

Contents

1 Preface	3
Translations	3
Windows Preparation	3
Shortcuts	5
2 Welcome	6
Choose a Language	7
Choose wallet mode	7
About the Simple/Bootstrap mode	9
3 Create a Wallet	11
Create new wallet	12
Add a password	12
Daemon settings	12
Starting a local node (full node)	16
Create new wallet from hardware	16
Create the wallet	16
Add a password	18
Daemon settings (Advanced mode)	18
Starting a local node (full node)	18
Restore wallet from keys or mnemonic seed	21
Restoring from seed	21
Restoring from keys	22
Open a wallet from file	23
4 Monero Account	24
5 Send Monero	26
Address Book	27
6 Receive Monero	28
Merchant view	29
7 Transactions History	31
8 Advanced Features	33
Solo mining	33
Prove - Check	33
Prove Transaction	33
Check Transaction	36
Shared RingDB	37
Sign - verify Message	39
Sign	40

Verify	40
Sign - verify File	40
Sign	40
Verify	40
9 Settings	42
Wallet	42
Interface	43
Local Node	44
Remote Node	44
Log	46
Info	47
Seed and keys	48
10 Binaries Verification	50
11 About remote nodes	51
Bootstrap nodes	51
12 Common issues and solutions	52

Chapter 1

Preface

This guide is open source and maintained by ErCiccione, of the Monero community. If you have suggestions or wish to contribute to the development of this guide, feel free to open Pull Requests or Issues on the GitHub repository where this document is maintained: github.com/monero-ecosystem/monero-GUI-guide.

For the online version of this guide, click [here](#)

Translations

This document will be localized into several languages. You will find all available translations in the [dedicated section on GitHub](#) after the next version (1.6) of this guide will be released.

Windows Preparation

If you are on Windows:

- Make sure that your antivirus does not block the program.

A miner is included in the Monero GUI software and, as a result, most anti virus software flags the Monero GUI files as malware and subsequently quarantines them. Fortunately, however, there's a fairly trivial work around you can use. That is:

- Create a new directory / folder.
- Open your AV software.
- Add an exception for the newly created directory / folder. Put differently, you have to whitelist the newly created directory / folder.
- Extract the .zip file (in case of Windows) or the .tar.bz2 file (in case of Linux or Mac OS X) to the whitelisted directory / folder.

Note: if you're using Windows and did not install custom anti virus software, Windows Defender is, most likely, acting as active anti virus software.

- The first time you start the wallet you must give permission to connect to the network via a pop-up. Check the appropriate boxes and click **Allow access**.

