

节点程序参数说明

2018-08-01

- 节点程序参数说明
 - [nodepc 程序参数说明](#)
 - [potato::bnet_plugin 插件参数说明](#)
 - [potato::chain_plugin 插件参数说明](#)
 - [potato::http_plugin 插件参数说明](#)
 - [potato::mongo_db_plugin 插件参数说明](#)
 - [potato::net_plugin 插件参数说明](#)
 - [potato::producer_plugin 插件参数说明](#)
 - [potato::wallet_plugin 插件参数说明](#)
 - [默认配置文件示例](#)

除了打印输出参数，大部分都可以在配置文件里配置，直接由控制台传入的参数会覆盖配置文件中的参数。参数括号内的为默认值。

nodepc 程序参数说明

参数	说明
-h / --help	打印帮助文档
-v / --version	打印版本号（GIT 提交记录）
--print-default-config	打印默认配置文件模板
-d / --data-dir arg	指定数据文件目录
--config-dir arg	指定 config.ini 配置文件所在目录
-c / --config arg (=config.ini)	指定配置文件名（默认名为 config.ini）
-l / --logconf arg (=logging.json)	指定日志配置文件名及路径（默认名为 logging.json）
--plugin arg	启用插件，可重复

potato::bnet_plugin 插件参数说明

bnet_plugin 插件主要功能是同步区块

参数	说明
--bnet-endpoint arg (=0.0.0.0:4321)	监听地址
--bnet-follow-irreversible arg (=0)	请求其它节点的不可逆块（默认 0）
--bnet-threads arg	用于处理网络消息的线程数
--bnet-connect arg	远程节点地址; 可以根据需要多个节点组成一个网络

参数	说明
--bnet-no-trx	请求其他节点没有挂起的事务

potato::chain_plugin 插件参数说明

chain_plugin 插件主要功能是区块读写、执行合约，默认开启。

参数	说明
--blocks-dir arg ("blocks")	指定区块数据文件所在目录（绝对路径或相对于应用程序 data-dir）
--checkpoint arg	Pairs of [BLOCK_NUM,BLOCK_ID] that should be enforced as checkpoints.
--abi-serializer-max-time-ms arg (=15000)	重置 ABI 序列化最大时间（默认 15000ms）
--chain-state-db-size-mb arg (=1024)	链状态数据库的最大大小（以 MiB 为单位）
--chain-state-db-guard-size-mb arg (=128)	当链状态数据库中剩余的可用空间低于此大小时（在 MiB 中），安全地关闭节点。
--reversible-blocks-db-size-mb arg (=340)	可逆块数据库的最大大小（以 MiB 为单位）
--reversible-blocks-db-guard-size-mb arg (=2)	当可逆块数据库中剩余的可用空间低于此大小时（在 MiB 中），安全地关闭节点。
--contracts-console	打印合约的输出到控制台
--actor-whitelist arg	帐户已添加到 actor 白名单（可重复）
--actor-blacklist arg	帐户已添加到 actor 黑名单（可重复）
--contract-whitelist arg	合约帐户已添加到合约白名单（可重复）
--contract-blacklist arg	合约帐户已添加到合约黑名单（可重复）
--action-blacklist arg	action 添加到 action 黑名单（可重复）
--key-blacklist arg	公钥添加到不应包含在权限中的密钥黑名单中（可重复）
--read-mode arg (=speculative)	数据库读取模式 ("speculative" or "head"). 在"speculative"模式中，数据库包含直到块头所做的更改以及尚未包含在区块链中的事务所做的更改。 在"head"模式下，数据库包含直到当前 head 块的更改。

