# PlatON CLI

需求文档

# 文档介绍

## 1.1目的

本文档作为与社区开发人员相互沟通文档，定义了PlatON CLI业务需求和产品设计，供社区开发人员参考使用。

## 1.2范围

文档包括了PlatON CLI的需求概述，模块功能说明（业务流程，产品操作流程说明等）

# 需求概述

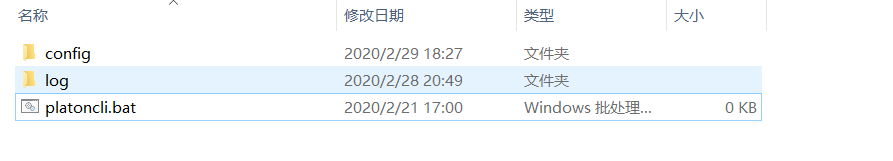
## 2.1软件结构

### 2.1.1 整体结构概览

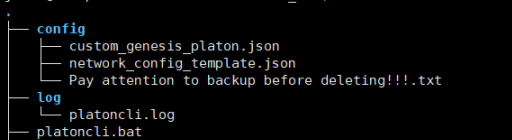
PlatON CLI为安装后为全局的命令使用，但是有其自己的文件结构，可再任意路径下使用

PlatON CLI工具总体结构如下：

Windows：



Linux：



### 2.1.2具体文件夹描述

#### 2.1.2.1 config文件夹

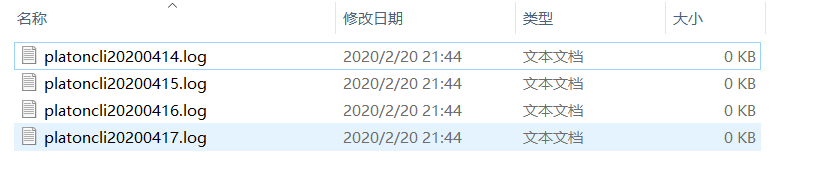
**genesis.json:** 用户自定义创世区块文件（支持用户可以创建私有节点，自定义创世区块文件）

**node\_config.json** 用户的交易发送目标节点ip和rpcport

**具体network\_config.json的内容如下**

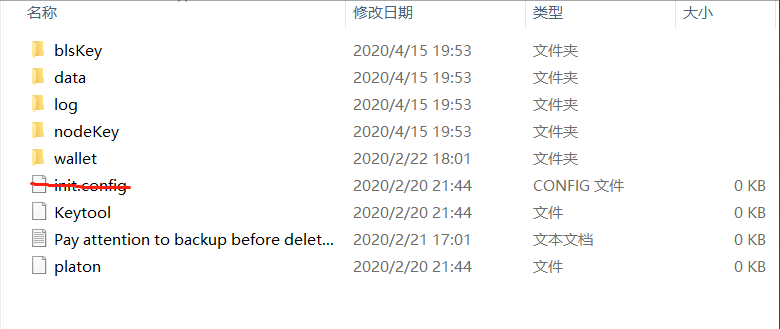
**需要手工配置，若完成配置，命令中通过name读取对应的ip和rpc端口发送指定节点**

#### 2.1.2.2 log文件夹



**Platoncli.log:**  CLI日志文件(按日期存储对应日志文件)

#### 2.1.2.3 节点初始化后拉取对应OS的**platon**文件夹



**blskey:**  blsPrivKey 和blsPubKey

**nodekey:** nodePrivKey 和nodePubKey

**Log**: 底层日志文件

**data:** 底层同步数据

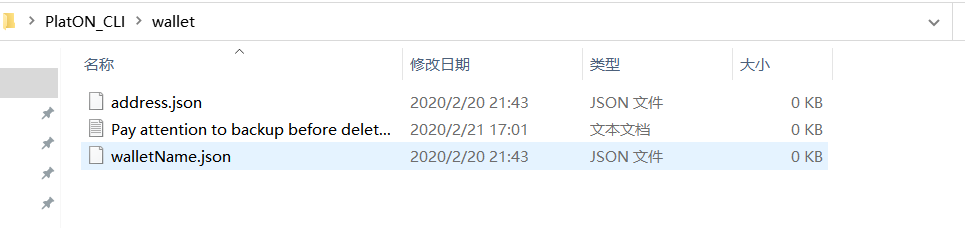
**keytool:** 生成nodekey、blskey、钱包文件工具

**platon:** 底层客户端

**Pay attention to backup before deleting!!!.txt:** 为防止用户误删重要文件，添加重要提示

**Wallet**：创建的钱包文件

#### 2.1.2.4 **wallet**文件夹



**address.json:** 以地址命名的钱包文件

**walletName.json:** 以用户自定义名称命名的钱包文件

**Pay attention to backup before deleting!!!.txt:** 为防止用户误删重要文件，添加重要提示

## 2.2产品形态

基于命令行的节点管理工具

## 2.3产品名称

platoncli —— <暂定>

## 2.4产品目标

·满足全节点从部署到质押的全流程

·高效的管理全节点

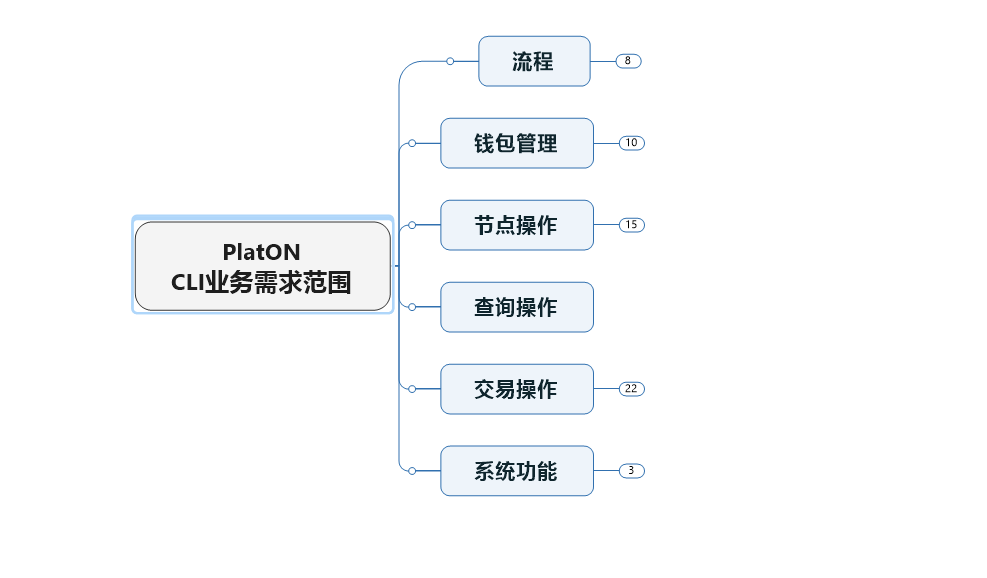
·简单便捷的参与PlatON和Alaya的治理

·安全的钱包管理模块

·基于良好的封装命令简化命令行使用难度

·可读性好的操作手册

## 2.5需求范围



## 2.6功能清单

|  |  |  |  |
| --- | --- | --- | --- |
| **需求** | **功能** | | **描述** |
| 钱包管理模块 | 创建钱包 | 常规钱包、  离线钱包 | 生成单个钱包文件  生成批量钱包文件 |
| ~~多重签名钱包（后续版本）~~ |
| 删除钱包 | | 删除指定地址或者名称的钱包文件 |
| 查看本地钱包 | | 通过名称或者地址查询单个钱包文件；查看默认路径下全部钱包文件 |
| 修改钱包密码 | | 指定地址或者名称修改钱包文件密码 |
| 备份和删除 | 助记词备份 | 通过助记词备份钱包文件 |
| 私钥备份 | 通过私钥备份钱包文件 |
| 私钥恢复 | 通过助记词或者私钥恢复钱包文件 |
| 离线签名 | | 对未签名交易进行离线签名 |
| 查询钱包余额 | | 查询钱包余额 |
| 节点操作模块 | 节点初始化 | | 节点的初始化下载platon相应的文件，建立工作目录 |
| 节点启动/重启 | | 启动节点；重启已经运行中的节点 |
| 节点停止 | | 停止节点进程 |
| 节点重置 | | 重置节点数据，清空节点（清除数据以及清除节点私钥） |
| 节点升级 | | 节点版本升级 |
| 节点回滚 | | 节点数据回滚重新同步 |
| 生成nodekey | | 生成节点的公私钥文件 |
| 生成blskey | | 生成节点的blskey公私钥对文件 |
| 节点的升级检测 | | 检测当前工作目录下节点的版本以及链上的最新版本 |
| 区块导出 | | 存量数据的区块导出功能 |
| 区块导入 | | 区块数据导入功能 |
| 私钥迁移 | | Blskey和nodekey迁移 |
| 交易模块 | 常规交易转账 | | 普通交易转账，区分在线交易或者生成离线待签名交易【后续考虑缓存私钥，提高体验；用户自定义选择】 |
| 发送已签名离线交易 | | 已签名交易数据发送 |
| 根据hash查询交易信息 | | 根据hash查询交易信息 |
| 根据hash查询交易回执信息 | | 根据hash查询交易回执信息 |
| 质押模块 | 质押交易 | | 创建验证人 |
| 修改质押交易 | | 修改质押信息 |
| 增持质押交易 | | 增持质押 |
| 退出质押交易 | | 退出验证人 |
| 查询当前共识周期验证人列表 | | 查询当前周期共识验证人列表 |
| 查询当前结算周期验证人列表 | | 查询当前结算周期验证人列表 |
| 查询所有实时候选人列表 | | 查询所有实时候选人列表 |
| 根据nodeid查询节点质押信息 | | 根据nodeid查询节点质押信息 |
| 查询当前的结算周期的质押奖励 | | 查询当前的结算周期的质押奖励 |
| 委托模块 | 创建委托交易 | | 委托 |
| 减持/撤销委托交易 | | 减持/撤销委托 |
| 提取委托奖励交易 | | 提取委托奖励 |
| 查询当前账户地址所有委托的节点的NodeId和质押id | | 查询委托相关质押节点信息 |
| 查询账户在各个节点未领取的委托奖励 | | 查询账户在各个节点未领取的委托奖励 |
| 治理模块 | 提交提案交易 | | 提交文本/参数/升级/取消提案 |
| 提案投票交易 | | 提案投票 |
| 版本声明交易 | | 版本声明 |
| 举报双签交易 | | 举报双签 |
| 查询提案列表交易 | | 交易 |
| 根据提案id查询提案信息 | | 根据提案id查询提案信息 |
| 查询提案结果 | | 查询提案结果 |
| 查询节点的链生效版本 | | 查询节点的链生效版本 |
| 查询最新的治理参数值 | | 查询最新的治理参数值 |
| 查询治理参数列表 | | 查询治理参数列表 |
| 查询提案的累计可投票人数 | | 查询提案的累计可投票人数 |
| 查询节点是否被多签举报 | | 查询节点是否被多签举报 |
| 链基本信息模块 | 查询当前最高块 | | 查询当前最高块 |
| 根据块高查询区块信息 | | 根据块高查询区块信息 |
| 根据区块hash查询区块信息 | | 根据区块hash查询区块信息 |
| 查询当前的结算周期的区块奖励 | | 查询当前的结算周期的区块奖励 |
| 查询区块打包的平均时间 | | 查询区块打包的平均时间 |
| 锁仓相关 | 创建锁仓计划交易 | | 创建锁仓计划 |
| 获取锁仓计划 | | 获取锁仓计划 |
| 帮助 | 查询命令 | | 获取操作命令列表 |

# 模块功能说明

## 初次启动初始化Client

前置条件：

1. 正常使用的PlatON CLI工具
2. 成功安装PlatON CLI工具

创建网络步骤：

1. 任目录下
2. 展示目前PlatON CLI工具的版本号
   1. platoncli init --withnode --type m ，用户初始化工作目录,同时下载platon相应的文件，并默认生成nodekey和blsKey文件，本质上是[节点初始化流程](#_3.1.1节点初始化)；参数--type 为连接的网络类型，不同的网络对应的创世区块文件不同；
   2. platoncli init ，当前目录下生成wallet夹
   3. 是否加入网络Y/N，是则输入IP和prot，否则成为离线钱包
   4. 配置网络
   5. 交互式输入：
      1. Please enter the node IP: XXXXXX交互式输入，回车后进入输入prot
      2. Please enter the node prot: XXXXXX，输入后默认此ip:prot默认为交易发送

加入网络步骤:

1. 输入后校验节点是否连接正常
   1. 使用telnet判断，正常则显示成功
   2. 失败则提示用户是否切换节点，
      1. Y，则返回交互式重新输入ip和prot
      2. N则提示连接节点失败

|  |  |  |  |
| --- | --- | --- | --- |
| **操作** | **必填性** | **命令参数** | **参数说明** |
| init | optional | --withnode,-w | 初始化cli工具，并初始化节点 |
| 若带--withnode则为must，否则为optional | --type,-t | 参数内容为链接网络类型 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **必填性** | **参数详解** | | |
| --type，-t | must | 连接主网 | 连接测试网 | 连接私有网络 |
| --type m | --type t | --type p |
| -t m | -t t | -t p |

Ps: 若初始化节点，则根据-t 参数中连接网络类型确认chainId，意味着打包CLI工具前就明确各种不同网络的chainId并写入**node\_config.json 中chainId字段**，若是为私有网络则，根据提供自定义创世区块文件**genesis.json**中chainId，写入文件配置文件中chainId字段

