 游戏结束 | 游戏结束 | 服务端 | 如果只剩下好人或者坏人在场上，宣告游戏结束 |