以下参数不能在配置文件中设置，必须由控制台传入执行。

参数	说明
--genesis-json arg	从文件中读取 genesis state

参数	说明
--genesis-timestamp arg	覆盖 genesis state 文件中的初始时间戳
--print-genesis-json	从 blocks.log 中提取 genesis state 为 JSON，打印到控制台，然后退出
--extract-genesis-json arg	从 blocks.log 中提取 genesis state 为 JSON，写入指定文件，然后退出
--fix-reversible-blocks	如果该数据库处于错误状态，则恢复可逆块数据库
--force-all-checks	不要跳过重放不可逆块时可以跳过的任何检查
--replay-blockchain	清除链状态数据库并重放所有块
--hard-replay-blockchain	清除链状态数据库，从块日志中恢复尽可能多的块，然后重放这些块
--delete-all-blocks	清除链状态数据库和块日志
--truncate-at-block arg (=0)	在此块编号处停止硬重放/阻止日志恢复（如果设置为非零数字）
--import-reversible-blocks arg	用从指定文件导入的块替换可逆块数据库，然后退出
--export-reversible-blocks arg	以可移植格式将可逆块数据库导出到指定文件然后退出

potato::http_plugin 插件参数说明

http_plugin 插件主要功能是提供 http/https 服务，默认开启。

参数	说明
--http-server-address arg (=127.0.0.1:8888)	侦听传入 http 连接的本地 IP 和端口; 设置为空以禁用。
--https-server-address arg	侦听传入 https 连接的本地 IP 和端口; 留空以禁用。
--https-certificate-chain-file arg	采用 PEM 格式的带有证书链的文件名，以显示在 https 连接上。
--https-private-key-file arg	采用 PEM 格式的 https 私钥文件。
--access-control-allow-origin arg	指定每个请求返回的 Access-Control-Allow-Origin。
--access-control-allow-headers arg	指定每个请求上返回的 Access-Control-Allow-Header。
--access-control-max-age arg	指定每个请求返回的 Access-Control-Max-Age。
--access-control-allow-credentials	指定每个请求返回的 Access-Control-Allow-Credent。是否允许发送 Cookie
--max-body-size arg (=1048576)	传入 RPC 请求允许的最大 body 大小（以字节为单位）
--verbose-http-errors	将错误日志附加到 HTTP 响应

参数	说明
--http-validate-host arg (=false)	验证 host, 如果设置为 false, 则任何传入的“host”标头都被视为有效
--http-alias arg	此外, 可以多次指定传入 HTTP 请求的“主机”标头的可接受值。默认情况下包括 http/https 服务器地址。

potato::mongo_db_plugin 插件参数说明

mongo_db_plugin 插件主要功能是提供交易信息二次整理存储功能, 方便第三方查询。

参数	说明
-q [--mongodb-queue-size] arg (=1024)	nodepc 和 MongoDB 插件线程之间的目标队列大小。
--mongodb-abi-cache-size arg (=2048)	用于序列化数据的 abi 缓存的最大大小。
--mongodb-wipe	需要同时使用--replay-blockchain, --hard-replay-blockchain 或--delete-all-blocks 来擦除 mongodb。此选项需要防止意外擦除 mongodb。
--mongodb-block-start arg (=0)	如果指定, 那么只有 abi 数据被推送到 mongodb, 直接从指定的块开始。
-m [--mongodb-uri] arg	MongoDB URI 连接字符串, 查看: https://docs.mongodb.com/master/reference/connection-string/ . 如果未指定, 则禁用插件. 如果未在 URI 中指定, 则使用默认数据库'POTATO'. 示例: mongodb://127.0.0.1:27017/POTATO

potato::net_plugin 插件参数说明

net_plugin 插件主要提供 p2p 网络通信、区块同步功能, 默认开启。

参数	说明
--p2p-listen-endpoint arg (=0.0.0.0:9876)	本地 IP 和端口, 用于侦听传入的 p2p 连接。
--p2p-server-address arg	外部可访问的主机: 用于标识此节点的端口。默认为 p2p-listen-endpoint。
--p2p-peer-address arg	要连接的 p2p 节点的公共节点。 根据需要使用多个 p2p-peer-address 选项来组成网络。
--p2p-max-nodes-per-host arg (=1)	来自任何单个 IP 地址的最大客户端节点数
--agent-name arg (="potato Test Agent")	提供的名称用于在 p2p 中标识此节点。