#### 提示语

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| System build -- path, user initializes new working directory | 用户初始化新的工作目录 |
| System used -- path, the user uses the existing working directory | 用户使用已存在的工作目录 |
| Join the network, transaction sending requires a clear node IP and port. Follow the steps, or fill in the configuration file in XXXX file under xxxxx directory | 加入网络，交易发送需要明确的节点ip和port，请按步骤执行，或者请用户在XXXXX目录下，xxxx文件手动填写配置文件 |
| If the profile is not configured, the transaction module cannot be used normally | 若不配置文件，交易模块无法正常使用 |
| If the profile is not configured, Normal use of wallet template | 若不配置文件，钱包模板使用正常 |
| Node module cannot be used in this mode | 此模式下无法使用node模块 |
| Connection exception，Whether to switch node Y/N | 无法加入到网络，是否切换节点 |
| Success | 连接成功 |
| Execute the following command to initialize the node:  platoncli node init --withnode | 请执行以下命令初始化节点：  platoncli node init --withnode |

## 下载说明

两种下载方式

1. 提供官方网站下载ZIP包
2. 提供cmd命令行直接下载

## 3.1**节点操作**

### 3.1.1节点初始化

#### 3.1.1.1描述

前置条件：

1. 正确下载platon cli工具
2. 成功安装并正确使用
3. 网络通畅

步骤：

1. 用户指定文件夹下使用platoncli工具
2. 输入节点初始化命令
3. 判断参数提供路径下是否已经存在工作目录
   1. 是，则提示用户，是否覆盖Y/N

i 是，则覆盖文件，并提示风险

ii 否，则初始化工作目录失败，建议切换目录

* 1. 否，则进行下一步

1. 根据PlatON CLI工具所在os系统拉取对应的platon节点运行所须的文件
2. 若用户需要修改启动端口优先修改node\_config.json文件
3. 节点生成keytool生成nodekey和blskey

i 生成后的nodekey和blskey展示在命令行界面同时展示生成的地址

ii 默认p2p端口：16789

iii 默认rpc端口：6789

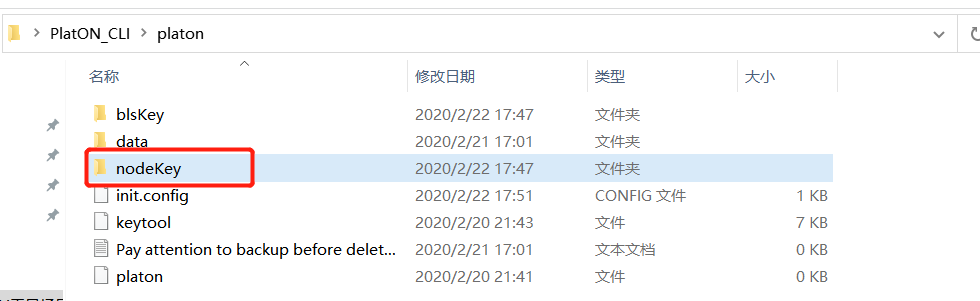
1. 初始化成功

判断安装目录是否存在PlatON工作目录，判断依据

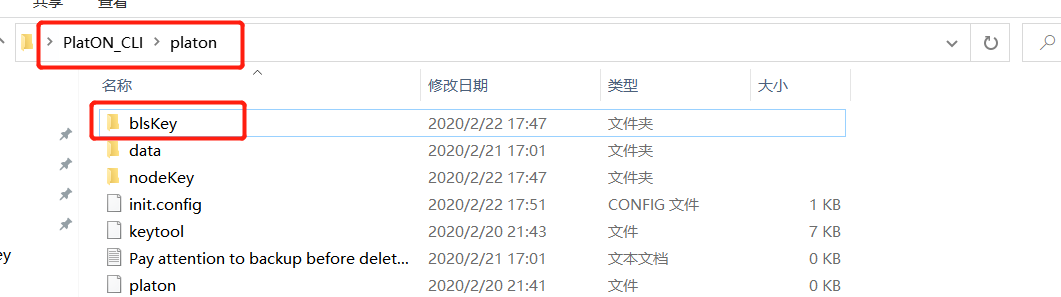
1. 指定目录下是否有platon客户端

说明：

1. nodekey生成的目录为工具目录下platon/nodeKey，nodePubKey和nodePirvKey分别存储，nodePubKey就是节点的nodeId



1. blskey生成的目录为工具目录下platon/blsKey，nodePubKey和nodePirvKey分别存储



1. 提供用户修改节点端口号和底层日志级别，若需要自定义则需要优先修改此文件
2. Nodekey和blskey展示的格式如下

nodePubKey: xxxxxxxxxxxxxxxxxxxxxx

nodePrivKey: xxxxxxxxxxxxxxxxxxxxxx

path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

blsPubKey: xxxxxxxxxxxxxxxxxxxxxx

blsPrivKey: xxxxxxxxxxxxxxxxxxxxxx

path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

1. 后续启动命令中所需的nodeid——nodePubKey，节点的端口号从node\_config.json中获取
2. 启动的命令rpcaddress 默认localhost，安全保护rpc端口只能本机使用
3. 启动命令 rpcapi支持全部api接口

#### 3.1.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node init --t

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **参数说明** |
| node | init | --config,-c | 参数内容内配置文件，默认为：config/node\_config.json |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **必填性** | **参数详解** | | |
| --type，-t | must | 连接主网 | 连接测试网 | 连接私有网络 |
| --type m | --type t | --type p |
| -t m | -t t | -t p |

Ps: 若初始化节点，则根据-t 参数中连接网络类型确认chainId，意味着打包CLI工具前就明确各种不同网络的chainId并写入**node\_config.json 中chainId字段**，若是为私有网络则，根据提供自定义创世区块文件**genesis.json**中chainId，写入文件配置文件中chainId字段

#### 3.1.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Initialization failed | 节点初始化失败 |
| Initialization successful | 节点初始化成功 |
| XX: command not found | 命令不存在 |
| Under this directory, the node has been initialized | 该目录下，节点已经初始化 |
| Overwrite this working directory,Y/N? | 是否覆盖此工作目录 |
| Cover risks, please operate carefully | 覆盖有风险，请谨慎操作 |
| PlatON related file acquisition failed | platon相关文件获取失败 |
| Bad parameters,Command cannot be resolved | 输入错误参数/--type未填写，或则填写错误的参数值 |
| NodeKey Create failure | Nodekey生成失败 |
| BlskKey Create failure | Blskey生成失败 |
| Keytool exception | 调用keytool异常/失败 |

### 3.1.2节点停止

#### 3.1.2.1描述

前置条件：

1. 节点已初始化且进程正常运行中
2. 电脑网络通畅

步骤：

1. 用户指定文件夹下使用platoncli工具
2. 用户输入节点停止命令
   1. 显示当前工作目录并提示用户Y/N继续执行，
   2. 是则执行下一步
   3. 否则提示用户命令自定义节点目录
3. 检测是否有节点进程运行中
   1. 若有节点运行中，则停止节点
   2. 若没有节点运行中，提示用户无节点进程

D.节点停止成功

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node stop

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **参数说明** |
| node | stop | nil | 停止节点 |

#### 3.1.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Stop success | 节点停止成功 |
| Node process not detected | 未检测到节点进程 |
| Stop failure | 节点停止失败 |
| XX: command not found | 命令不存在 |
| Current workspace: XXXXXXXX, continue Y/N? | 当前工作空间：xxxxxxxxx，是否继续执行 |
| Please switch the target working directory and use the command ： system used -- path | 请切换目标工作目录，使用命令 system used --path |

### 3.1.3节点重置

#### 3.1.3.1描述

前置条件：

1. 已有正确初始化的节点

步骤：

1. 用户指定文件夹下使用platoncli工具
2. 用户输入命令
   1. 显示当前工作目录并提示用户Y/N继续执行
   2. 是则执行下一步
   3. 否则提示用户命令切换自定义节点目录
3. 检测节点是否初始化过
   1. 若已经初始化，则进行下一步
   2. 若无初始化提示，提示用户节点未初始化，无需重置
4. 检测是否有节点进程在运行中
5. 若在运行中，则提示用户Y/N是否停止节点

i 用户键入Y，则停止节点，进行下一步

ii 用户键入N，提示已有节点进程运行中

1. 若无运行中进入下一步
2. 删除前提示用户，根据实际情况备份所需要的文件
3. 完成节点重置

PS：是否初始化依据

1. 指定目录下是否有platon客户端

PS：是否已部署依据

1. 是否有nodekey和blskey文件
2. 是否有有data数据
3. 满足其中之一就说明部署过

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node reset 只清除data文件和日志

命令：platoncli node reset -c 清除全部文件包括key

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | reset | --clear,-c | optional | 填入此参数，表示清除节点全部数据，包括data、nodekey、blskey、log日志，不写入此参数只清除data文件 |

#### 3.1.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Node not initialized, no reset required | 节点未初始化无需重置 |
| During the node process running，Reset failure | 节点进程运行中，用户放弃停止节点 |
| according to the need back up files | 删除前提示用户，根据需要备份文件 |
| Reset success | 重置成功 |
| Reset failure | 重置失败 |
| XX: command not found | 命令不存在 |
| Current workspace: XXXXXXXX, continue Y/N? | 当前工作空间：xxxxxxxxx，是否继续执行 |
| Please switch the target working directory and use the command ： system used -- path | 请切换目标工作目录，使用命令 system used --path |

### 3.1.4节点启动/重启

#### 3.1.4.1描述

前置条件：

1. 节点正确完成初始化
2. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户执行启动/重启
   1. 显示当前工作目录并提示用户Y/N继续执行
   2. 是则执行下一步
   3. 否则提示用户命令切换工作空间
3. 读取node\_config.json获取到端口，日志级别等
4. 读取节点的blskey和nodekey
5. 检测节点是否部署
   1. 若已经部署，则进行下一步
   2. 若无部署提示，未部署节点启动失败
6. 检测是否有节点进程在运行中
   1. 若在运行中，则提示用户Y/N是否重启节点

i 用户键入Y，则先停止节点，启动节点

ii 用户键入N，提示已有节点进程运行中，用户放弃重启

1. 启动/重启节点
2. 启动成功
3. 启动成功后在命令行界面打印底层日志5s

PS：是否初始化依据

1.指定目录下是否有platon客户端

PS：是否已部署依据

1. 是否有nodekey和blskey文件
2. 是否有有data数据
3. 满足其中之一就说明部署过

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node start

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **参数说明** |
| node | start | nil | 启动/重启节点 |

#### 3.1.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Startup failure | 节点启动失败 |
| Startup success | 节点启动成功 |
| Node not initialized | 节点未初始化 |
| During the node process running，Startup failure | 节点进程运行中，重启失败 |
| Failed to read configuration file | 读取node\_config.json配置失败 |
| Nodekey file not found | Nodekey未找到 |
| Failed to read Nodekey file | 读取Nodekey文件失败 |
| Blskey file not found | Blskey 未找到 |
| Failed to read Blskey file | 读取Blskey 文件失败 |
| There is a risk in restarting to ensure that the node is not the current Validator | 有检测到进程运行中，用户选择重启节点，给与提示 |
| XX: command not found | 命令不存在 |
| Current workspace: XXXXXXXX, continue Y/N? | 当前工作空间：xxxxxxxxx，是否继续执行 |
| Please switch the target working directory and use the command ： system used -- path | 请切换目标工作目录，使用命令 system used --path |

### 3.1.5节点升级

#### 3.1.5.1描述

前置条件：

1. 正确的渠道得知github升级提案
2. 支持节点升级提案并投票
3. 电脑网络通畅
4. 正规渠道下载升级对应的platon
5. 已初始化正常运行的节点

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入命令，并提供下载完成后的PlatON客户端路径
   1. 显示当前工作目录并提示用户Y/N继续执行
   2. 是则执行下一步
   3. 否则提示用户命令切换工作空间
3. 检测节点是否初始化
4. 若已经初始化，则进行下一步
5. 若无初始化提示，提示用户节点未初始化，提示用户下载最新的PlatON CLI工具
6. 检测是否有节点进程运行中
7. 若在运行中，则提示用户Y/N是否停止节点

i 用户键入Y，则停止节点，进行下一步

ii 用户键入N，提示已有节点进程运行中

1. 若无运行中进入下一步
2. 根据用户提供的PlatON客户端路径替换工具中的PlatON文件
3. 替换后启动节点
4. 节点升级成功启动节点，并在命令行界面打印底层日志5s

PS：是否初始化依据

1. platon/data 目录下是否有区块数据
2. platon/nodekey 目录下是否存在文件
3. platon/blskey 目录下是否存在文件

PS：是否有节点进程依据

1.是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.5.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node upgrade --path C:\Users\Administrator\Desktop\platon

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | upgrade | --path,-p | must | 下载对应的PlatON客户端文件路径 |

#### 3.1.5.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Node not initialized，Download the latest PlatON CLI ，link http://xxx.xxx.xxx | 节点未初始化，请下载最新platon cli客户端,并提供下载地址 |
| The process is running. Please stop the process and upgrade，Upgrade failed | 节点进程运行中，流程中用户放弃停止节点 |
| invalid path，Upgrade failed | --path路径无法解析 |
| Not found platon，Upgrade failed | 没找到platon文件 |
| Upgrade failed | 升级失败 |
| Upgrade success | 升级成功 |
| Startup failure | 节点启动失败 |
| XX: command not found | 命令不存在 |
| Current workspace: XXXXXXXX, continue Y/N? | 当前工作空间：xxxxxxxxx，是否继续执行 |
| Please switch the target working directory and use the command ： system used -- path | 请切换目标工作目录，使用命令 system used --path |

