NAME

zend - manual page for zend v2.0.14

DESCRIPTION

Zen Daemon version v2.0.14

Usage:

zend [options]

Start Zen Daemon

OPTIONS

-?

This help message

-alerts

Receive and display P2P network alerts (default: 1)

-alertnotify=<cmd>

Execute command when a relevant alert is received or we see a really long fork (%s in cmd is replaced by message)

-blocknotify=<cmd>

Execute command when the best block changes (%s in cmd is replaced by block hash)

-checkblocks=<n>

How many blocks to check at startup (default: 288, 0 = all)

-checklevel=<n>

How thorough the block verification of -checkblocks is (0-4, default: 3)

-conf=<file>

Specify configuration file (default: zen.conf)

-daemon

Run in the background as a daemon and accept commands

-datadir=<dir>

Specify data directory

-disabledeprecation=<version>

Disable block-height node deprecation and automatic shutdown (example: -disabledeprecation=2.0.14)

-exportdir=<dir>

Specify directory to be used when exporting data

-dbcache=<n>

Set database cache size in megabytes (4 to 16384, default: 100)

-loadblock=<file>

Imports blocks from external blk000??.dat file on startup

-maxorphantx=<n>

Keep at most <n> unconnectable transactions in memory (default: 100)

-mempooltxinputlimit=<n>

Set the maximum number of transparent inputs in a transaction that the mempool will accept (default: 0 = no limit applied)

-par=<n>

Set the number of script verification threads (-8 to 16, 0 = auto, <0 = leave that many cores free, default: 0)

-pid=<file>

Specify pid file (default: zend.pid)

-prune=<n>

Reduce storage requirements by pruning (deleting) old blocks. This mode disables wallet support and is incompatible with -txindex. Warning: Reverting this setting requires re-downloading the entire blockchain. (default: 0 = disable pruning blocks, >550 = target size in MiB to use for block files)

-reindex

Rebuild block chain index from current blk000??.dat files on startup

-sysperms

Create new files with system default permissions, instead of umask 077 (only effective with disabled wallet functionality)

-txindex

```
       Maintain a full transaction index, used by the getrawtransaction rpc call (default: 0)

Connection options:

-addnode=<ip>

       Add a node to connect to and attempt to keep the connection open

-banscore=<n>

       Threshold for disconnecting misbehaving peers (default: 100)

-bantime=<n>

       Number of seconds to keep misbehaving peers from reconnecting (default: 86400)

-bind=<addr>

       Bind to given address and always listen on it. Use [host]:port notation for IPv6

-connect=<ip>

       Connect only to the specified node(s)

-discover

       Discover own IP addresses (default: 1 when listening and no -externalip or -proxy)

-dns

       Allow DNS lookups for -addnode, -seednode and -connect (default: 1)

-dnsseed

       Query for peer addresses via DNS lookup, if low on addresses (default: 1 unless -connect)

-externalip=<ip>

       Specify your own public address

-forcednsseed

       Always query for peer addresses via DNS lookup (default: 0)

-listen

       Accept connections from outside (default: 1 if no -proxy or -connect)

-listenonion

       Automatically create Tor hidden service (default: 1)

-maxconnections=<n>

       Maintain at most <n> connections to peers (default: 125)

-maxreceivebuffer=<n>

       Maximum per-connection receive buffer, <n>*1000 bytes (default: 5000)

-maxsendbuffer=<n>

       Maximum per-connection send buffer, <n>*1000 bytes (default: 1000)

-onion=<ip:port>

       Use separate SOCKS5 proxy to reach peers via Tor hidden services (default: -proxy)

-onlynet=<net>

       Only connect to nodes in network <net> (ipv4, ipv6 or onion)

-permitbaremultisig

       Relay non-P2SH multisig (default: 1)

-port=<port>

       Listen for connections on <port> (default: 9033 or testnet: 19033)

-proxy=<ip:port>

       Connect through SOCKS5 proxy

-proxyrandomize

       Randomize credentials for every proxy connection. This enables Tor stream isolation (default: 1)

-seednode=<ip>

       Connect to a node to retrieve peer addresses, and disconnect

-timeout=<n>

       Specify connection timeout in milliseconds (minimum: 1, default: 5000)

-torcontrol=<ip>:<port>
```

```
       Tor control port to use if onion listening enabled (default: 127.0.0.1:9051)

-torpassword=<pass>

       Tor control port password (default: empty)

-tlsvalidate=<0 or 1>

       Connect to peers only with valid certificates (default: 0)

-tlskeypath=<path>

       Full path to a private key

-tlskeypwd=<password>

       Password for a private key encryption (default: not set, i.e. private key will be stored unencrypted)

-tlscertpath=<path>

       Full path to a certificate

-tlstrustdir=<path>

       Full path to a trusted certificates directory

-whitebind=<addr>

       Bind to given address and whitelist peers connecting to it. Use [host]:port notation for IPv6

-whitelist=<netmask>

       Whitelist  peers  connecting  from the given netmask or IP address. Can be specified multiple times. Whitelisted peers
       cannot be DoS banned and their transactions are always relayed, even if they are already in the mempool,  useful  e.g.
       for a gateway

Wallet options:

-disablewallet

       Do not load the wallet and disable wallet RPC calls

-keypool=<n>

       Set key pool size to <n> (default: 100)

-paytxfee=<amt>

       Fee (in ZEN/kB) to add to transactions you send (default: 0.00)

-rescan

       Rescan the block chain for missing wallet transactions on startup

-salvagewallet

       Attempt to recover private keys from a corrupt wallet.dat on startup

-sendfreetransactions

       Send transactions as zero-fee transactions if possible (default: 0)

-spendzeroconfchange

       Spend unconfirmed change when sending transactions (default: 1)

-txconfirmtarget=<n>

       If paytxfee is not set, include enough fee so transactions begin confirmation on average within n blocks (default: 2)

-maxtxfee=<amt>

       Maximum  total  fees (in ZEN) to use in a single wallet transaction; setting this too low may abort large transactions
       (default: 0.10)

-upgradewallet

       Upgrade wallet to latest format on startup

-wallet=<file>

       Specify wallet file (within data directory) (default: wallet.dat)

-walletbroadcast

       Make the wallet broadcast transactions (default: 1)

-walletnotify=<cmd>

       Execute command when a wallet transaction changes (%s in cmd is replaced by TxID)

-zapwallettxes=<mode>

       Delete all wallet transactions and only recover those parts of the blockchain through -rescan on startup (1 = keep  tx
       meta data e.g.  account owner and payment request information, 2 = drop tx meta data)

ZeroMQ notification options:

-zmqpubhashblock=<address>
```