参数	说明
--allowed-connection arg (=any)	允许连接的节点类型。可以是“any”或“producers”或“specified”或“none”。如果'specified', 则必须至少指定一次 peer-key。 如果只是'producer', 则不需要 peer-key。 “producers”和“specified”可以合并。
--peer-key arg	允许连接的可选公钥。（可重复）
--peer-private-key arg	[公钥,私钥]组，（可重复）
--max-clients arg (=25)	接受连接的最大客户端数，使用 0 表示没有限制。
--connection-cleanup-period arg (=30)	清理死连接之前要等待的秒数
--network-version-match arg (=0)	True 要求与 p2p 版本完全匹配。
--sync-fetch-span arg (=100)	同步时从任何单个节点中检索的块数
--max-implicit-request arg (=1500)	在未先发送通知的情况下发送的最大事务或块消息大小
--use-socket-read-watermark arg (=0)	启用实验套接字读取水印优化

potato::producer_plugin 插件参数说明

producer_plugin 插件主要提供区块生产功能，默认开启。

参数	说明
-e [--enable-stale-production]	启用区块生产，即使链是不是最新的。
-x [--pause-on-startup]	在生产暂停的状态下启动此节点。
--max-transaction-time arg (=30)	允许推送事务的代码执行的最长时间
--max-irreversible-block-age arg (= -1)	限制 DPOS 不可逆块的最大时间（以秒为单位），此节点将生成块（使用负值表示无限制）
-p [--producer-name] arg	该节点的生产者的 ID（例如 potatobpa）
--signature-provider arg (=<public-key>=KEY:<private-key>)	签名密钥 <public-key>=KEY:<private-key> public-key 有效的 potato 链公钥 private-key 对应的 potato 链私钥
--keosd-provider-timeout arg (=5)	限制允许将块发送到 keosd 提供程序进行签名的最长时间（以毫秒为单位）

参数	说明
--greylist-account arg	无法访问扩展 CPU / NET 虚拟资源的帐户
--produce-time-offset-us arg (=0)	非最后一个块的偏移产生时间以微秒为单位。 负数导致块越快出现，正数导致块稍后出去
--last-block-time-offset-us arg (=0)	最后一个块的偏移量以微秒为单位产生时间。 负数导致块越快出现，正数导致块稍后出去

potato::wallet_plugin 插件参数说明

wallet_plugin 插件主要私钥生成和保存功能。

参数	说明
--wallet-dir arg (= ".")	钱包文件的路径（绝对路径或相对于应用程序数据目录 data-dir）
--unlock-timeout arg (=900)	解锁钱包超时（以秒为单位）（默认为 900（15 分钟））。钱包将在指定的不活动秒数后自动锁定。
--yubihsm-authkey key_num	启用 YubiHSM 支持

