Server API

**1. 获取public key**

* API： http://{HOST}/{VERSION}/api/auth/getSecretKey
* Method： GET
* Desc： 获取Server RSA Public\_Key(公钥)
* 请求参数：NULL
* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| secretKey | String | 服务端公钥 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

secretKey:"r\nMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDXfH0eMqhpOxaDNyWZ1yjjv15f\r\n7qm0hnllBDLwI7O1ANYSStEQaqC3JS3dbx76OI2rhukFu2+f38jY4uJWc1FA3PuB\r\n1Ko1rtDIpmz2e0Gb2Qlx/h4qUkLuNdZGn8oa+xAvP4r7LrLQXgBj6KOwRfSvRYfU\r\ny4o9otafl9Dg6WmwRwIDAQAB\r\n"

},

message: “Get secret key success”

}

Fail：

{

status: 400,

data: {},

message: “Get public key fail”

}

**2. 获取手机验证码**

* API： http://{HOST}/{VERSION}/api/auth/obtainSMSCode
* Method： POST
* Desc： 获取手机验证码
* 请求参数：e.g. { codeType: “register”, telephone: “13621004542”, clientPublicKey: “public key” }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| codeType | String | 请求类别（register/others） | Yes |
| telephone | String | 电话号码 | Yes |
| clientPublicKey | String | 客户端公钥 | Yes |
| data | Object | 参数内容 | Yes |

* **备注： 限流：60s / 条， 60min / 5条， 24h / 10条**
* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| skipVerify | Boolean | 是否跳过验证 | Yes |
| verifyCode | String | 验证码 | Yes |
| expiresAt | String | 过期时间 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

skipVerify: false, (value: true / false),

verifyCode: “12345”,

expiresAt: “xxxxxxxxxx”

},

message: “Send SMS verify code success, expires time is 15min”

}

Fail：

{

status: 400,

data: {},

message: “Get telephone validate code fail”

}

**~~3. 验证手机验证码~~**

* API： http://{HOST}/{VERSION}/api/auth/verifySMSCode
* Method： POST
* Desc：验证手机验证码
* 请求参数：e.g. { params: { params: { codeType: “login”, telephone: “13621004542”, ’code’: ‘12345’, clientPublicKey: “public key” } } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| codeType | String | 请求类别（login/others） | Yes |
| telephone | String | 电话号码 | Yes |
| code | String | 验证码 | Yes |
| clientPublicKey | String | 客户端公钥 | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |
| secretKey | String | AES秘钥 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

user: {

"\_id": "59e07ba748104a2d60a6c0bd",

"telephone": “13631210000”,

"nickname": "Jamestest3",

"status": true,

"role": 0,

"signature": "This guy is lazy. He doesn't fill in anything...",

"male": false

},

secretKey: "DE\_MA\_XI\_YA!!!" ,

"token": "7y6GxZZ7Kj34b33ZABcyG4AHmMmJ4WJJ",

},

message: “Telephone verify success”

}

Fail：

{

status: 400,

data: {},

message: “Get telephone validate code fail”

}

**4. 用户注册(Register)**

* API： http://{HOST}/{VERSION}/api/auth/signup
* Method： POST
* Desc： 用户注册
* 请求参数：e.g. { params: { params: { user: { userData }, verifyCode:”12345”, clientPublicKey: “public key” } } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| telephone/email | String/String | 电话号码/邮件 | Yes |
| password | String | 密码 | Yes |
| nickname | String | 用户昵称 | Yes |
| deviceID | String | 设备ID | YES |
| avatar | String | 头像 | No |
| countryCode | String | 国际码 | No |
| male | Boolean | 是否女生 | No |
| signature | String | 签名 | No |
| verifyCode | String | 验证码 | YES |
| clientPublicKey | String | 客户端公钥 | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |
| secretKey | String | AES秘钥 | Yes |
| token | String | 用户Token | Yes |

* 返回示例

Success：

{

status: 200,

data: {

user: {

"\_id": "59e07ba748104a2d60a6c0bd",

"telephone": “13631210000”,

"nickname": "Jamestest3",

"avatar": "http://baidu.com",

"status": true,

"role": 0,

"signature": "This guy is lazy. He doesn't fill in anything...",

"male": false

},

secretKey: "DE\_MA\_XI\_YA!!!" ,

token: "UUID"

},

message: “User regist success”

}

Fail：

{

status: 400,

data: {},

message: “Sorry, Please fill in telephone”

}

**5. 用户登录(Login)**

* API： http://{HOST}/{VERSION}/api/auth/signin
* Method： POST
* Desc： 用户登录
* 请求参数：e.g. { params: { params: { user: { userData }, clientPublicKey: “public key” } } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| telephone | String | 用户名 | Yes |
| password | String | 密码 | Yes |
| deviceID | String | 设备ID | Yes |
| clientPublicKey | String | 客户端公钥 | Yes |
| verifyCode | String | 验证码 | No |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |
| verifyTelephone | Boolean | 异常登录？ | Yes |
| secretKey | String | AES秘钥 | Yes |
| token | String | 用户Token | Yes |

* 返回示例

Success：

{

status: 200,

data: {

user: {

"\_id": "59e07ba748104a2d60a6c0bd",

"telephone": "13631210000",

"nickname": "Jamestest3",

"avatar": "http://baidu.com",

"status": true,

"role": 0,

"signature": "This guy is lazy. He doesn't fill in anything...",

"male": false

},

verifyTelephone: false, (value: true / false)

secretKey: "DE\_MA\_XI\_YA!!!" ,

token: "UUID"