Enable publish hash block in <address>

-zmqpubhashtx=<address>

Enable publish hash transaction in <address>

-zmqpubrawblock=<address>

Enable publish raw block in <address>

-zmqpubrawtx=<address>

Enable publish raw transaction in <address>

Debugging/Testing options:

-debug=<category>

Output debugging information (default: 0, supplying <category> is optional). If <category> is not supplied or if <category> = 1, output all debugging information. <category> can be: addrman, alert, bench, coindb, db, estimatefee, http, libevent, lock, mempool, net, partitioncheck, pow, proxy, prune, rand, reindex, rpc, selectcoins, tor, zmq, zrpc, zrpcunsafe (implies zrpc).

-experimentalfeatures

Enable use of experimental features

-help-debug

Show all debugging options (usage: --help -help-debug)

-logips

Include IP addresses in debug output (default: 0)

-logtimestamps

Prepend debug output with timestamp (default: 1)

-minrelaytxfee=<amt>

Fees (in ZEN/kB) smaller than this are considered zero fee for relaying (default: 0.000001)

-printtoconsole

Send trace/debug info to console instead of debug.log file

-shrinkdebugfile

Shrink debug.log file on client startup (default: 1 when no -debug)

-limitdebuglogsize

Limit the debug.log file size to 10Mb (default: 1 when no -debug)

-testnet

Use the test network

Node relay options:

-datacarrier

Relay and mine data carrier transactions (default: 1)

-datacarriersize

Maximum size of data in data carrier transactions we relay and mine (default: 80)

Block creation options:

-blockminsize=<n>

Set minimum block size in bytes (default: 0)

-blockmaxsize=<n>

Set maximum block size in bytes (default: 2000000)

-blockprioritysize=<n>

Set maximum size of high-priority/low-fee transactions in bytes (default: 1000000)

Mining options:

-gen

Generate coins (default: 0)

-genproclimit=<n>

Set the number of threads for coin generation if enabled (-1 = all cores, default: 1)

-equihashsolver=<name>

Specify the Equihash solver to be used if enabled (default: "default")

-mineraddress=<addr>

Send mined coins to a specific single address

-minetolocalwallet

Require that mined blocks use a coinbase address in the local wallet (default: 1)

RPC server options:

-server

Accept command line and JSON-RPC commands

-rest

Accept public REST requests (default: 0)

-rpcbind=<addr>

Bind to given address to listen for JSON-RPC connections. Use [host]:port notation for IPv6. This option can be specified multiple times (default: bind to all interfaces)

-rpcuser=<user>

Username for JSON-RPC connections

-rpcpassword=<pw>

Password for JSON-RPC connections

-rpcport=<port>

Listen for JSON-RPC connections on <port> (default: 8232 or testnet: 18232)

-rpcallowip=<ip>

Allow JSON-RPC connections from specified source. Valid for <ip> are a single IP (e.g. 1.2.3.4), a network/netmask (e.g. 1.2.3.4/255.255.255.0) or a network/CIDR (e.g. 1.2.3.4/24). This option can be specified multiple times

-rpcthreads=<n>

Set the number of threads to service RPC calls (default: 4)

Metrics Options (only if -daemon and -printtoconsole are not set):

-showmetrics

Show metrics on stdout (default: 1 if running in a console, 0 otherwise)

-metricsui

Set to 1 for a persistent metrics screen, 0 for sequential metrics output (default: 1 if running in a console, 0 otherwise)

-metricsrefreshtime

Number of seconds between metrics refreshes (default: 1 if running in a console, 600 otherwise)

COPYRIGHT

This is experimental software.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit <https://www.openssl.org/> and cryptographic software written by Eric Young.