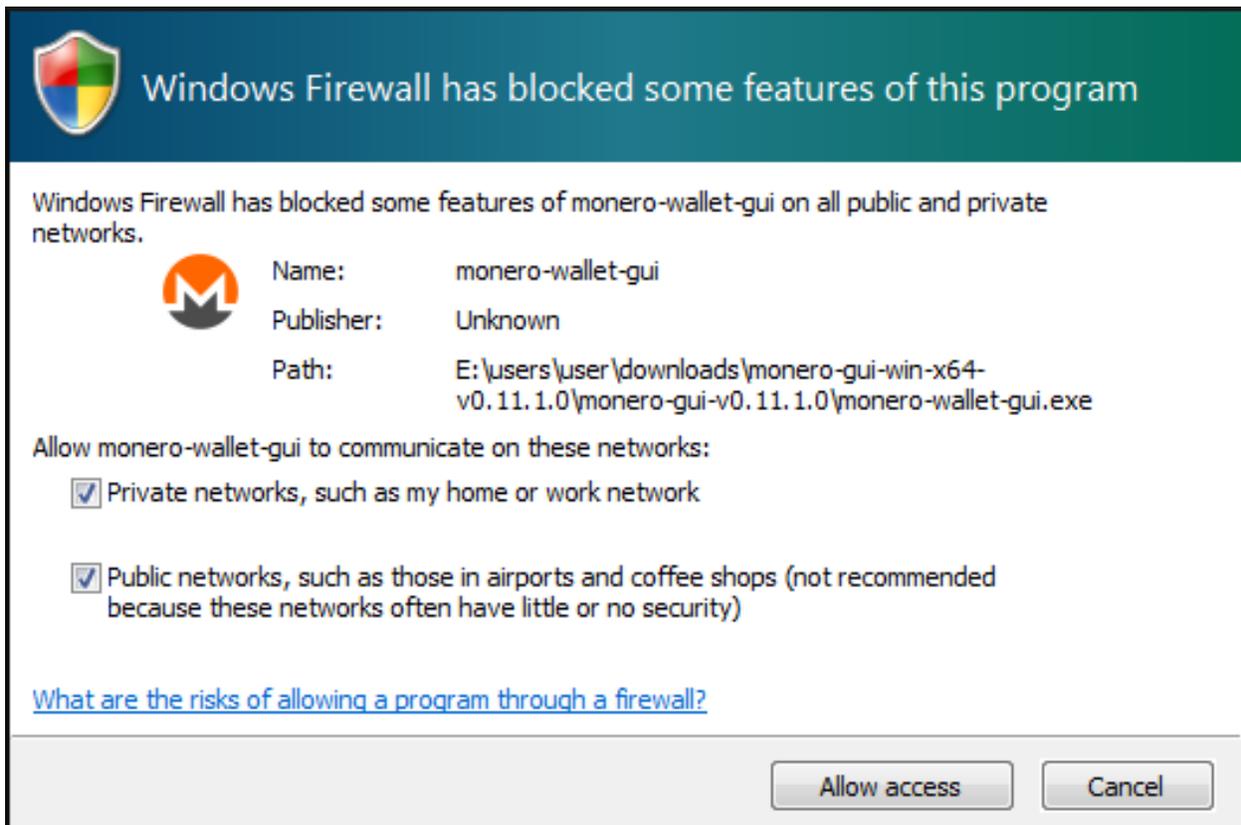


Figure 1.1: win firewall check

Shortcuts

Some shortcuts are available to make the user experience with the GUI easier:

On all operating systems:

Press **Ctrl** to view each page shortcut. Press **ctrl+"shortcut"** to go to that page

On Linux/Windows:

ctrl + tab -> go to the next page

ctrl + shift + tab -> go to the previous page

On macOS:

cmd + tab -> go to the next page

cmd + shift + tab -> go to the previous page

Chapter 2

Welcome

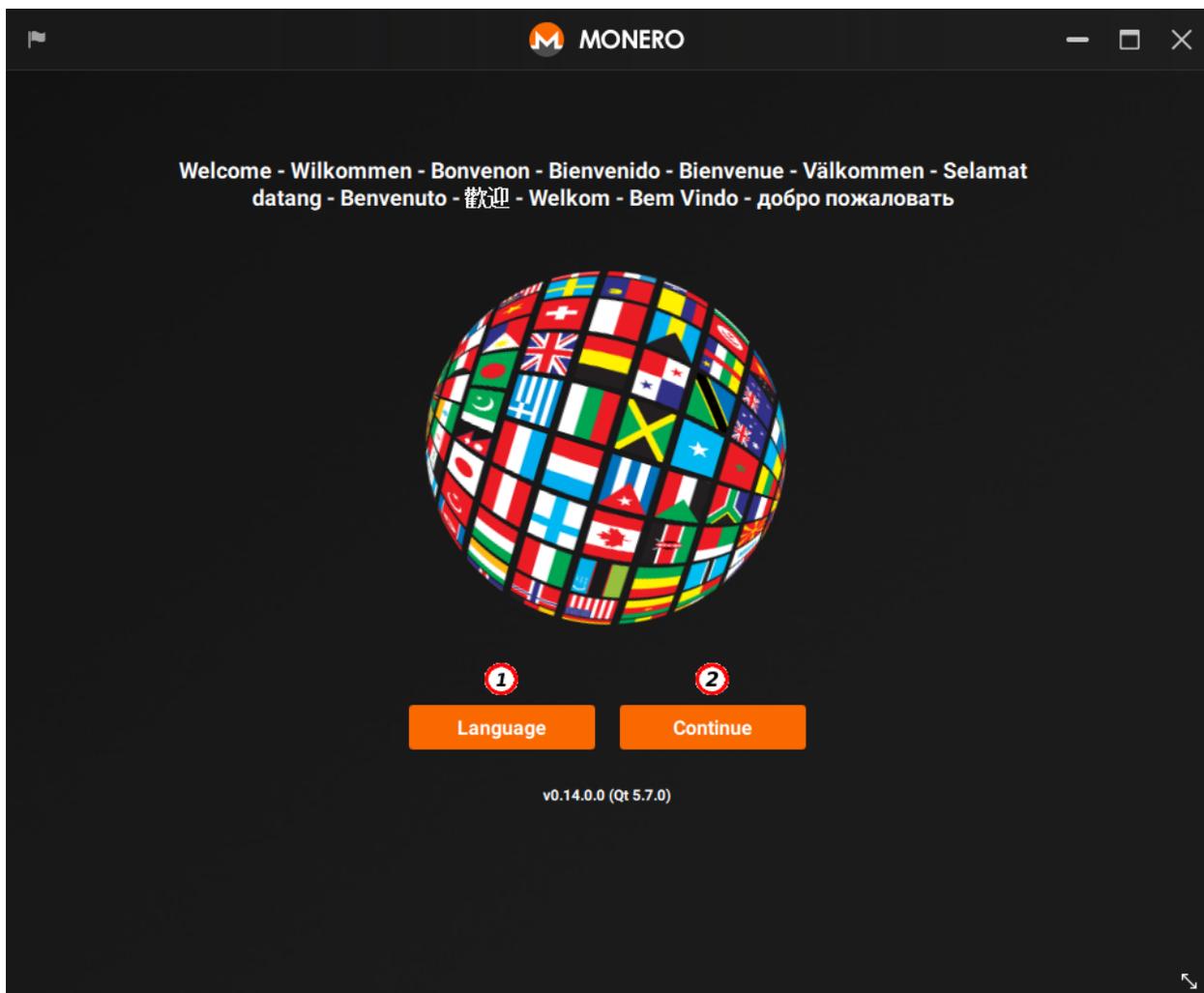


Figure 2.1: Welcome

Extract the package and click on `monero-wallet-gui`.

Welcome to Monero GUI Wallet.

- (1) **Language:** Click here to see a list of available languages.
- (2) **Continue:** If you are satisfied with the current language, continue to the next screen.