### 3.1.6节点回滚

#### 3.1.6.1描述

前置条件：

1. 正确的渠道得知数据回滚
2. 正规渠道获取回滚data数据
3. 下载data数据
4. 已初始化正常运行的节点

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入命令，并提供下载完成后的data路径
   1. 显示当前工作目录并提示用户Y/N继续执行
   2. 是则执行下一步
   3. 否则提示用户命令切换工作空间
3. 检测节点是否初始化
4. 若已经初始化，则进行下一步
5. 若无初始化提示，提示须初始化节点
6. 检测是否有节点进程运行中
7. 若在运行中，则提示用户Y/N是否停止节点

i 用户键入Y，则停止节点，进行下一步

ii 用户键入N，提示已有节点进程运行中

1. 根据用户提供下载后的data数据路径，替换data文件夹
2. 操作成功后重启节点
3. 启动节点在命令行界面打印5s底层日志

PS：是否初始化依据

1. platon/data 目录下是否有区块数据
2. platon/nodekey 目录下是否存在文件
3. platon/blskey 目录下是否存在文件

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.6.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli node rollback --path C:\Users\Administrator\Desktop\data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | rollback | --path,-p | must | 下载回滚后的区块数据所在路径 |

#### 3.1.6.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Node not initialized, node must be initialized, rollback failed | 节点未初始化，请初始化节点，回滚失败 |
| Rollback failure | 回滚失败 |
| Rollback success | 回滚成功 |
| invalid path | 无效的路径 |
| Data file not found | Data文件未找到 |
| Node process is running, please stop after operation，rollback failed | 节点进程运行中，请停止后操作，回滚失败 |
| Startup failure | 启动失败 |
| XX: command not found | 命令不存在 |
| Current workspace: XXXXXXXX, continue Y/N? | 当前工作空间：xxxxxxxxx，是否继续执行 |
| Please switch the target working directory and use the command ： system used -- path | 请切换目标工作目录，使用命令 system used --path |

### 3.1.7生成nodekey

#### 3.1.7.1描述

前置条件：

1. 正常使用的PlatON CLI工具

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入参数生成nodekey
3. 根据参数提供路径，判断目录下是否有存在nodepubkey和nodeprivkey
   1. 是，提示用户存在，是否覆盖生成
   2. 否，进行下一步
4. 生成完成

#### 3.1.7.1命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node generateNodeKey --path C:\Users\Administrator\Desktop\data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | generateNodeKey | --path,-p | must | 生成nodeKey的到指定目录 |

#### 3.1.7.1 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| invalid path | 无效的路径 |
| XX: command not found | 命令不存在 |
| Generate success | 生成成功 |
| Generate fail | 生成失败 |
| The key file already exists in the directory. Do you want to overwrite it Y/N？ | 目录下已存在key文件，是否覆盖生成 |

### 3.1.8生成blskey

#### 3.1.8.1描述

前置条件：

1. 正常使用的PlatON CLI工具

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入参数生成blskey
3. 根据参数提供路径，判断目录下是否有存在blspubkey和blsprivkey
   1. 是，提示用户存在，是否覆盖生成
   2. 否，进行下一步
4. 生成完成

#### 3.1.8.1命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node generateBlsKey --path C:\Users\Administrator\Desktop\data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | generateBlsKey | --path,-p | must | 生成blsKey的到指定目录 |

#### 3.1.8.1 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| invalid path | 无效的路径 |
| XX: command not found | 命令不存在 |
| Generate success | 生成成功 |
| Generate fail | 生成失败 |
| The key file already exists in the directory. Do you want to overwrite it Y/N？ | 目录下已存在key文件，是否覆盖生成 |

### 3.1.9节点升级检测

#### 3.1.9.1描述

前置条件：

1. 正常使用的PlatON CLI工具

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入检测命令
3. 展示出链上最新的版本号以及当前节点的版本号

PlatON Version： xxxx 链上最新版本

Current version： xxxx 当前版本

Version declared: xxxx 已声明的版本

* 1. 若链上最新版本高于当前版本，提示用户节点需要做升级操作
  2. 若版本一致，则无需提示

#### 3.1.9.1命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node check version

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | check |  |  |  |

#### 3.1.9.1 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| The current version is not the latest version. It is recommended that the user change the version of Plato. Please provide the latest download address according to the official and use the command:  platoncli node upgrade --path  Upgrading | 当前版本不是最新版本，建议用户更platon版本，请根据官方提供最新的platon下载地址，并使用命令：  platoncli node upgrade --path  进行升级 |

### 3.1.10查询当前工作目录下的节点信息

#### 3.1.10.1描述

前置条件：

1. 正常使用的PlatON CLI工具

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入检测命令
3. 展示出当前工作目录下的节点信息
   1. NodeId:xxxxxxxxxx
   2. NodeName: xxxxxxxxxx
   3. BenefitAddress:xxxxxxxxxx
4. 若当前工作目录下没有运行的节点则提示

当前目录下没有运行的节点

1. 若当前节点没有质押，则nodeName和benfitAddress为空

#### 3.1.10.1命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node info

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | info |  |  |  |

#### 3.1.10.1 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| There are no running nodes in the current directory | 当前目录下没有运行的节点 |

### 3.1.11区块导出

#### 3.1.11.1描述

前置条件：

1. 节点正常运行
2. PlatON CLI工具正常
3. 节点已初始化
4. 停止节点进程

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 检测节点是否部署
   1. 是，则进行下一步
   2. 没部署，提示用户无部署，无数据导出
3. 检测节点进程是否运行中
   1. 用户输入Y/N是否停止节点

i：Y，则停止节点，进入下一步

ii：N，提示用户，手动停止节点

1. 根据参数指定路径导出数据，数据打包成ZIP格式
2. 导出完成

PS：是否初始化依据

1. platon/data 目录下是否有区块数据
2. platon/nodekey 目录下是否存在文件
3. platon/blskey 目录下是否存在文件

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.11.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node blockexport --path C:\Users\Administrator\Desktop\data

导出数据格式要求ZIP包，命令执行完，将ZIP包放入指定路径中

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | blockexport | --path,-p | must | 指定数据导出路径 |

#### 3.1.11.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Node not initialized, node must be initialized, export failed | 节点未初始化，请初始化节点，导出失败 |
| export failure | 导出失败 |
| export success | 导出成功 |
| invalid path | 无效的路径 |
| Data file not found | Data文件未找到 |
| Node process is running, please stop after operation，export failed | 节点进程运行中，请停止后操作，导出失败 |
| XX: command not found | 命令不存在 |

### 3.1.12区块导入

#### 3.1.12.1描述

前置条件：

1. PlatON CLI工具正常
2. Data数据正确

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 检测节点是否部署
   1. 是，则进行下一步
   2. 没部署，提示用户无部署，无数据导出
3. 检测节点进程是否运行中
4. 用户输入Y/N是否停止节点

i：Y，则停止节点，进入下一步

ii：N，提示用户，手动停止节点

1. 根据参数指定路径导入数据
2. 导入完成

PS：是否初始化依据

1. platon/data 目录下是否有区块数据
2. platon/nodekey 目录下是否存在文件
3. platon/blskey 目录下是否存在文件

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.12.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node blockinput --path C:\Users\Administrator\Desktop\data

将导出的ZIP包的path作为参数，命令执行后直接将ZIP包解压到

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | blockinpute | --path,-p | must | 指定数据导入路径 |

#### 3.1.12.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Node not initialized, node must be initialized, impute failed | 节点未初始化，请初始化节点，导入失败 |
| impute failure | 导入失败 |
| impute success | 导入成功 |
| invalid path | 无效的路径 |
| Data file not found | Data文件未找到 |
| Node process is running, please stop after operation，impute failed | 节点进程运行中，请停止后操作，导入失败 |
| XX: command not found | 命令不存在 |

### 3.1.13数据迁移

#### 3.1.13.1描述

前置条件：

1. PlatON CLI工具正常
2. Data数据正确
3. 目标迁移目录以及完成初始化

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入迁移命令
3. 检测节点是否部署
4. 是，则进行下一步
5. 没部署，提示用户无部署，无数据导出
6. 检测节点进程是否运行中
7. 用户输入Y/N是否停止节点

i：Y，则停止节点，进入下一步

ii：N，提示用户，手动停止节点

1. 检测迁移目标路径下是否存在工作空间(初始化)
2. 根据参数解析，具体迁移的key文件还是data文件
3. 迁移数据
4. 完成

PS：是否初始化依据

1. 指定目录下是否有platon客户端

PS：是否已部署依据

1. 是否有nodekey和blskey文件
2. 是否有有data数据
3. 满足其中之一就说明部署过

PS：是否有节点进程依据

1. 是否有进程的占用端口号与platon/node\_config.json一致

依据有待补充

#### 3.1.13.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli node move -o $original\_path -t $tar\_path --movetype key/data/all

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| node | move | --original,-o | must | 原数据目录，只需要platon工作目录即可 |
| --target,t | must | 迁移目标数据目录，只需要platon工作目录即可 |
| --movetype,-m | must | 迁移类型  key —— 迁移nodekey和blskey  data —— 迁移区块数据  all -- 迁移key和data |

#### 3.1.13.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Key file not found | Key文件未找到 |
| Data file not found | Data文件未找到 |
| Node process is running, please stop after operation，impute failed | 节点进程运行中，请停止后操作，导入失败 |
| XX: command not found | 命令不存在 |
| Working directory not detected for target path | 目标路径未检测到工作目录 |
| No working directory detected under the original path | 原路径下未检测到工作目录 |
| Migration failure | 迁移失败 |
| Node not deployed, no files to migrate | 节点未部署，没有可迁移的文件 |

## 3.2 钱包管理

特别说明：密码长度限制8-12位，超过位数则无法输入

### 3.2.1 创建钱包

#### 3.2.1.1描述

前置条件：

1. PlatON CLI工具安装正常

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 输入创建钱包文件命令
3. 判断是否存在此命名的钱包文件
4. 交互式输入命令
5. 输入钱包文件密码两次
   1. 若两次不一致生成失败，提示用户两次输入密码不一致
   2. 若成功则进入下一步
   3. 密码在界面上不显示具体内容，只显示”\*”，根据实际用户输入多少位展示多少个”\*”
   4. 批量生成钱包文件密码密码一致
6. 生成后展示钱包文件生成所在路径
   1. 单个钱包文件生成展示

WalletName: xxxxxxxxx.json

PublicKey: xxxxxxxxxxxxxxxxxxxxxx

Addresss:xxxxxxxxxxxxxxxxxxxxx

Path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

* 1. 批量生成钱包文件展示

Path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

WalletName: 1.json

WalletName: 2.json

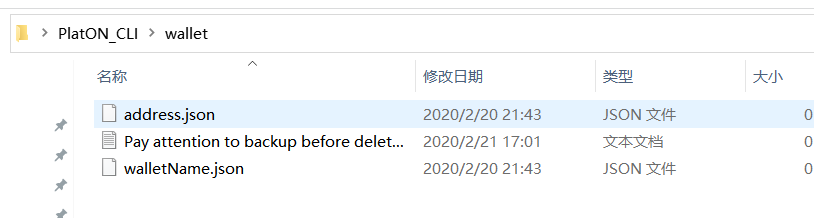
WalletName: 3.json

1. 提示用户是否进入备份流程
   1. 用户键入Y，进入备份流程

i 默认已助记词备份钱包文件，

* 1. 用户键入N，提示用户安全因素请备份钱包
  2. 批量生成不进入此环节

G.生成钱包完成



默认生成钱包的目录

#### 3.2.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account new --name <walletName>

命令：platoncli account new --batch <Number of wallet generated in batch>

命令：platoncli account multiple --name <walletName>

命令：platoncli account multiple --batch <Number of wallet generated in batch>

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **参数说明** |
| account | new | --name,-n | 钱包文件名称 |
| --batch，-b | 批量生成钱包个数 |
| ~~multiple~~ | ~~--name,-n~~ | ~~多重签名钱包文件名称~~ |
| ~~--batch，-b~~ | ~~批量生成多重签名钱包个数~~ |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **参数** | **必填性** | **参数详解** | **数据类型** | **解释** |
| --name，-n | 单个钱包生成must | 指定生成钱包的名称，不填写默认生成已钱包地址.json命名的钱包文件 | 不支持中文，最长42位  支持大小写字母，常用符号，数字的字符串 | 生成单个钱包文件--name必填；若批量生成钱包文件--name可输入，但是无效，批量生成默认钱包文件为钱包文件的address.json |
| 批量生成填写无效 |  |  |
| --batch，-b | 若须批量生成  must | 指定生成钱包个数 | 只支持数字2-50 |
| 若生成单个钱包无需填写 |  |  |

|  |  |  |
| --- | --- | --- |
| **参数** | **必填性** | **备注** |
| 密码 | must | 密码长度大于等6于位，  支持并区分大小写，数字，常用符号 |

**生成单个钱包文件**

platoncli account new/multiple --name testAccount1

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

WalletName: xxxxxxxxx.json

PublicKey: xxxxxxxxxxxxxxxxxxxxxx

Addresss:xxxxxxxxxxxxxxxxxxxxx

Path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Enter backup or not Y/N?

Y

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

Backup Wallet name：xxxxxxxxx.json

Backup Address:xxxxxxxxxxxxxxxxxxxxx

Mnemonic code：cloth tortoise elephant scrub shrug deal dizzy keen idle glide issue

write this mnemonic phrase in a safe place.