默认配置文件示例

```
# the endpoint upon which to listen for incoming connections (potato::bnet_plugin)
bnet-endpoint = 0.0.0.0:4321

# the number of threads to use to process network messages (potato::bnet_plugin)
# bnet-threads =

# remote endpoint of other node to connect to; Use multiple bnet-connect options
as needed to compose a network (potato::bnet_plugin)
# bnet-connect =

# this peer will request no pending transactions from other nodes
(potato::bnet_plugin)
bnet-no-trx = false

# the location of the blocks directory (absolute path or relative to application
data dir) (potato::chain_plugin)
blocks-dir = "blocks"

# Pairs of [BLOCK_NUM,BLOCK_ID] that should be enforced as checkpoints.
(potato::chain_plugin)
# checkpoint =

# Override default WASM runtime (potato::chain_plugin)
# wasm-runtime =

# Maximum size (in MB) of the chain state database (potato::chain_plugin)
```

```
chain-state-db-size-mb = 1024

# Maximum size (in MB) of the reversible blocks database (potato::chain_plugin)
reversible-blocks-db-size-mb = 340

# print contract's output to console (potato::chain_plugin)
contracts-console = false

# Account added to actor whitelist (may specify multiple times)
(potato::chain_plugin)
# actor-whitelist =

# Account added to actor blacklist (may specify multiple times)
(potato::chain_plugin)
# actor-blacklist =

# Contract account added to contract whitelist (may specify multiple times)
(potato::chain_plugin)
# contract-whitelist =

# Contract account added to contract blacklist (may specify multiple times)
(potato::chain_plugin)
# contract-blacklist =

# Track actions which match receiver:action:actor. Actor may be blank to include
all. Receiver and Action may not be blank. (potato::history_plugin)
# filter-on = *

# PEM encoded trusted root certificate (or path to file containing one) used to
validate any TLS connections made. (may specify multiple times)
# (potato::http_client_plugin)
# https-client-root-cert =

# true: validate that the peer certificates are valid and trusted, false: ignore
cert errors (potato::http_client_plugin)
https-client-validate-peers = 1

# The local IP and port to listen for incoming http connections; set blank to
disable. (potato::http_plugin)
http-server-address = 0.0.0.0:8888

# The local IP and port to listen for incoming https connections; leave blank to
disable. (potato::http_plugin)
https-server-address = 0.0.0.0:443

# Filename with the certificate chain to present on https connections. PEM format.
Required for https. (potato::http_plugin)
https-certificate-chain-file = /etc/letsencrypt/live/jocky123.com/fullchain.pem

# Filename with https private key in PEM format. Required for https
(potato::http_plugin)
https-private-key-file = /etc/letsencrypt/live/jocky123.com/privkey.pem

# Specify the Access-Control-Allow-Origin to be returned on each request.
```

```
(potato::http_plugin)
access-control-allow-origin = *

# Specify the Access-Control-Allow-Headers to be returned on each request.
(potato::http_plugin)
# access-control-allow-headers =

# Specify the Access-Control-Max-Age to be returned on each request.
(potato::http_plugin)
# access-control-max-age =

# Specify if Access-Control-Allow-Credentials: true should be returned on each
request. (potato::http_plugin)
access-control-allow-credentials = false

# The maximum body size in bytes allowed for incoming RPC requests
(potato::http_plugin)
max-body-size = 1048576

# Append the error log to HTTP responses (potato::http_plugin)
verbose-http-errors = false

# If set to false, then any incoming "Host" header is considered valid
(potato::http_plugin)
http-validate-host = 0

# Additionally acceptable values for the "Host" header of incoming HTTP requests,
can be specified multiple times. Includes http/s_server_address by default.
(potato::http_plugin)
# http-alias =

# The actual host:port used to listen for incoming p2p connections.
(potato::net_plugin)
p2p-listen-endpoint = 0.0.0.0:9876

# An externally accessible host:port for identifying this node. Defaults to p2p-
listen-endpoint. (potato::net_plugin)
# p2p-server-address =

# The public endpoint of a peer node to connect to. Use multiple p2p-peer-address
options as needed to compose a network. (potato::net_plugin)
# p2p-peer-address =

# Maximum number of client nodes from any single IP address (potato::net_plugin)
p2p-max-nodes-per-host = 1