Choose a Language

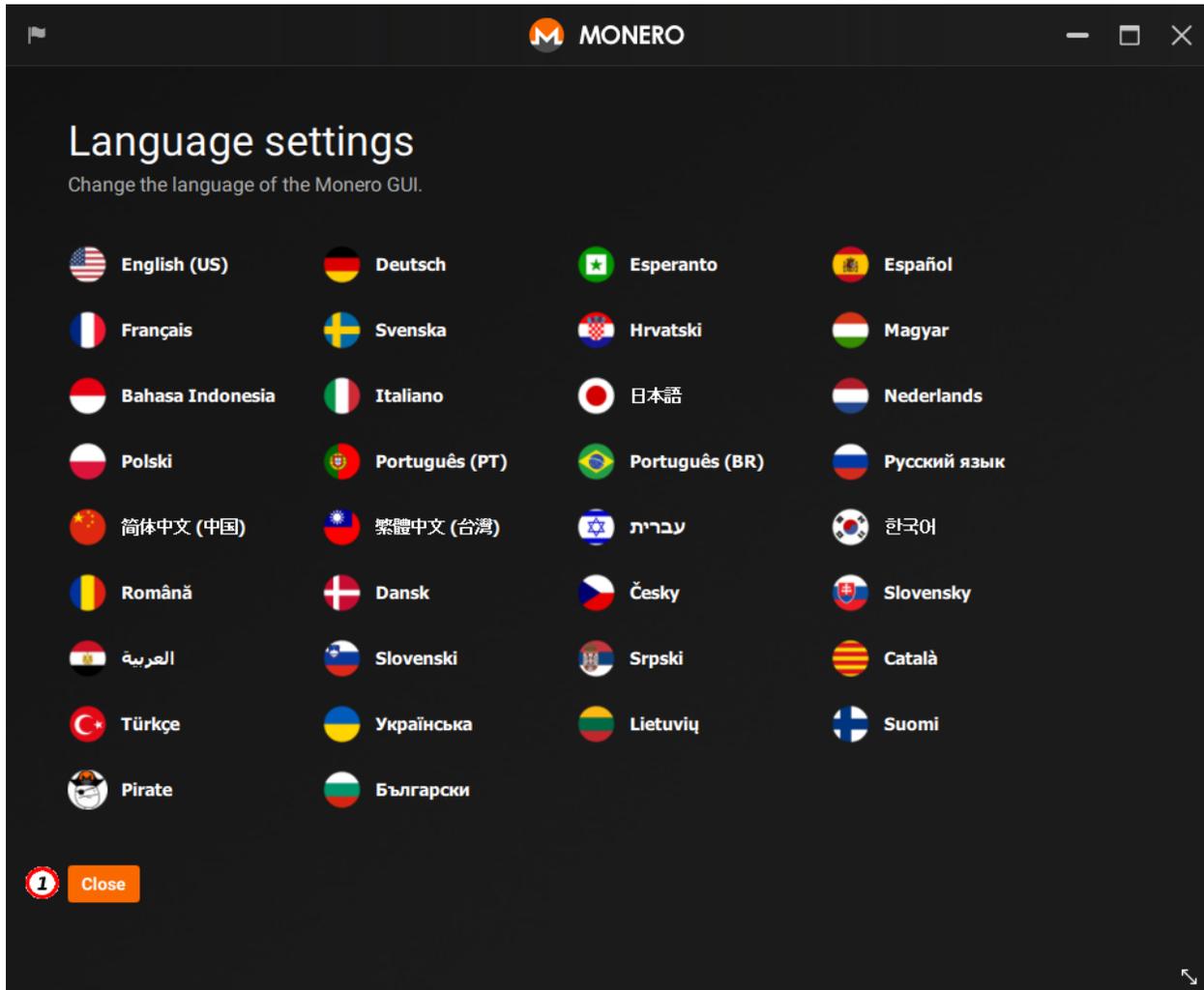


Figure 2.2: Language

You'll see a list of available languages, click on the one of your choice to change it and continue from the welcome screen. *Note:* You will be able to change the language later by clicking on the *flag* in the upper-left corner.

- (1) **Close:** Cancel and return to the previous screen.

Choose wallet mode

This page lets you choose the mode that best suits your accessibility and privacy needs. *Note:* Stagenet and Testnet only supports *Advanced mode*.