It is the important way to recover your account if you ever forget your password.

**批量钱包文件**

platoncli account new/multiple --batch 3

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

Path: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

WalletName: addres1.json

WalletName: addres2.json

WalletName: addres3.json

#### 3.2.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Existed wallet name | 钱包名已存在 |
| The number of password digits is less than 6 | 密码长度小于6位 |
| Password mismatch | 两次输入密码不一致 |
| Password error | 密码错误 |
| Recommended backup keystore | 建议备份钱包文件 |
| Invalid parameter | 无效的的参数 |
| Invalid command | 无效的命令 |
| Create failure | 创建失败 |
| Create success | 创建成功 |
| Incorrect password format | 密码格式不正确 |
| Incorrect name format | 钱包名称格式不正确 |
| Password required | 密码不能为空 |
| XX: command not found | 命令不存在 |

### 3.2.2 查看本地钱包

#### 3.2.2.1描述

前置条件：

1. 正常安装PlatON CLI工具

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令查看本地钱包
3. 在命令行展示
   1. 格式如下：

Name:xxxxxxxxxxxxx ,address: xxxxxxxxxxxxxxx

#### 3.2.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account check --address <walletName.json> 指定名称查询

命令：platoncli account check --address <walletaddres> 指定地址查询

命令：platoncli account check 查询全部

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** | **备注** |
| account | check | --address,-d | optional | 具体查看的钱包文件名称或者address，不填写具体参数查询默认目录下全部钱包文件 | 名称查询需要带后缀；  若是用户通过其他工具生成的钱包需要按命名规则修改名称，否在只能通过地址查询 |

#### 3.2.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件；查询名称超过长度； |
| Bad suffix | 根据名称查询后缀错误 |
| invalid parameter | 参数错误 |
| No wallet file in the directory，Please create a wallet file first | 默认路径下无钱包文件，请先生成钱包文件 |
| XX: command not found | 命令不存在 |

### 3.2.3 删除钱包

#### 3.2.3.1描述

前置条件：

1. 正常安装的PlatON CLI

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
   1. 支持根据address和name.json查询钱包文件
   2. 若未找到钱包文件，提示用户
   3. 若找到提示用户，是否备份钱包文件，删除后无法恢复
3. 输入所要删除的钱包文件密码
   1. 两次密码确认
4. 用户通过Y/N确认是否删除，提示用户是否备份钱包文件，删除后无法恢复
   1. Y，删除钱包
   2. N，删除失败
5. 并将操作记入日志
   1. 格式：

[操作时间]:Operation：delete wallet ,Target：钱包文件名称

#### 3.2.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account delete --address <walletName.json> 指定名称删除

命令：platoncli account delete --address <walletAddress> 指定地址删除

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | delete | --address,-d | must | 指定钱包文件名称  .json  或者钱包文件地址删除对应钱包文件 |

**删除钱包文件**

platoncli account delete --address testAccount1.json

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

It cannot be recovered after deletion. It is recommended to delete after backup

Continue to delete Wallet Y/N?

Y

Delete successful

#### 3.2.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| It cannot be recovered after deletion. It is recommended to delete after backup | 提示用户，是否备份钱包文件，删除后无法恢复 |
| Password mismatch | 两次输入密码不一致 |
| Password required | 密码不能为空 |
| Password error | 密码错误 |
| Delete successful | 删除成功 |
| Delete failed | 删除失败 |
| XX: command not found | 命令不存在 |

### 3.2.4 钱包备份

#### 3.2.4.1描述

前置条件：

1. 正常安装的PlatON CLI
2. 持有钱包以及正确的钱包密码

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入钱包备份命令
   1. 根据名称.json或者address在默认目录下找到所需备份钱包

i 找到钱包输入两次钱包密码进行

备份密码错误提示用户并备份失败

ii 未找到钱包提示用户

1. 备份完成
2. 操作记入日志
   1. 格式：

[操作时间]:Operation：backup wallet ,Target：钱包文件名称

#### 3.2.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account backups --type mnemonic --address <walletName.json>指定名称备份

命令：platoncli account backups --type private --address <walletAddress>指定地址备份

命令：platoncli account backups -t m -d <walletName.json>指定名称备份

命令：platoncli account backups -t p -d <walletAddress>指定地址备份

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | backups | --type,-t | must | 选择备份方式，助记词或者私钥  --type mnemonic 或者--type privatekey  参数支持简写 --type m,--type p |
| --address,-d | must | 选择备份钱包，通过名称.json或者地址 |

**助记词备份钱包文件**

platoncli account backups --type mnemonic --address testAccount1.json

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

Backup Wallet name：xxxxxxxxx.json

Backup Address:xxxxxxxxxxxxxxxxxxxxx

Mnemonic code：cloth tortoise elephant scrub shrug deal dizzy keen idle glide issue

write this mnemonic phrase in a safe place.

It is the important way to recover your account if you ever forget your password.

**私钥备份钱包文件**

platoncli account backups --type mnemonic --address testAccount1.json

Enter a passphrase to encrypt your key to disk: \*\*\*\*\*\*\*\*

Repeat the passphrase: \*\*\*\*\*\*\*\*

Backup Wallet name：xxxxxxxxx.json

Backup Address:xxxxxxxxxxxxxxxxxxxxx

Private key：xxxxxxxxxxxxxxxxxxxxx

It is the important way to recover your account if you ever forget your password.

#### 3.2.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| Password mismatch | 两次输入密码不一致 |
| Password required | 密码不能为空 |
| Password error | 密码错误 |
| Backup successful | 备份成功 |
| Backup failed | 备份失败 |
| XX: command not found | 命令不存在 |

### 3.2.5 恢复私钥

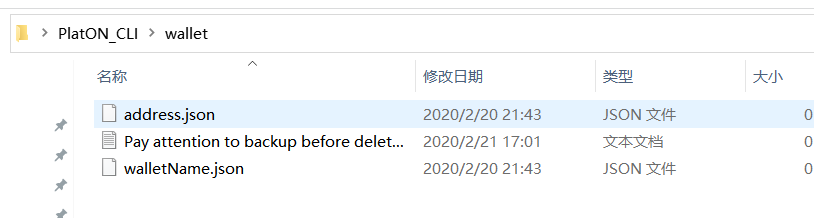
#### 3.2.5.1描述

前置条件：

1. 正常安装的PlatON CLI
2. 正确的助记词或者正确的私钥

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令进入恢复
3. 根据命令输入参数是通过何种方式恢复
   1. 助记词恢复
   2. 私钥
4. 输入两次密码
   1. 两次不一致，提示用户
   2. 密码错误，恢复失败
5. 成功恢复钱包到默认目录下，默认已address.json命名



默认生成钱包的目录

#### 3.2.5.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account recovery --type mnemonic 通过助记词恢复

命令：platoncli account recovery --type private 通过私钥恢复

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | recovery | --type,-t | must | 通过何种方式恢复  --type mnemonic 或者--type privatekey  参数支持简写 --type m,--type p |

#### 3.2.5.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Password mismatch | 两次输入的密码不匹配 |
| Incorrect password format | 密码格式不正确 |
| Existed wallet name | 账户名已存在 |
| Mnemonic phrase cannot be recognized | 无效助记词 |
| Private key cannot be parsed | 无效钱包私钥 |
| XX: command not found | 命令不存在 |
| Password error | 密码错误 |
| The number of password digits is less than 6 | 密码长度小于6位 |

### 3.2.6 修改钱包密码

#### 3.2.6.1描述

前置条件：

1. 正常安装的PlatON CLI
2. 默认路径下须有对应的钱包文件
3. 原始钱包密码

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入修改钱包文件密码命令
3. 根据用户输入参数扎到对应钱包文件
   1. 未找到钱包文件提示用户，文件未找到
   2. 找到钱包文件，进入下一步
4. 输入原密码
   1. 错误提示用户钱包文件密码错误，若错误提供第二次输入机会，第三次错误则结束流程，修改密码失败
   2. 成功，则进入下一步
5. 输入两边新密码
   1. 两次密码不一致提示用户，修改失败
   2. 成功则完成密码修改
6. 用户键入Y/N是否进入备份流程
7. 用户键入Y，进入备份流程

i 默认已助记词备份钱包文件，

1. 用户键入N，提示用户安全因素请备份钱包

#### 3.2.6.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account modify --address <walletName.json>指定名称钱包文件修改密码

命令：platoncli account modify --address <walletAddress>指定地址钱包文件修改密码

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | modify | --address,-d | must | 通过名称.json或者地址指定修改的钱包文件 |

**修改钱包文件密码**

platoncli account modify --address testAccount1

Enter the old passphrase for verification ：\*\*\*\*\*\*\*\*

WalletName: xxxxxxxxx.json

PublicKey: xxxxxxxxxxxxxxxxxxxxxx

Enter new password： \*\*\*\*\*\*\*\*

Repeat the new passphrase: \*\*\*\*\*\*\*\*

Success

#### 3.2.6.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Password mismatch | 两次输入的密码不匹配 |
| Incorrect password format | 密码格式不正确 |
| Success | 修改成功 |
| fail | 修改失败 |
| Password error | 密码错误 |
| The number of password digits is less than 6 | 密码长度小于6位 |

### 3.2.7 离线签名

#### 3.2.6.1描述

前置条件：

1. 离线环境下的PlatON CLI 工具
2. 在线环境下的PlatON CLI工具
3. 在线环境下指定a钱包的，生成的待签名交易
4. a钱包在离线环境下，并明确钱包密码

步骤：

1. 打开在线环境下的PlatON CLI工具
2. 输入命令指定钱包地址，根据需要生成转账或者合约调用的待签名交易
3. 打开离线环境下的PlatON CLI工具
4. 输入离线签名命令，提供参数，在线环境下生成的交易数据
5. 根据参数指定签名钱包
6. 解析待签名交易
   1. 判断from地址是否和参数指定的地址一致

i 若一致，则签名，打印信息并生成完整的签名数据，提供复制

ii 若不一致，则提示交易from地址与指定钱包不一致，请确认

1. 输入钱包密码
   1. 密码错误提示
   2. 成功则进行下一步，交易签名
2. 签名完成
3. 并记入操作日志

模板：

[操作时间]:Operation：offlineSign ,Target：钱包文件名称，to：xxxxxx，value：xxxx

打印信息模板：

1. 转账交易

From：

To：

Value：

ChainI: <根据待签名交易中解析读取出>

1. 合约调用交易

From：

To：

Contract：

Function：

Param: 此为具体调用合约的参数的json字符串显示

#### 3.2.7.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli account sign --address <walletName.json>--data xxxxxxxxxxxxx指定名称钱包文件离线签名

命令：platoncli account sign --address <walletAddress> --data xxxxxxxxxxxxx定地址钱包文件离线签名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | sign | --data ，-d | must | 待签名数据 |
| --address,-d | 指定签名钱包文件  名称.json或者地址 |

#### 3.2.6.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 签名成功 |
| fail | 签名失败 |
| From address mismatch | 待签名交易解析from和签名地址不一致 |
| Analytical exception | 解析待签名交易异常 |
| Password error | 密码错误 |

### 3.2.8查询钱包余额

#### 3.2.8.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入查询余额命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Address:xxxxxxxxxxxxxxxx

Balance:xxxxxxxxxxxxxxxx

#### 3.2.8.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli account getBalance --address xxxxxxxxxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| account | getBalance | --address，-d | must | 具体查询地址 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.2.8.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.3 交易操作

**说明：**发送在线交易和待签名交易生成时

若CLI工具目录下存在初始化的节点

则根据Nodepubkey，找到node\_config.json中的数据，例如chainId；

若CLI工具目录下不存在节点，但是已经配置远程节点地址；

若只有一个配置则默认读取，若存在多个远程节点地址配置，需要明确指明，否则报错

若离线CLI工具，无配置交易发送节点信息，则报错，无法生成待签名文件

### 3.3.1常规转账交易

#### 3.3.1.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 网络通畅
3. 钱包文件
4. 正确的钱包文件密码
5. 账户余额大于转账金额

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入转账命令
3. 回车后解析param参数模板并展示，且输入密码
4. 根据参数判断是离线交易还是在线交易
   1. 离线交易则生成待签名数据，流程结束
   2. 在线交易则需要用户输入密码
5. 判断密码是否正确
   1. 密码错误，提示用户
   2. 密码正确，发送交易
6. 交易发送成功，并返回交易hash
7. 并记入日志

在线

[操作时间]:Operation：transfer ,from：钱包文件名称，to：xxxxxx，value：xxxx

Type：online，txhash：xxx ， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：transfer ,from：钱包文件名称，to：xxxxxx，value：xxxx

Type：offline

#### 3.3.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli tx transfer --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件在线转账交易

命令：platoncli tx transfer --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件离线转账交易

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| Tx | transfer | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

**转账交易的参数模板内容如下**

命令：platoncli tx transfer --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"from": "",

"to": "",

"value": 0

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |
| --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** |
| String | must | from | 交易发送方 |
| String | must | to | 交易接收方 |
| BigInteger | optional | gas | 本次交易gas用量上线 |
| BigInteger | optional | gasPrice | Gas价格 |
| BigInteger | must | value | 转账金额，单位LAT |
| String | optional | data | 上链数据 |

**在线转账交易**

platoncli tx transfer --address <walletAddress> --param xxxxxxxxxxxxx --type online

Param：{"from": "","to": "","value": 0}

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

#### 3.3.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| From address mismatch | 待签名交易解析from和签名地址不一致 |
| Analytical exception | 解析待签名交易异常 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.3.2根据交易hash查询交易信息