# The name supplied to identify this node amongst the peers. (potato::net_plugin)
agent-name = "Potato Test Agent"

# Can be 'any' or 'producers' or 'specified' or 'none'. If 'specified', peer-key
must be specified at least once. If only 'producers', peer-key is not required.
'producers' and 'specified' may be combined. (potato::net_plugin)
allowed-connection = any
```



```
# Optional public key of peer allowed to connect. May be used multiple times.
(potato::net_plugin)
# peer-key =

# Tuple of [PublicKey, WIF private key] (may specify multiple times)
(potato::net_plugin)
# peer-private-key =

# Maximum number of clients from which connections are accepted, use 0 for no
limit (potato::net_plugin)
max-clients = 25

# number of seconds to wait before cleaning up dead connections
(potato::net_plugin)
connection-cleanup-period = 30

# True to require exact match of peer network version. (potato::net_plugin)
network-version-match = 0

# number of blocks to retrieve in a chunk from any individual peer during
synchronization (potato::net_plugin)
sync-fetch-span = 100

# maximum sizes of transaction or block messages that are sent without first
sending a notice (potato::net_plugin)
max-implicit-request = 1500

# Enable block production, even if the chain is stale. (potato::producer_plugin)
enable-stale-production = false

# Start this node in a state where production is paused (potato::producer_plugin)
pause-on-startup = false

# Limits the maximum time (in milliseconds) that is allowed a pushed transaction's
code to execute before being considered invalid (potato::producer_plugin)
max-transaction-time = 30

# Limits the maximum age (in seconds) of the DPOS Irreversible Block for a chain
this node will produce blocks on (use negative value to indicate unlimited)
(potato::producer_plugin)
max-irreversible-block-age = -1

# ID of producer controlled by this node (e.g. inita; may specify multiple times)
(potato::producer_plugin)
producer-name = potato

# (DEPRECATED - Use signature-provider instead) Tuple of [public key, WIF private
key] (may specify multiple times) (potato::producer_plugin)
# private-key =
["PC6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV", "5KQwrPbwdL6PhXujxW37FSSQZ
1JiwsST4cqQzDeyXtP79zkvFD3"]

# Key=Value pairs in the form <public-key>=<provider-spec>
# Where:
```

```

# <public-key>      is a string form of a valid EOSIO public key
#
# <provider-spec>    is a string in the form <provider-type>:<data>
#
# <provider-type>    is KEY, or KEOSD
#
# KEY:<data>         is a string form of a valid EOSIO private key which maps
to the provided public key
#
# KEOSD:<data>       is the URL where keosd is available and the appropriate
wallet(s) are unlocked (potato::producer_plugin)
signature-provider =
PC6MRyAjQq8ud7hVNYcfnVPJqcVpscN5So8BhtHuGYqET5GDW5CV=KEY:5KQwrPbwdL6PhXujxW37FSSQZ
1JiwsST4cqQzDeyXtP79zkvFD3

# Limits the maximum time (in milliseconds) that is allowed for sending blocks to a
keosd provider for signing (potato::producer_plugin)
keosd-provider-timeout = 5

# The queue size between nodepc and SQL DB plugin thread. (potato::sql_db_plugin)
sql_db-queue-size = 256

# Sql DB URI connection string If not specified then plugin is disabled. Default
database 'EOS' is used if not specified in URI. (potato::sql_db_plugin)
# sql_db-uri =

# The path of the wallet files (absolute path or relative to application data dir)
(potato::wallet_plugin)
wallet-dir = "."

# Timeout for unlocked wallet in seconds (default 900 (15 minutes)). Wallets will
automatically lock after specified number of seconds of inactivity. Activity is
defined as any wallet command e.g. list-wallets. (potato::wallet_plugin)
unlock-timeout = 900

# potato key that will be imported automatically when a wallet is created.
(potato::wallet_plugin)
# potato-key =

# Plugin(s) to enable, may be specified multiple times
# plugin =
plugin = potato::producer_plugin
#plugin = potato::producer_api_plugin
plugin = potato::http_plugin
plugin = potato::chain_plugin
plugin = potato::chain_api_plugin
#plugin = potato::history_plugin
#plugin = potato::history_api_plugin
plugin = potato::bnet_plugin
plugin = potato::net_plugin
#plugin = potato::net_api_plugin
plugin = potato::wallet_plugin
plugin = potato::wallet_api_plugin

```

```
#plugin = potato::sql_db_plugin  
#sql_db-uri = mysql://db=potato host=localhost port=3306 user=root  
password='*****'  
  
plugin = potato::mongo_db_plugin  
mongodb-uri = mongodb://127.0.0.1:27017/POTATO
```