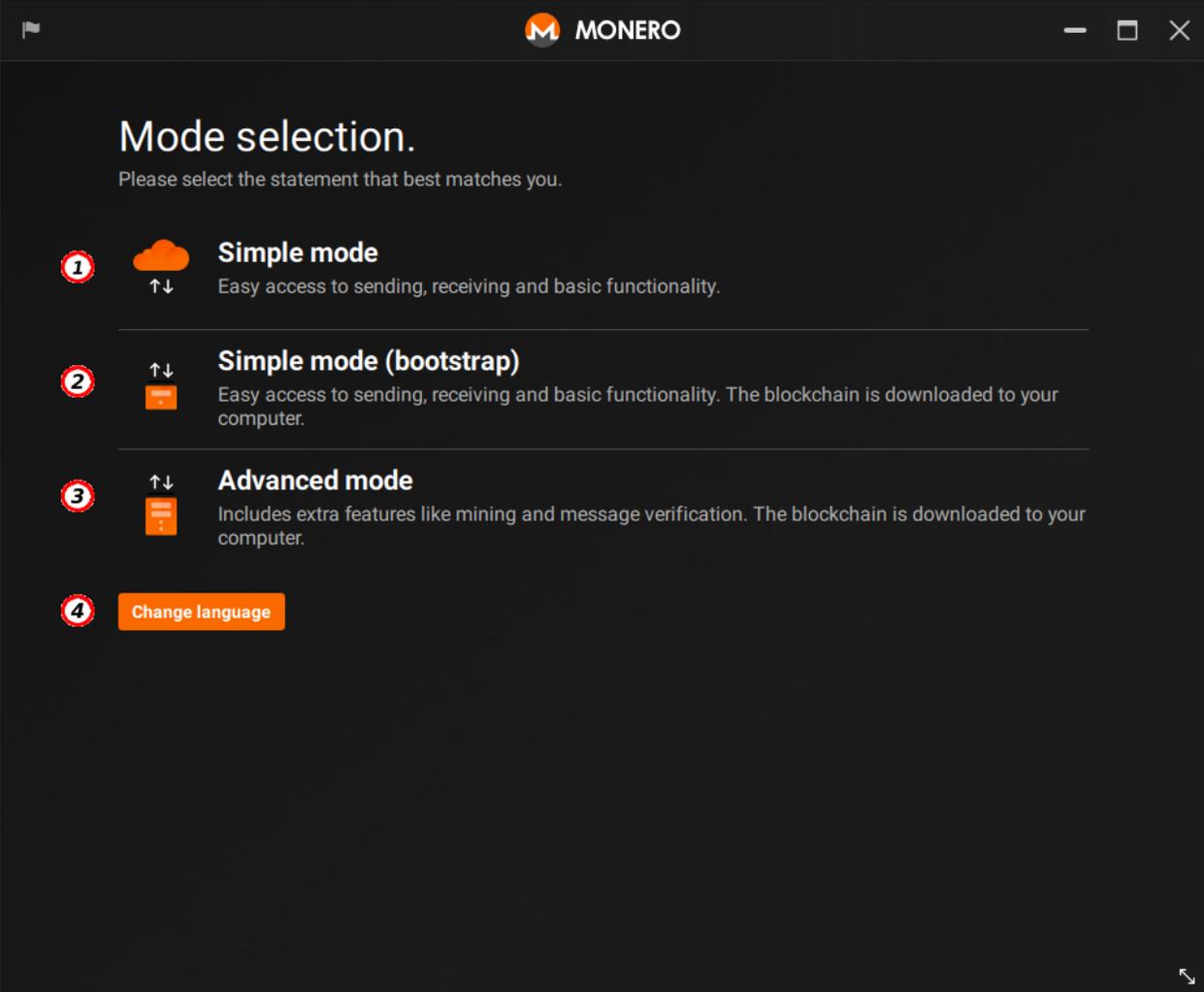


Figure 2.3: mode

- (1) **Simple mode:** In this mode your wallet will display a simple interface with easy access to basic functionality like sending, receiving, transactions history, address book, accounts, and merchant mode, as well as hardware wallet support. It will connect to a third-party server (a remote node), and it will not start a local node in your computer (the blockchain will not be downloaded). Choose this option if you don't have enough space to store the blockchain file (at least 90 GB), if you don't require maximum privacy and you accept the privacy risks of using a remote node to connect to the Monero network.
- (2) **Simple mode (bootstrap):** In this mode your wallet will also display a simple interface with easy access to basic functionality. Additionally it will start a local node in the background. This local node will start the download of the blockchain file (at least 90 GB), which can take many hours or even days. While the blockchain file is being downloaded, your local node will temporarily connect to a third-party server (a remote node known as bootstrap node), which will allow you to start using Monero immediately. After downloading and synchronizing the blockchain and the wallet, your local node will only temporarily connect to the bootstrap node when your blockchain is not updated. Be aware that using a remote node (including a bootstrap node) to connect to the Monero network may reduce your privacy.
- (3) **Advanced mode:** In this mode your wallet will display all advanced features available, like changing fees/transaction priority, solo mining, message verification, transaction proof/check, offline transaction signing, key images importing, custom remote node (including .onion remote nodes), custom bootstrap node, starting a local node without setting a bootstrap node (maximum privacy), changing blockchain location, and daemon log. In this mode you can choose if you want to connect to the Monero network via a remote node or a local node. If you decide to run a local node, the blockchain file will be downloaded. If you want to start using Monero immediately, you can also set a bootstrap node (remote node) in the Settings > Node page. To achieve maximum privacy in your connection to the Monero network, it's recommended to avoid using a remote/bootstrap node and instead use advanced mode to start a local node without setting a bootstrap node.
- (4) **Change language:** This will let you choose the language from the startup page (again).

About the Simple/Bootstrap mode

Upon choosing simple mode or bootstrap mode, an acknowledgement page will be shown, explaining the features, behavior, and privacy constraints of the mode.

Please carefully read those informations before accepting the implications of this mode and proceed to the next step.

You must understand the inherent risks of using a remote node for your privacy, and you should avoid using it with large amounts of money. If you connect to a *malicious* remote node, it could track your IP address, your "restore height" and associated block request data, and send you inaccurate information to learn more about transactions you make.