#### 3.3.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入查询根据交易hash查询交易信息命令
3. 发送命令
4. 返回查询结果

查询返回结构：

blockHash:"0xc48a3a2575a25d20cef32a43d95f8f6cbabd3a99edf6fabf73f3104d59fbeec9",

blockNumber: 2508,

from: "0x2e95e3ce0a54951eb9a99152a6d5827872dfb4fd",

gas: 210000,

gasPrice: 1000000000,

hash: "0x4d8e0bd8700763811b53d9047f87517ff7a63c74daa35c11a45ccae1b5ff4489",

input: "0xc9880000000000000000",

nonce: 6,

transactionIndex: 0,

value: 单位LAT

未查询到，返回结构，数值为空

#### 3.3.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli tx getTransaction -h xxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| tx | getTransaction | --hash,-h | must | String类型，具体查询区块hash |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.3.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.3.3根据交易hash查询交易回执信息

#### 3.3.3.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

blockHash: "0xc48a3a2575a25d20cef32a43d95f8f6cbabd3a99edf6fabf73f3104d59fbeec9",

blockNumber: 2508,

contractAddress: null,

cumulativeGasUsed: 21168,

from: "0x2e95e3ce0a54951eb9a99152a6d5827872dfb4fd",

gasUsed: 21168,

logs: [],

status: succes/fail,

to: "0xceca295e1471b3008d20b017c7df7d4f338a7fba",

transactionHash: "0x4d8e0bd8700763811b53d9047f87517ff7a63c74daa35c11a45ccae1b5ff4489",

transactionIndex: 0

未查询到，返回结构，数值为空

#### 3.3.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

Platoncli tx getTransaction -h xxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| tx | getTransaction | --hash,-h | must | String类型，具体查询区块hash |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.3.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.3.4发送已签名交易数据

#### 3.3.4.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 正确的钱包文件签名数据

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入发送签名数据交易
   1. 解析数据中的交易并展示

展示模板

转账交易

From：

To：

Value：

合约调用交易

From：

To：

Contract：

Function：

Param: 此为具体调用合约的参数的json字符串显示

1. 发送命令
2. 返回交易hash
3. 打印日志

日志模板：

[操作时间]:Operation：send\_offline ,from：钱包文件名称，txhash：xxx ，Transaction content：{“根据解析展示具体交易模板”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

#### 3.3.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli tx send\_offline --data xxxxxxxxxxxxx 发送签名完成后的交易数据

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| tx | send\_offline | --date, -d | must | 已签名的交易data数据 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.3.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.4 质押相关

**说明：**发送在线交易和待签名交易生成时

若CLI工具目录下存在初始化的节点

则根据Nodepubkey，找到node\_config.json中的数据，例如chainId；

若CLI工具目录下不存在节点，但是已经配置远程节点地址；

若只有一个配置则默认读取，若存在多个远程节点地址配置，需要明确指明，否则报错

若离线CLI工具，无配置交易发送节点信息，则报错，无法生成待签名文件

### 3.4.1 创建验证人

#### 3.4.1.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令查看最新的质押参数
3. 命令：platoncli staking create --template
4. 提示当前用户工作目录<[提示内容](#_3.3.2.3 提示语&异常)>所操作的节点Y/N是否继续
   1. Y，进行下一步
   2. N，返回重新输入命令
5. [提示用户](#_3.3.2.3 提示语&异常)质押门槛是可治理参数，请关注platScan区块链浏览器，具体提示内容请看 提示语&异常
6. 命令支持质押的参数json字符串和json文件
   1. 用户根据实际需要将参数json填写正确
7. 执行命令
   1. 解析命令判断是在线交易还是离线交易
      1. 在线交易，需要检测当前节点进程是否运行中
   2. 判断质押金额是否超过质押门槛
   3. 命令中若有重复的参数与param参数相同，以param参数为准
   4. 离线签名的时候，from地址也需要和param参数地址保持一致，否则签名失败
   5. 检查param参数的的必填性和正确性
8. 发送交易
9. 返回交易hash
10. 并记入日志

日志模板：

在线

[操作时间]:Operation：staking ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：staking ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.4.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：platoncli staking create --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件创建验证人

命令：platoncli staking create --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件创建验证人

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | create | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名， |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置，而目录下没有节点，则报错；若没有节点，但是有多个配置节点，提示用户指明交易发送节点 |

命令：platoncli tx staking --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

查询模板带出字段中的文字

例子：

{

"type": 0,

"benefitAddress": "收益地址",

"nodeId": "",自动读取当前工作目录下nodeId

"externalId": "keybase账户公钥",

"nodeName": "节点名称",

"website": "节点的第三方主页",

"details": "节点描述",

"amount": 0.00,

"delegateRewardPer": 0

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | type | 0: 自由金额，1: 锁仓金额 | 数字长度为1，只能填入1或者0 |
| String | must | benefitAddress | 收益账户,用于接受出块奖励和质押奖励的收益账户，支持0x和不带0x | 带0x前缀长度为42位的十六进制数；  不带0x前缀长度为40位的十六进制数；  不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过42位 |
| String | must | nodeId | 节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | optional  建议填写 | externalId | externalId 外部Id，目前为keybase账户公钥 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过16位 |
| String | must | nodeName | 节点名称 | 支持中文、字母、数字，长度不得超过20位 |
| String | optional  建议填写 | website | 节点的第三方主页 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，长度不限 |
| String | optional  建议填写 | details | 节点的描述 | 支持中文、字母、数字，长度不限 |
| BigInteger | must | amount | 质押金额，单位LAT，质押门槛1000000LAT | 只能输入数字 |
| BigInteger | must | delegateRewardPer | 委托所得到的奖励分成比例，整数，比如传入1 = 0.001% ，10000=100% | 只能输入数字 |

**在线创建验证人**

platoncli tx staking --address <walletAddress> --param xxxxxxxxxxxxx --type online

Param：{"nodeId": "","amount": "","website":""}

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线创建验证人**

platoncli tx staking --address <walletAddress> --param xxxxxxxxxxxxx --type offline

Param：{"nodeId": "","amount": "","website": ""}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.5.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| Fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Not exceeding the stake threshold | 未超过质押门槛 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| No running nodes detected | 未检测到节点进程 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |
| The current working directory is: xxxxx, continue Y/N | 当前工作目录为:xxxxx，是否继续Y/N |
| StakeThreshold is a governable parameter，please check PlatScan | 质押门槛是可治理参数，具体门槛请查看PlatSacn区块链浏览器 |

### 3.4.2 修改质押信息

#### 3.4.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 节点已质押成功
4. 钱包文件，正确的钱包文件密码
5. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入修改质押信息命令
3. 解析命令中参数
   1. 参数是否完整
   2. Param参数中各项的必填性正确性
4. 发送命令
5. 返回交易hash
6. 打印日志

日志模板：

在线

[操作时间]:Operation：updateStaking ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：updateStaking ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.4.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking update --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件修改质押信息

命令：

platoncli staking update --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件修改质押信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | update | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置，而目录下没有节点，则报错；若没有节点，但是有多个配置节点，提示用户指明交易发送节点 |

命令：platoncli staking update --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

调用查询模板，将该几点的原有数据带出，显示，基于工作目录定位节点

意味着，调用查询模板的命名需要去查询该节点的信息，并将信息回填；

例子：

{

"benefitAddress": "",

"nodeId": "",

"externalId": "",

"nodeName": "",

"website": "",

"details": "",

~~"perNodeName": "",~~

"delegateRewardPer": 0

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | optional | benefitAddress | 收益账户,用于接受出块奖励和质押奖励的收益账户 | 带0x前缀长度为42位的十六进制数；  不带0x前缀长度为40位的十六进制数；  不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过42位 |
| String | must | nodeId | 节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | optional | externalId | externalId 外部Id，目前为keybase账户公钥 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过16位 |
| String | optional | nodeName | 节点名称 | 支持中文、字母、数字，长度不得超过20位 |
| String | optional | website | 节点的第三方主页 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，长度不限 |
| String | optional | details | 节点的描述 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，长度不限 |
| BigInteger | must | delegateRewardPer | 委托所得到的奖励分成比例，整数，比如传入1 = 0.001% ，10000=100% | 只能输入数字 |

**在线修改验证人信息**

platoncli staking update --address <walletAddress> --param xxxxxxxxxxxxx Param：{"nodeId": "","website":"" }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线修改验证人信息**

platoncli staking update --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"nodeId": "","website": ""}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.4.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |

### 3.4.3 增持质押

#### 3.4.3.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 节点已质押成功
4. 钱包文件，正确的钱包文件密码
5. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入增持质押命令
3. [文字提示](#_3.3.4.3 提示语&异常)用户增持质押的门槛是可治理参数需要关注具体的值，具体提示内容请看 提示语&异常
4. 解析命令中参数
   1. 参数是否完整
   2. Param参数中各项的必填性正确性
5. 发送命令

E. 返回交易hash

F. 打印日志

日志模板：

在线

[操作时间]:Operation：addStaking ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：addStaking ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.4.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking increase --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件增持质押

命令：

platoncli staking increase --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件增持质押

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | increase | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置，而目录下没有节点，则报错；若没有节点，但是有多个配置节点，提示用户指明交易发送节点 |

命令：platoncli staking increase --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"type": 0,

"nodeId": "",自动读取当前工作目录下nodeId

"amount": 0.00

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | type | 0: 自由金额，1: 锁仓金额 | 数字长度为1，只能填入1或者0 |
| String | must | nodeId | 节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| BigInteger | must | amount | 增持的金额单位LAT/ATP，最低增持门槛为10LAT/ATP | 只能填入数字 |

**在线增持质押**

platoncli staking increase --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"nodeId": "","amount":0.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线增持质押**

platoncli staking increase --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"nodeId": "","amount":0 ... }

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.4.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |
| The increase staking threshold is a governable parameter, please pay attention to platscan | 增持质押门槛是可治理参数，请关注 PlatScan |

### 3.4.4 退出验证人

#### 3.4.4.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 节点已质押成功
4. 钱包文件，正确的钱包文件密码
5. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入退出验证人命令
3. 解析命令中参数
4. 参数是否完整
5. Param参数中各项的必填性正确性
6. 发送命令
7. 返回交易hash
8. 提示用户退出质押人锁定周期是可治理参数，具体提示内容请看 提示语&异常

F. 打印日志

日志模板：

在线

[操作时间]:Operation：unStaking ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：unStaking ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.4.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking unStaking --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件退出验证人

命令：

platoncli staking unStaking --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件退出验证人

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | unStaking | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置，而目录下没有节点，则报错；若没有节点，但是有多个配置节点，提示用户指明交易发送节点 |

命令：platoncli staking unStaking --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"nodeId": ""

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | nodeId | 节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |

**在线退出验证人**

platoncli staking unStaking --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"nodeId": "" }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

Unstaking , token locking period is a governable parameter，please check PlatScan

**离线退出验证人**

platoncli staking unStaking --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"nodeId": ""}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.4.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |
| Unstaking , token locking period is a governable parameter，please check PlatScan | 退出质押，token锁定周期是可治理参数，请查看PlatScan |

### 3.4.5查询当前共识周期的验证人列表

#### 3.4.5.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

{

[BenefitAddress: "", 用于接受出块奖励和质押奖励的收益账户string

Details :"",节点的描述 string

NodeId:"", 被质押的节点Id string

NodeName:"", 被质押节点的名称 string

Released: 0,发起质押账户的自由金额的锁定期质押的LAT BigInteger

ReleasedHes: 0,发起质押账户的自由金额的犹豫期质押的LAT BigInteger

RestrictingPlan: 0,发起质押账户的锁仓金额的锁定期质押的LAT BigInteger

RestrictingPlanHes: 0,发起质押账户的锁仓金额的犹豫期质押的LAT BigInteger

Shares: 0,当前候选人总共质押加被委托的LAT数目 BigInteger

StakingAddress: "",发起质押时使用的账户 string

StakingBlockNum: 0,发起质押时的区块高度 BigInteger

StakingEpoch:0,当前变更质押金额时的结算周期 BigInteger

Status:“节点可用”,候选人的状态，0: 节点可用，1: 节点不可用 ，2:节点出块率低但没有达到移除条件的，4:节点的von不足最低质押门槛(只有倒数第三bit为1)，8:节点被举报双签，16:节点出块率低且达到移除条件(倒数第五位bit为1); 32: 节点主动发起撤销 string

ValidatorTerm:0,验证人的任期 BigInteger

Website:"",节点的第三方主页 string

delegateTotal:0,节点被委托的生效总数量 BigInteger

delegateRewardTotal:0,候选人当前发放的总委托奖励 BigInteger

nextRewardPer:0,下一个结算周期奖励分成比例 BigInteger

rewardPer:0,当前结算周期奖励分成比例 BigInteger

],[],[]

}

#### 3.4.5.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking getValidatorList

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| Staking | getValidatorList | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.4.5.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.4.6查询当前结算周期的验证人列表