},

message: “Login success, welcome”

}

Fail：

{

status: 400,

data: {},

message: "Sorry, Your password is invalid"

}

**6. 获取联系人/个人信息资料**

* API： http://{HOST}/{VERSION}/api/user/getUserProfile
* Method： POST
* Desc： 获取个人、联系人资料信息
* 请求参数：e.g. { params: { params: { targetUserID: “c2547c7c8sa4d8af45cx4zv56adf” } , token: “” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| token | String | 当前用户token | Yes |
| targetUserID | String | 要查询的userID | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

userProfile: {

"userID": "59fbf016f1bbf808bac67d5c",

"telephone": "13631270436",

"nickname": "james01",

"avatar": "http://baidu.com",

"signature": "",

"male": false,

"countryCode": ""

}

},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “Server unknow error, get user profile fail”

}

**7. 请求添加联系人**

* API： http://{HOST}/{VERSION}/api/user/requestContact
* Method： POST
* Desc： 请求添加联系人
* 请求参数：{ params: { params: { contactID: “”}, token: “” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| contactID | String | 联系人userID | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This user is already your contact”

}

**8. 同意添加请求**

* API： http://{HOST}/{VERSION}/api/user/acceptContact
* Method： POST
* Desc： 添加联系人
* 请求参数：{ params: {params: { contactID: “”, remarkName: “”}, token: “” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| contactID | String | 联系人userID | Yes |
| remarkName | String | 备注 | No |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This user is already your friend”

}

**9. 拒绝添加请求**

* API： http://{HOST}/{VERSION}/api/user/rejectContact
* Method： POST
* Desc： 拒绝添加请求
* 请求参数：{ params: {params: { contactID: “” }, token: “” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| contactID | String | 联系人userID | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This user is already your contact”

}

**10. 删除联系人**

* API： http://{HOST}/{VERSION}/api/user/deleteContact
* Method： POST
* Desc： 删除用户联系人
* 请求参数：{ params: {params: { contactID: “” }, token: “” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| contactID | String | 联系人userID | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This user is not your contact”

}

**11. 获取当前用户联系人列表信息**

* API： http://{HOST}/{VERSION}/api/user/getUserContacts
* Method： POST
* Desc： 获取用户联系人列表
* 请求参数：e.g. { “params”: { “params”: {} , “token”: “current user token” } }
* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

contacts:[{

"userID": "59fbf016f1bbf808bac67d5c",

"nickname": "james01",

"avatar": "http://baidu.com",

"signature": "",

"male": false,

},

{

"userID ": "59fbf016f1bbf808bac67d5c",

"nickname": "james02",

"avatar": "http://baidu.com",

"signature": "",

"male": false,

}]

},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “Server unknow error, get user contacts list fail”

}

**~~12. 获取当前用户黑名单列表~~**

* API： http://{HOST}/{VERSION}/api/user/getBlackList
* Method： POST
* Desc： 获取用户黑名单列表
* 请求参数：null
* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* data参数解析

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| user | Object | 用户信息 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

userList:[{

"userID": "59fbf016f1bbf808bac67d5c",

"nickname": "james01",

"avatar": "http://baidu.com",

"status": true,

"signature": "",

"male": false,

},

{

"userID": "59fbf016f1bbf808bac67d5c",

"nickname": "james02",

"avatar": "http://baidu.com",

"status": true,

"signature": "",

"male": false,

}]

},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “Server unknow error, get black list fail”

}

**13. 更新联系人备注**

* API： http://{HOST}/{VERSION}/api/user/updateRemarkName
* Method： POST
* Desc：更新联系人备注名
* 请求参数：e.g. { “params”: { “params”: { “contactID”: “”, remarkName: “” } , “token”: “user token” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| contactID | String | 联系人userID | Yes |
| remarkName | String | 备注 | Yes |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This user is not your contact”

}

**14. 搜索用户**

* API： http://{HOST}/{VERSION}/api/user/searchUser
* Method： POST
* Desc：更新联系人备注名
* 请求参数：e.g. { params: { params: { queryCondition: “james”, pageSize: 0 } , token: “user token” } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| queryCondition | String | 条件(telephone || nickName) | Yes |
| pageSize | Int | 页码( default: 0 ) | No |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

userList:[{

"userID": "59fbf016f1bbf808bac67d5c",

"nickname": "james01",

"avatar": "http://baidu.com",

"status": true,

"signature": "",

"male": false,

},

{

"userID": "59fbf016f1bbf808bac67d5f",

"nickname": "james02",

"avatar": "http://baidu.com",

"status": true,

"signature": "",

"male": false,

}]

},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “Server unknow error, search user fail”

}

**15. Token 校验**

* API： http://{HOST}/{VERSION}/api/auth/tokenVerify
* Method： POST
* Desc：校验token
* 请求参数：e.g. { “params”: { “token”: “current user token”, “params”: {} } }

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| Token | String | 当前用户token | YES |

* 返回参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 说明 | 必含 |
| status | int | 状态 | Yes |
| data | Object | 数据 | Yes |
| message | String | 描述 | Yes |

* 返回示例

Success：

{

status: 200,

data: {

"userID": "59fbf016f1bbf808bac67d5c"

},

message: “”

}

Fail：

{

status: 400,

data: {},

message: “This token is invalid, please login again”

}

Constants

1. HOST：”127.0.0.1”
2. VERSION：”v1”
3. Aes secret: 'DE\_MA\_XI\_YA!!!'
4. RongCloud - appKey: "82hegw5u8dlsx"