#### 3.4.6.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

{

[BenefitAddress: "", 用于接受出块奖励和质押奖励的收益账户string

Details :"",节点的描述 string

NodeId:"", 被质押的节点Id string

NodeName:"", 被质押节点的名称 string

Released: 0,发起质押账户的自由金额的锁定期质押的LAT BigInteger

ReleasedHes: 0,发起质押账户的自由金额的犹豫期质押的LAT BigInteger

RestrictingPlan: 0,发起质押账户的锁仓金额的锁定期质押的LAT BigInteger

RestrictingPlanHes: 0,发起质押账户的锁仓金额的犹豫期质押的LAT BigInteger

Shares: 0,当前候选人总共质押加被委托的LAT数目 BigInteger

StakingAddress: "",发起质押时使用的账户 string

StakingBlockNum: 0,发起质押时的区块高度 BigInteger

StakingEpoch:0,当前变更质押金额时的结算周期 BigInteger

Status:“节点可用”,候选人的状态，0: 节点可用，1: 节点不可用 ，2:节点出块率低但没有达到移除条件的，4:节点的von不足最低质押门槛(只有倒数第三bit为1)，8:节点被举报双签，16:节点出块率低且达到移除条件(倒数第五位bit为1); 32: 节点主动发起撤销

ValidatorTerm:0,验证人的任期BigInteger

Website:"",节点的第三方主页 string

delegateTotal:0,节点被委托的生效总数量 BigInteger

delegateRewardTotal:0,候选人当前发放的总委托奖励 BigInteger

nextRewardPer:0,下一个结算周期奖励分成比例 BigInteger

rewardPer:0,当前结算周期奖励分成比例 BigInteger

],[],[]

}

#### 3.4.6.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking getVerifierList

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| Staking | getVerifierList | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.4.6.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.4.7查询所有实时候选人列表

#### 3.4.7.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

{

[BenefitAddress: "", 用于接受出块奖励和质押奖励的收益账户string

Details :"",节点的描述 string

NodeId:"", 被质押的节点Id string

NodeName:"", 被质押节点的名称 string

Released: 0,发起质押账户的自由金额的锁定期质押的LAT BigInteger

ReleasedHes: 0,发起质押账户的自由金额的犹豫期质押的LAT BigInteger

RestrictingPlan: 0,发起质押账户的锁仓金额的锁定期质押的LAT BigInteger

RestrictingPlanHes: 0,发起质押账户的锁仓金额的犹豫期质押的LAT BigInteger

Shares: 0,当前候选人总共质押加被委托的LAT数目 BigInteger

StakingAddress: "",发起质押时使用的账户 string

StakingBlockNum: 0,发起质押时的区块高度 BigInteger

StakingEpoch:0,当前变更质押金额时的结算周期 BigInteger

Status:“节点可用”,候选人的状态，0: 节点可用，1: 节点不可用 ，2:节点出块率低但没有达到移除条件的，4:节点的von不足最低质押门槛(只有倒数第三bit为1)，8:节点被举报双签，16:节点出块率低且达到移除条件(倒数第五位bit为1); 32: 节点主动发起撤销

ValidatorTerm:0,验证人的任期BigInteger

Website:"",节点的第三方主页 string

delegateTotal:0,节点被委托的生效总数量 BigInteger

delegateRewardTotal:0,候选人当前发放的总委托奖励 BigInteger

nextRewardPer:0,下一个结算周期奖励分成比例 BigInteger

rewardPer:0,当前结算周期奖励分成比例 BigInteger

],[],[]

}

#### 3.4.7.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query getCandidateList

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getCandidateList | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.4.7.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.4.8根据nodeId查询节点质押信息

#### 3.4.8.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

BenefitAddress: "", 用于接受出块奖励和质押奖励的收益账户string

Details :"",节点的描述 string

NodeId:"", 被质押的节点Id string

NodeName:"", 被质押节点的名称 string

Released: 0,发起质押账户的自由金额的锁定期质押的LAT BigInteger

ReleasedHes: 0,发起质押账户的自由金额的犹豫期质押的LAT BigInteger

RestrictingPlan: 0,发起质押账户的锁仓金额的锁定期质押的LAT BigInteger

RestrictingPlanHes: 0,发起质押账户的锁仓金额的犹豫期质押的LAT BigInteger

Shares: 0,当前候选人总共质押加被委托的LAT数目 BigInteger

StakingAddress: "",发起质押时使用的账户 string

StakingBlockNum: 0,发起质押时的区块高度 BigInteger

StakingEpoch:0,当前变更质押金额时的结算周期 BigInteger

Status:“节点可用”,候选人的状态，0: 节点可用，1: 节点不可用 ，2:节点出块率低但没有达到移除条件的，4:节点的von不足最低质押门槛(只有倒数第三bit为1)，8:节点被举报双签，16:节点出块率低且达到移除条件(倒数第五位bit为1); 32: 节点主动发起撤销

ValidatorTerm:0,验证人的任期BigInteger

Website:"",节点的第三方主页 string

delegateTotal:0,节点被委托的生效总数量 BigInteger

delegateRewardTotal:0,候选人当前发放的总委托奖励 BigInteger

nextRewardPer:0,下一个结算周期奖励分成比例 BigInteger

rewardPer:0,当前结算周期奖励分成比例 BigInteger

未查询到，返回结构，数值为空

#### 3.4.8.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking getCandidateInfo --nodeid<nodeid>

命令：

platoncli staking getCandidateInfo -n <nodeid>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | getCandidateInfo | --nodeid,-n | must | 质押节点id |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.4.8.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.4.9查询当前结算周期的质押奖励

#### 3.4.9.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Reward :0 当前结算周期的质押奖励

未查询到，返回结构，数值为空

#### 3.4.9.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli staking getStakingReward

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| staking | getStakingReward | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.4.9.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.5 委托相关

**说明：**发送在线交易和待签名交易生成时

若CLI工具目录下存在初始化的节点

则根据Nodepubkey，找到node\_config.json中的数据，例如chainId；

若CLI工具目录下不存在节点，但是已经配置远程节点地址；

若只有一个配置则默认读取，若存在多个远程节点地址配置，需要明确指明，否则报错

若离线CLI工具，无配置交易发送节点信息，则报错，无法生成待签名文件

### 3.5.1 委托

#### 3.5.1.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入委托命令
3. [文字提示](#_3.3.6.3 提示语&异常)用户委托的门槛是可治理参数需要关注具体的值，具体提示内容请看 提示语&异常
4. 解析命令中参数
5. 参数是否完整
6. Param参数中各项的必填性正确性
7. 发送命令
8. 返回交易hash

F. 打印日志

日志模板：

在线

[操作时间]:Operation：delegate ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：delegate ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.5.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli delegate new --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件委托

命令：

platoncli delegate new --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件委托

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| delegate | new | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli delegate new --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"type": 0,

"nodeId": "",

"amount": 0.00

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | type | 0: 自由金额，1: 锁仓金额 | 数字长度为1，只能填入1或者0 |
| String | must | nodeId | 委托的节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| BigInteger | must | amount | 委托金额，单位LAT | 只能输入数字 |

**在线委托**

platoncli delegate new --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"nodeId": "" }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线委托**

platoncli delegate new --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"nodeId": ""}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.5.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| Not exceeding the delegate threshold | 未超过委托门槛 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |
| Delegation threshold is a governable parameter. Please check platscan | 委托门槛是可治理参数，请查看PlatScan |

### 3.5.2 减持/撤销委托

#### 3.5.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 已对用户目标节点发起过委托

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入减持/撤销委托命令
3. [文字提示](#_3.3.6.3 提示语&异常_1)用户委托的门槛是可治理参数需要关注具体的值，具体提示内容请看 提示语&异常
4. 解析命令中参数
5. 参数是否完整
6. Param参数中各项的必填性正确性
7. 发送命令
8. 返回交易hash
9. 打印日志

日志模板：

在线

[操作时间]:Operation：unDelegate ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：unDelegate ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.5.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

Platoncli delegate unDelegate --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件减持/撤销委托

命令：

platoncli delegate unDelegate --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件减持/撤销委托

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| delegate | unDelegate | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli delegate unDelegate --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"stakingBlockNum": 0,

"nodeId": "",

"amount": 0.00

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | stakingBlockNum | 接受委托节点的质押快高 | 只支持输入数字 |
| String | must | nodeId | 委托的节点id，16进制 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| BigInteger | must | amount | 撤销委托的金额，单位LAT | 只支持输入数字 |

**在线撤销委托**

platoncli delegate unDelegate --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"nodeId": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线撤销委托**

platoncli delegate unDelegate --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"nodeId": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.5.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |
| Withdrawal threshold is a governable parameter,The remaining token will be redeemed if it is lower than the threshold  ,Please check platscan | 赎回金额是可治理参数，所剩金额低于门槛时将全部赎回，请查看PlatScan |

### 3.5.3提取委托奖励

#### 3.5.3.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 提取地址目标做过委托交易
6. 同时委托交易有收益

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入提取委托收益命令
3. 解析命令中参数
   1. Param参数中各项的必填性正确性
   2. 参数完整性
4. 发送命令
5. 返回交易hash
6. 打印日志

日志模板：

在线

[操作时间]:Operation：withdrawDelegateRward ,from：钱包文件名称，Type：online，txhash：xxx ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：withdrawDelegateRward ,from：钱包文件名称，Type：offline

#### 3.5.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli delegate withdrawDelegateReward --address <walletName.json> 指定名称钱包文件提取委托奖励

命令：

platoncli delegate withdrawDelegateReward --address <walletAddress> --offline定地址钱包文件提取委托奖励

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| delegate | withdrawDelegateReward | --address,-d | must | 发送交易地址或者名称.json  提取已此地址委托的收益  同时也是收益发放的地址 |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

**在线提取委托收益**

platoncli delegate withdrawDelegateReward--address <walletAddress>

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线提取委托收益**

platoncli delegate withdrawDelegateReward--address <walletAddress> --offline

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.5.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.5.4查询当前账户地址所委托的节点的NodeId和质押Id

#### 3.5.4.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

{

[

Address:””, 委托人账户地址 string

NodeId:””,验证人节点id string

StakingBlockNumber:0 发起质押的区块高度 BigInteger

],[],[]

}

未查询到，返回结构，数值为空

#### 3.5.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

Platoncli delegate getRelatedListByDelAddr --addrss <walletName.json>

命令：

platoncli delegate getRelatedListByDelAddr -d <walletaddress>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| delegate | getRelatedListByDelAddr | --address，-d | must | 委托人账户地址 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.5.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.5.5查询账户在各节点未提取委托奖励

#### 3.5.5.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 解析param
4. 确保参数的正确性和必填性
5. 发送命令
6. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

Data:

[

{

NodeId:””，节点id，string

StakingNumber:0, 节点的质押块高，BigInteger

Reward:0, 未领取的委托收益，BigInteger

}

]

未查询到，返回结构，数值为空

#### 3.5.5.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli delegate getDelegateReward --param xxxxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getDelegateReward | --param,-p | must | 查询的参数 |
| --template | must | 参看具体的参数模板 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli delegate getDelegateReward --template 查看查询参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"address": "" ，

"nodeIDs": ["id1","id2"]

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | address | 要查询的账户地址 | 带0x前缀长度为42位的十六进制数；  不带0x前缀长度为40位的十六进制数；  不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过42位 |
| Array | optional | nodeIDs | 要查询的节点id的string数组，如果为空则查询账户委托的所有节点 | 不输入则查询全部，不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |

#### 3.5.5.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.6 治理相关

**说明：**发送在线交易和待签名交易生成时

若CLI工具目录下存在初始化的节点

则根据Nodepubkey，找到node\_config.json中的数据，例如chainId；

若CLI工具目录下不存在节点，但是已经配置远程节点地址；

若只有一个配置则默认读取，若存在多个远程节点地址配置，需要明确指明，否则报错

若离线CLI工具，无配置交易发送节点信息，则报错，无法生成待签名文件

### 3.6.1 提交参数/升级/取消提案

#### 3.6.1.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 节点为验证节点
6. 正确的在指定的github上提出提案

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入提交文本/参数/升级/取消提案命令
3. 解析命令中参数
4. 参数是否完整
5. Param参数中各项的必填性正确性
6. 发送命令
7. 返回交易hash
8. 打印日志

日志模板：

在线

[操作时间]:Operation：submit [具体的提案类型] Proposal ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：submit [具体的提案类型] Proposal ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.6.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government submitProposal --address <walletName.json> --param xxxxxxxxxxxxx --module CancelProposal 指定名称钱包文件提交文本/参数/升级/取消提案

命令：

platoncli government submitProposal --address <walletAddress> --param xxxxxxxxxxxxx --offline --moduleTextProposal 定地址钱包文件提交文本/参数/升级/取消提案

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | submitProposal | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --module,-m | must | 提案类型包括  取消提案CancelProposal,  参数提案ParamProposal,  升级提案  VersionProposal, |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli government submitProposal --template 查看交易参数<--param>中具体参数模板，此为特色，会列举出四种不同提案的参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

**升级提案例子：**

{

"validator": "",

"pIDID": "",

"endVotingRound": 0.00,

"newVersion": 0

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | validator | 提交提案的验证人，nodeId | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | must | pIDID | PIPID | 不支持输入中文 |
| BigInteger | must | endVotingRound | 投票共识轮数量。说明：假设提交提案的交易，被打包进块时的共识轮序号时round1，则提案投票截止块高，就是round1 + endVotingRounds这个共识轮的第230个块高（假设一个共识轮出块250，ppos揭榜提前20个块高，250，20都是可配置的 ），其中0 < endVotingRounds <= 4840（约为2周，实际论述根据配置可计算），且为整数） | 只支持输入数字 |
| BigInteger | must | newVersion | 升级版本 | 只支持输入数字 |

**参数提案例子：**

{

"verifier": "",

"pIDID": "",

"module": "",

"name": "",

"newValue": ""

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | verifier | 提交提案的验证人 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | must | pIDID | PIPID | 不支持输入中文 |
| String | must | module | 参数模块 | 不支持输入中文 |
| String | must | name | 升级版本 | 不支持输入中文 |
| String | must | newValue | 参数新值 | 不支持输入中文 |

**取消提案例子：**

{

"verifier": "",

"pIDID": "",

"endVotingRound": 0.00,

"canceledProposalID": ""

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | verifier | 提交提案的验证人 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | must | pIDID | PIPID | 不支持输入中文 |
| BigInteger | must | endVotingRound | 投票共识轮数量。说明：假设提交提案的交易，被打包进块时的共识轮序号时round1，则提案投票截止块高，就是round1 + endVotingRounds这个共识轮的第230个块高（假设一个共识轮出块250，ppos揭榜提前20个块高，250，20都是可配置的 ），其中0 < endVotingRounds <= 4840（约为2周，实际论述根据配置可计算），且为整数） | 只支持输入数字 |
| String | must | canceledProposalID | 待取消的提案ID | 不支持输入中文 |

**在线提交取消提案**

platoncli government submitProposal --address <walletAddress> --param xxxxxxxxxxxxx --module CancelProposal

Param：{"": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线提交升级提案**

platoncli government submitProposal --address <walletAddress> --param xxxxxxxxxxxxx --offline --module VersionProposal

Param：{"": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.6.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.2 提案投票

#### 3.6.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 只能是验证人或者候选人

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入提案投票命令
3. 解析命令中参数
4. 参数是否完整
5. Param参数中各项的必填性正确性
6. 发送命令
7. 返回交易hash
8. 打印日志

日志模板：

在线

[操作时间]:Operation：vote ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：vote ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.6.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government vote --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件给提案投票

命令：

platoncli government vote --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件给提案投票

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | vote | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli government vote --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"verifier": "",

"proposalId": "",

"option": ""

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | verifier | 声明的节点，只能是验证人/候选人 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |
| String | must | proposalId | 提案ID，所投的提案交易的hash | 带0x，66位十六进制数；不带0x64位十六进制数 |
| String | must | option | 投票类型；  YEAS 赞成票  NAYS 反对票  ABSTENTIONS 弃权票 | 只支持英文，YEAS，NAYS ， ABSTENTIONS ，其他单词报错，不区分大小写 |

**在线给提案投票**

platoncli government vote --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"verifier": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线提案投票**

platoncli government vote --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"verifier": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.6.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.3版本声明

#### 3.6.3.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 只能是验证人或者候选人
6. 节点超过升级提案的中的结束快高前未投票

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入版本声明命令
3. 解析命令中参数
4. 参数是否完整
5. Param参数中各项的必填性正确性
6. 发送命令
7. 返回交易hash
8. 打印日志

日志模板：

在线

[操作时间]:Operation：declareVersion ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：declareVersion ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.6.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government declareVersion --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件给提案投票

命令：

platoncli government declareVersion --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件给提案投票

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | declareVersion | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli government declareVersion --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"activeNode": ""

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | activeNode | 声明的节点id，只能是验证人/候选人 | 不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过128位 |

**在线版本声明**

platoncli government declareVersion --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"activeNode": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线版本声明**

platoncli government declareVersion --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"activeNode": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.6.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.6查询提案列表

#### 3.6.6.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

{

[

proposalId:””, 提案ID String

proposer:””, 提案节点ID String

proposalType:”文本提案”， 提案类型， 1：文本提案； 2：升级提案；3参数提案 4取消提案 int

piPid:””, 提案PIPID String

submitBlock:””, 提交提案的块高 BigInteger

endVotingBlock:””, 提案投票结束的块高 BigInteger

newVersion:””, 升级版本 BigInteger

toBeCanceled:””, 提案要取消的升级提案ID BigInteger

activeBlock:””, 如果投票通过）生效块高（endVotingBlock + 20 + 4\*250 < 生效块高 <= endVotingBlock + 20 + 10\*250） BigInteger

],[],[],

}

未查询到，返回结构，数值为空

#### 3.6.6.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government listProposal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | listProposal | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.6.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.4举报双签

#### 3.6.4.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅
5. 只能是验证人或者候选人
6. 发现区块出现双签

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入举报双钱命令
3. 解析命令中参数
4. 参数是否完整
5. Param参数中各项的必填性正确性
6. 发送命令
7. 返回交易hash
8. 打印日志

日志模板：

在线

[操作时间]:Operation：reprotDoubleSign ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：reprotDoubleSign ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.6.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government reportDoubleSign --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件举报双签

命令：

platoncli government reportDoubleSign --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件举报双签

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | reportDoubleSign | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli government reportDoubleSign --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"type": 0,

"**data**": "",

"verify": ""

}

**data模板包含3种证据类型，分别是：duplicatePrepare、duplicateVote、duplicateViewchange**

**每种类型包含多个证据，所以是数组结构，解析时需注意**

{

"duplicatePrepare": [{

"prepareA": {

"epoch": 1, 共识轮epoch值 必填 int

"viewNumber": 1, 共识轮view值 必填 int

"blockHash": "", 区块hash 必填 string

"blockNumber": 1, 区块number 必填 int

"blockIndex": 1,区块在一轮view中的索引值 必填 int

"validateNode": {

"index": 0, 验证人在一轮epoch中的索引值 必填 int

"address": "",验证人地址 必填 string

"nodeId": "", 验证人nodeID 必填 string

"blsPubKey": "" 验证人bls公钥 必填string

},

"signature": "" 消息签名 必填 string

},

"prepareB": {

"epoch": 1, 共识轮epoch值 必填 int

"viewNumber": 1, 共识轮view值 必填 int

"blockHash": "",区块hash 必填 string

"blockNumber": 1,区块number 必填 int

"blockIndex": 1, 区块在一轮view中的索引值 必填 int

"validateNode": {

"index": 0, 验证人在一轮epoch中的索引值 必填 int

"address": "", 验证人地址 必填 string

"nodeId": "",验证人nodeID 必填 string

"blsPubKey": "" 验证人bls公钥 必填string

},

"signature": "" 消息签名 必填 string

}

}],

"duplicateVote": [{

"voteA": {

"epoch": 1,共识轮epoch值 必填 int

"viewNumber": 1,共识轮view值 必填 int

"blockHash": "",区块hash 必填 string

"blockNumber": 1,区块number 必填 int

"blockIndex": 1,区块在一轮view中的索引值 必填 int

"validateNode": {

"index": 0, 验证人在一轮epoch中的索引值 必填 int

"address": "",验证人地址 必填 string

"nodeId": "",验证人nodeID 必填 string

"blsPubKey": ""验证人bls公钥 必填string

},

"signature": ""消息签名 必填 string

},

"voteB": {

"epoch": 1, 共识轮epoch值 必填 int

"viewNumber": 1,共识轮view值 必填 int

"blockHash": "",区块hash 必填 string

"blockNumber": 1, 区块number 必填 int

"blockIndex": 1, 区块在一轮view中的索引值 必填 int

"validateNode": {

"index": 0, 验证人在一轮epoch中的索引值 必填 int

"address": "",验证人地址 必填 string

"nodeId": "",验证人nodeID 必填 string

"blsPubKey": ""验证人bls公钥 必填string

},

"signature": ""消息签名 必填 string

}

}],

"duplicateViewchange": [{

"viewA": {

"epoch": 1, 验证人在一轮epoch中的索引值 必填 int

"viewNumber": 1,共识轮view值 必填 int

"blockHash": "",区块hash 必填 string

"blockNumber": 1,区块number 必填 int

"validateNode": {

"index": 0,验证人在一轮epoch中的索引值 必填 int

"address": "",验证人地址 必填 string

"nodeId": "",验证人nodeID 必填 string

"blsPubKey": ""验证人bls公钥 必填string

},

"signature": "", 消息签名 必填 string

},

"viewB": {

"epoch": 1,验证人在一轮epoch中的索引值 必填 int

"viewNumber": 1,共识轮view值 必填 int

"blockHash": "",区块hash 必填 string

"blockNumber": 1,区块number 必填 int

"validateNode": {

"index": 0,验证人在一轮epoch中的索引值 必填 int

"address": "",验证人地址 必填 string

"nodeId": "",验证人nodeID 必填 string

"blsPubKey": ""验证人bls公钥 必填string

},

"signature": "" 消息签名 必填 string

}

}]

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | Int | 代表双签类型，  1：prepareBlock，  2：prepareVote，  3：viewChange | 只支持输入数字 |
| String | must | String | 单个证据的json值，格式参照RPC接口Evidences | String类型长度不限 |

**在线举报双签**

platoncli government reportDoubleSign --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"data": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线举报双签**

platoncli government reportDoubleSign --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"data": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.6.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.5根据提案id查询提案信息

#### 3.6.5.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

proposalId:””, 提案ID String

proposer:””, 提案节点ID String

proposalType:”文本提案”， 提案类型， 1：文本提案； 2：升级提案；3参数提案 4取消提案 int

piPid:””, 提案PIPID String

submitBlock:””, 提交提案的块高 BigInteger

endVotingBlock:””, 提案投票结束的块高 BigInteger

newVersion:””, 升级版本 BigInteger

toBeCanceled:””, 提案要取消的升级提案ID BigInteger

activeBlock:””, 如果投票通过）生效块高（endVotingBlock + 20 + 4\*250 < 生效块高 <= endVotingBlock + 20 + 10\*250） BigInteger

未查询到，返回结构，数值为空

#### 3.6.5.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government getProposal --proposalId xxxxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | getProposal | --proposalId,--pid | must | 查询提案的proposalid,参数类型String |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.5.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.7查询提案结果

#### 3.6.7.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

proposalId:””, 提案ID String

yeas:0, 赞成票票数 BigInteger

nays:0， 反对票票数 BigInteger

abstentions:0, 弃权票票数 BigInteger

accuVerifiers:0, 在整个投票期内有投票资格的验证人总数 BigInteger

status:”投票中”, 提案状态 string

- Voting：0x01，投票中

- Pass：0x02，投票通过

- Failed：0x03，投票失败

- PreActive：0x04，（升级提案）预生效

- Active：0x05，（升级提案）生效

- Canceled：0x06，被取消

未查询到，返回结构，数值为空

#### 3.6.7.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government getTallyResult -pid xxxxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | getTallyResult | --proposalId,--pid | must | nil |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.7.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.8查询节点的链生效版本

#### 3.6.8.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Versiong: 0 版本信息

未查询到，返回结构，数值为空

#### 3.6.8.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government getActiveVersion

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | getActiveVersion | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.8.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.9查询最新的治理参数值

#### 3.6.9.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Module: “” 参数模块，string

Name:”” 参数名称，string

未查询到，返回结构，数值为空

#### 3.6.9.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government getGovernParamValue

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | getGovernParamValue | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.9.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.10查询治理参数列表

#### 3.6.10.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

GovernParam：[

ParamItem: {

Module:””, 参数模块，string

Name:””, 参数名称，string

Desc:””,参数说明，string

}, 参数项，对象

ParamValue:{

Stalevalue:””,旧参数值，string

Value:””, 参数值，string

ActiveBlock:0 块高，BigInteger (当前块高>=ActiveBLock，将取Value;否则取StaleValue)

}参数值，对象

]

未查询到，返回结构，数值为空

#### 3.6.10.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government listGovernParam -m

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | listGovernParam | --module,-m | must | 查询具体的参数模块，输入“”空串，表示查询所有治理参数 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.10.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.11查询提案的累计可投票人数

#### 3.6.11.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

voteTurnout:0,累积可投票人数，BigInteger

Yeas:0,赞成票数，BigInteger

Nays:0,反对票数，BigInteger

Abstentions:0,弃权票数，BigInteger

未查询到，返回结构，数值为空

#### 3.6.11.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government getAccuVerifiersCount -h xxxxxxxxxxxx -p xxxxxxxxxxxx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | getAccuVerifiersCount | --proposalId,--pid | must | 查询的提案的id，string |
| --hash,-h | must | 当前最新区块的hash |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.11.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.6.12查询节点是否已被举报过多签

#### 3.6.12.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Status:Y/N,是否被举报过多签，string

DuplicateSignHash:””,举报交易的hash，当staus为N是，显示0x

未查询到，返回结构，数值为空

#### 3.6.12.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli government checkDuplicateSign -t xxxxxxxxxxxx -d xxxxxxxxxxxx -n 100

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| government | checkDuplicateSign | --type,-t | must | 双签类型：  prepareBlock;  prepareVote;  viewChange;  String |
| --address,-d | must | 举报的节点地址，string |
| --number,-n | must | 多签的块高，BigIntger |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.6.12.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.7 链相关基本信息查询

### 3.7.2查询查询当前最高块高

#### 3.7.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入查询当前最高块号命令
3. 发送命令
4. 返回查询结果

查询返回结构：

BlockNubmer：xxxxxx

#### 3.7.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query blockNumber

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | blockNumber | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.7.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.7.3根据区块块高查询区块信息

#### 3.7.3.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入查询根据区块number查询block信息命令
3. 发送命令
4. 返回查询结果

查询返回结构：

difficulty: 0,

extraData: "",

gasLimit: ,

gasUsed: 0,

hash: "",

logsBloom: "",

miner: "",

nonce: "",

number: 100,

parentHash: "",

receiptsRoot: "",

size: 726,

stateRoot: "",

timestamp: ,

totalDifficulty: 0,

transactions: [],

transactionsRoot: ""

#### 3.7.3.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query getBlockByNumber -n 100

查询默只带有交易的hash列表<sdk中boolean参数false>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getBlockByNumber | --number,-n | must | int类型，具体查询区块的块高 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.7.3.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.7.4根据区块hash查询区块信息

#### 3.7.4.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入查询根据区块hash查询block信息命令
3. 发送命令
4. 返回查询结果

查询返回结构：

difficulty: 0,

extraData: "",

gasLimit: ,

gasUsed: 0,

hash: "",

logsBloom: "",

miner: "",

nonce: "",

number: 100,

parentHash: "",

receiptsRoot: "",

size: 726,

stateRoot: "",

timestamp: ,

totalDifficulty: 0,

transactions: [],

transactionsRoot: ""

未查询到，返回结构，数值为空

#### 3.7.4.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query getBlockByHash -h xxxxxx

查询默只带有交易的hash列表<sdk中boolean参数false>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getBlockByHash | --hash,-h | must | String类型，具体查询区块hash |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.7.4.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.7.7查询当前结算周期的区块奖励

#### 3.7.7.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Reward :0 当前结算周期的区块奖励

未查询到，返回结构，数值为空

#### 3.7.7.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query getPackageReward

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getPackageReward | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.7.7.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.7.9查询打包区块的平均时间

#### 3.7.9.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

Time:0 打包区块的平均时间

未查询到，返回结构，数值为空

#### 3.7.9.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli query getAvgPackTime

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| query | getAvgPackTime | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.7.9.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.8 锁仓相关

**说明：**发送在线交易和待签名交易生成时

若CLI工具目录下存在初始化的节点

则根据Nodepubkey，找到node\_config.json中的数据，例如chainId；

若CLI工具目录下不存在节点，但是已经配置远程节点地址；

若只有一个配置则默认读取，若存在多个远程节点地址配置，需要明确指明，否则报错

若离线CLI工具，无配置交易发送节点信息，则报错，无法生成待签名文件

### 3.8.1创建锁仓计划

#### 3.8.1.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 钱包文件，正确的钱包文件密码
4. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入创建锁仓计划命令
3. 解析命令中参数
   1. 参数是否完整
   2. Param参数中各项的必填性正确性
4. 发送命令
5. 返回交易hash
6. 打印日志

日志模板：

在线

[操作时间]:Operation：createRestrictingPlan ,from：钱包文件名称，Type：online，txhash：xxx ，param：{“显示param中参数”:””} ， blocNmuber：xxx， nodeIP: xxxxx，nodeRpcport:xxxxx

离线

[操作时间]:Operation：createRestrictingPlan ,from：钱包文件名称，Type：offline，param：{“显示param中参数”:””};

#### 3.8.1.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli hedge createRestrictingPlan --address <walletName.json> --param xxxxxxxxxxxxx 指定名称钱包文件创建锁仓计划

命令：

platoncli hedge createRestrictingPlan --address <walletAddress> --param xxxxxxxxxxxxx --offline定地址钱包文件建锁仓计划

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| hedge | createRestrictingPlan | --param，-p | must | 交易参数json字符串，或者交易参数json文件路径 |
| --address,-d | must | 发送交易地址或者名称.json |
| --offline,-o | optional | 在线交易或者离线交易 offline 不输入默认为在线交易，并生成二维码图片放置在桌面上，提供ATON离线扫码签名 |
| --template | optional | 查看交易参数模板，与其他参数共存没有效果，单独执行查看 |
| --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

命令：platoncli hedge createRestrictingPlan --template 查看交易参数<--param>中具体参数模板

<--param xxxx.json 或者 --param json字符串，非必填参数，字段可不写入json中>

例子：

{

"account":"",

"plans": [

{

"epoch": 0,

"amount": 0.00

}

]

}

查看模板同样以以下方式展示，可不用表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| String | must | account | 锁仓计划释放到的仓库 | 带0x前缀长度为42位的十六进制数；  不带0x前缀长度为40位的十六进制数；  不支持输入中文，只能输入数字和英文字母，不区分大小写，最长不得超过42位 |
| Array | must | plans | 锁仓计划列表(数组) |  |

**Plans**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类型** | **必填性** | **参数名称** | **参数解释** | **参数限制** |
| BigInteger | must | epoch | 表示结算周期的倍数。与每个结算周期出块数的乘积表示在目标区块高度上释放锁定的资金。如果 account 是激励池地址， 那么 period 值是 120（即，30\*4） 的倍数。另外，period,每周期的区块数至少要大于最高不可逆区块高度 | 只支持输入数字 |
| BigInteger | must | amount | 表示目标区块上待释放的金额。 | 只支持输入数字 |

**在线创建锁仓计划**

platoncli hedge createRestrictingPlan --address <walletAddress> --param xxxxxxxxxxxxx

Param：{"activeNode": "",.... }

Enter password： \*\*\*\*\*\*\*

Success

Txhash：xxxxxxxxx

**离线创建锁仓计划**

platoncli hedge createRestrictingPlan --address <walletAddress> --param xxxxxxxxxxxxx --offline

Param：{"activeNode": "",....}

Data:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 待签名数据

Success

#### 3.8.1.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| Not inquired | 未找到到钱包文件； |
| XX: command not found | 命令不存在 |
| Password required | 密码为空 |
| Incorrect password format | 密码格式不正确 |
| Success | 发送成功 |
| fail | 发送失败 |
| Password error | 密码错误 |
| Analytical exception | 解析交易信息异常 |
| Parameter error | Param参数错误 |
| Parameter XX is Empty | Param检测中必填参数为空 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

### 3.8.2获取锁仓计划

#### 3.8.2.1描述

前置条件：

1. 正常运行的PlatON CLI工具
2. 节点正常运行
3. 网络通畅

步骤：

1. A用户指定文件夹下使用platoncli工具
2. 用户输入命令
3. 发送命令
4. 返回查询结果

查询返回结构：

由于数据过多，展示内容字段如下，但是在命令行界面展示为json字符串，用户自行复制格式化查看数据

RestrictingItem: 保存锁仓信息对象

{

Balance:0，锁仓余额，BigInteger

Pledge:0,质押/抵押金额，BigInteger

Debt:0,欠释放金额，BigInteger

RestrictingInfo: 锁仓分录信息

[

{

blockNumber:0,释放区块高度，BigInteger

Amount:0, 释放金额，BigInteger

}，

{

blockNumber:0,释放区块高度，BigInteger

Amount:0, 释放金额，BigInteger

}

]

}

未查询到，返回结构，数值为空

#### 3.8.2.2 命令&数据项说明

参数提供两种方式，全名和简写；

命令：

platoncli hedge GetRestrictingInfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **必填性** | **参数说明** |
| hedge | GetRestrictingInfo | --config,-c | optional | 指定交易发送的ip和端口的配置文件，全局配置network文件中若配置，可以通过指定配置中名称获取ip和prot，若未填写network配置 |

#### 3.8.2.3 提示语&异常

|  |  |
| --- | --- |
| 提示语 | 场景 |
| XX: command not found | 命令不存在 |
| Success | 发送成功 |
| fail | 发送失败 |
| The currently running node is not checked out | 当加入网络后，没有指定配置文件发送交易的节点 |
| Incorrect config configuration file | Config配置文件解析失败 |

## 3.9 帮助

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块** | **操作** | **命令参数** | **参数说明** | |
| account | new | --name,-n | Set wallet file name | |
| --batch，-b | Number of wallets generated in batch | |
| multiple | -name,-n | Create multisignature wallet file | |
| --batch，-b | Number of wallets generated in batch | |
| delete | --address,-d | Specify wallet file name or address | |
| check | --address,-d | It is not required to query the list, but to view the specific wallet | |
| modify | --address,-d | Specify wallet file name or address | |
| backups | --type,-t | Backup wallet mode,provide mnemonic and privatekey | |
| --address,-d | Specify wallet file name or address | |
| sign | --date，-d | Unsigned data | |
| --address,-d | Specify wallet file name or address | |
| recovery | --type,-t | Recovery mode,provide mnemonic and privatekey | |
| getBalance | --address，-d | Find Address balance | |
| --config，-c | Send transaction network profile path | |
| Wallet management module,For wallet generation and management | | | |
| node | init | --path,-p | Create the path of workspace working directory and download the corresponding platform version of Platon client | |
| build | --type,-t | Connection network type,provide main or test or private | |
| start | nil | Start or restart | |
| stop | nil | Stop node process | |
| reset | --clear,-c | -c, Clear all node data, including the nodekey & blskey. Without this parameter, only data will be cleared | |
| upgrade | --path，-p | Platon cli File path includes files | |
| rollback | --data,-d | Rollback data File path includes files | |
| --client,-c | Rollback platon Client File path includes files | |
| generateNodeKey | --path,-p | NodeKey File path includes files | |
| generateblsKey | --path,-p | blsKey File path includes files | |
| check | nil | Node version detection | |
| info | nil | Query node information in current working directory | |
| blockexport | --path,-p | Export block storage path | |
| blockinpute | --path,-p | Inpute block storage path | |
| move | --original,-o | Source path | |
| --target,t | Target path | |
| --movetype,-m | Migration type,  Nodekey&blsKey or data | |
| Node management module,Create deployment nodes, start and stop nodes, update nodes and other operations | | | |
| tx | transfer | | --param，  -p | Transfer transaction parameters |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View -- param template |
| --config，-c | Send transaction network profile path |
| send\_offline | | --data, -d | Signed transaction data |
| --config，-c | Send transaction network profile path |
| getTransaction | | --hash,-h | Transaction hash |
| --config，-c | Send transaction network profile path |
| getTransactionReceipt | | --hash,-h | Transaction hash |
| --config，-c | Send transaction network profile path |
| Transaction management module | | | |
| staking | create | | --param，-p | Staking param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View Staking param template |
| --config，-c | Send transaction network profile path |
| update | | --param,-p | updateStaking param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View updateStaking param template |
| --config，-c | Send transaction network profile path |
| increase | | --param,-p | addStaking param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View addStaking param template |
| --config，-c | Send transaction network profile path |
| unStaking | | --param,-p | unStaking param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View unStaking param template |
| --config，-c | Send transaction network profile path |
| getValidatorList | | --config，-c | Send transaction network profile path |
| getVerifierList | | --config，-c | Send transaction network profile path |
| getCandidateList | | --config，-c | Send transaction network profile path |
| getCandidateInfo | | --nodeid,-n | Staking node id |
| --config，-c | Send transaction network profile path |
| getStakingReward | | --config，-c | Send transaction network profile path |
| Stake management module | | | |
| delegate | new | | --param,-p | Delegate param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View delegate param template |
| --config，-c | Send transaction network profile path |
| unDelegate | | --param,-p | unDelegate param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View unDelegate param template |
| --config，-c | Send transaction network profile path |
| withdrawDelegateReward | | --offline,-o | online or offline |
| --address,-d | from address |
| --config，-c | Send transaction network profile path |
| getRelatedListByDelAddr | | -address，-d | Staking node wallet address |
| --config，-c | Send transaction network profile path |
| getDelegateReward | | --param,-p | Delegate nodeId list |
| --template | View Delegate param template |
| --config，-c | Send transaction network profile path |
| delegate management module | | | |
| government | submitProposal | | --param,-p | submitProposal param String json or json file |
| --address,-d | from address |
| --module,-m | Proposal type ,CancelProposal,ParamProposal,VersionProposal,TextProposal |
| --offline,-o | Generate offline transactions to be signed |
| --template | View submitProposal param template |
| --config，-c | Send transaction network profile path |
| vote | | --param,-p | Vote param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View Vote param template |
| --config，-c | Send transaction network profile path |
| declareVersion | | --param,-p | declareVersion param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View declareVersion param template |
| --config，-c | Send transaction network profile path |
| reportDoubleSign | | --param,-p | reportDoubleSign param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View reportDoubleSign param template |
| --config，-c | Send transaction network profile path |
| listProposal | | --config，-c | Send transaction network profile path |
| getProposal | | --proposalId,--pid | Proposal id |
| --config，-c | Send transaction network profile path |
| getTallyResult | | --proposalId,--pid | Proposal id |
| --config，-c | Send transaction network profile path |
| getActiveVersion | | --config，-c | Send transaction network profile path |
| getGovernParamValue | | --config，-c | Send transaction network profile path |
| listGovernParam | | --module,-m | Module name ,if --module null,find all module |
| --config，-c | Send transaction network profile path |
| getAccuVerifiersCount | | --proposalId,--pid | proposalID |
| --hash,-h | blockhash |
| --config，-c | Send transaction network profile path |
| checkDuplicateSign | | --type,-t | DuplicateSignType  prepareBlock;  prepareVote;  viewChange; |
| --address,-d | Informant address |
| --number,-n | DuplicatesSing block number |
| --config，-c | Send transaction network profile path |
| government management module | | | |
| query | blockNumber | | --config，-c | Send transaction network profile path |
| getBlockByNumber | | --number,-n | Block number |
| --config，-c | Send transaction network profile path |
| getBlockByHa  sh | | --hash,-h | Block hash |
| --config，-c | Send transaction network profile path |
| getPackageReward | | --config，-c | Send transaction network profile path |
| getAvgPackTime | | --config，-c | Send transaction network profile path |
| Chaindata Query management module | | | |
| hedge | createRestrictingPlan | | --param,-p | createRestrictingPlan param String json or json file |
| --address,-d | from address |
| --offline,-o | Generate offline transactions to be signed |
| --template | View createRestrictingPlan param template |
| --config，-c | Send transaction network profile path |
| GetRestrictingInfo | | --config，-c | Send transaction network profile path |
| Hedge management module | | | |
| help | Query command | | Get operation command list | help |