- (1) **Region:** Choose a region to pick up the remote nodes from. The closer the node is, the better the wallet performance will be.
- (2) **Acknowledgement:** Check This box if you have **fully understood** the privacy implications of a remote node and agree to **weaken your privacy** by using it.

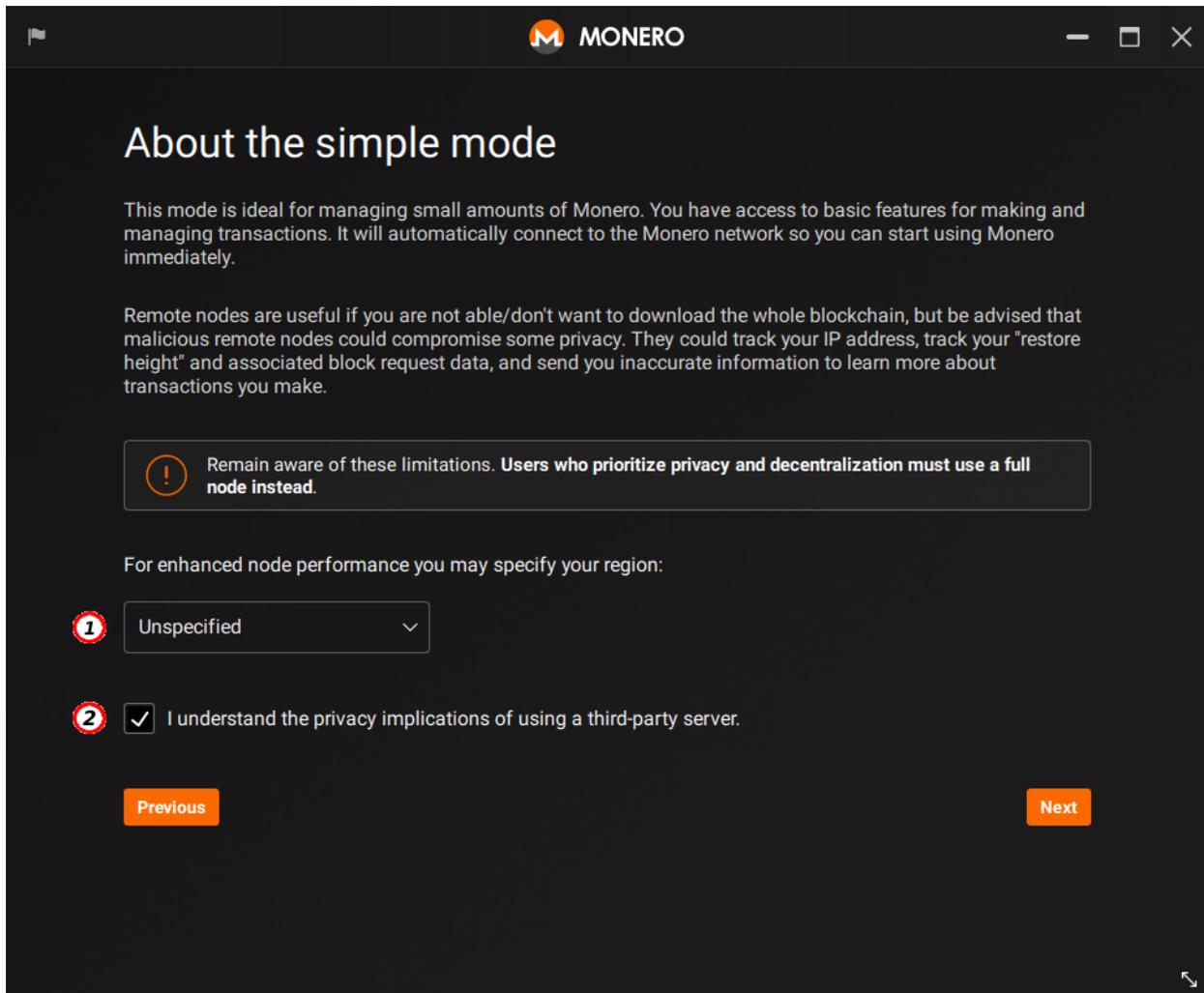


Figure 2.4: acknowledgement

Chapter 3

Create a Wallet

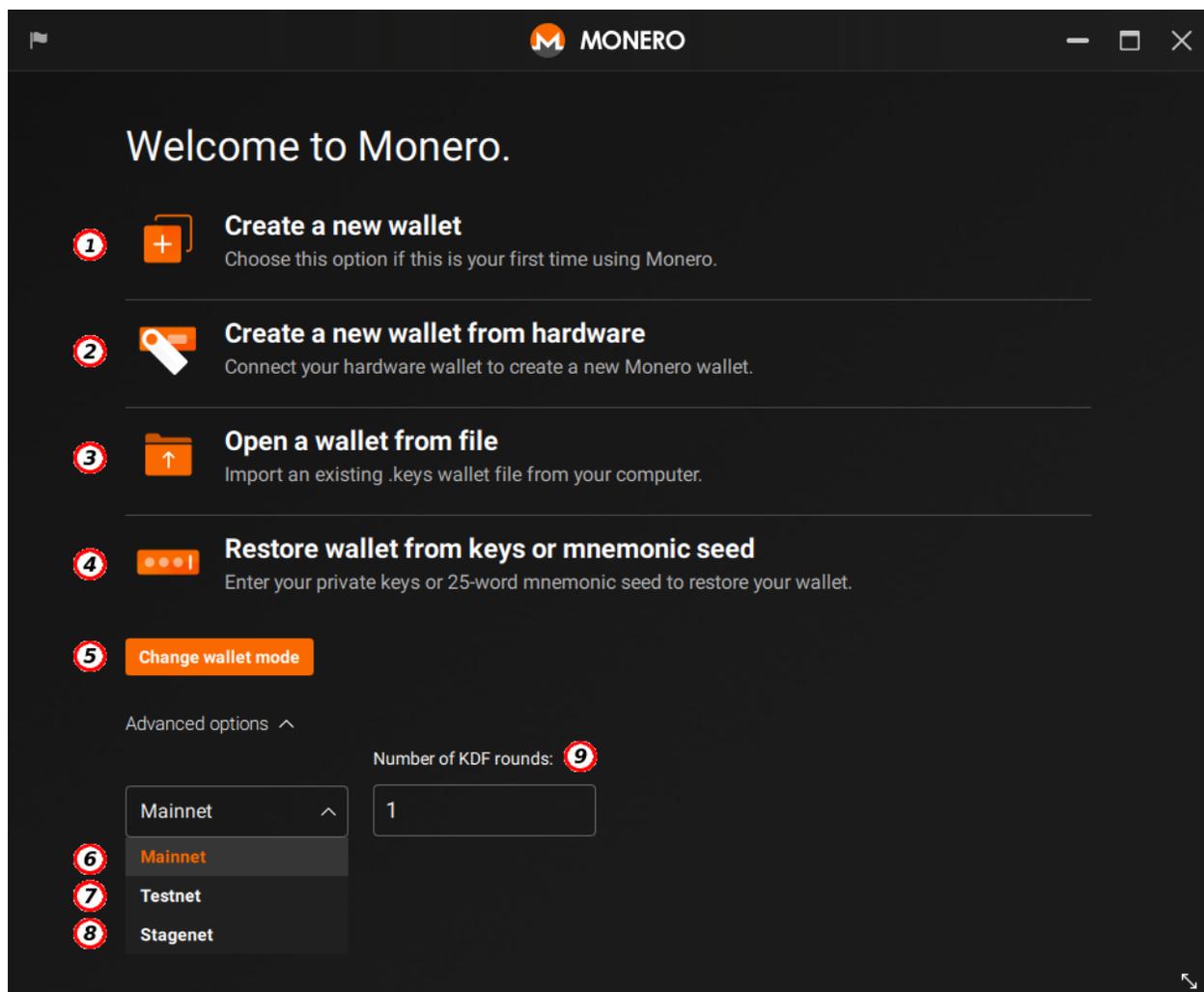


Figure 3.1: welcome

On this page you can create a new wallet, open an existing wallet, change wallet mode or change choose how to connect to the wallet and to the network:

- (1) **Create a new wallet:** Start the procedure to create a new wallet file. Choose this option if this is your first time using Monero.
- (2) **Create new wallet from hardware:** Create a new wallet file from a hardware wallet device (e.g., Ledger or Trezor).
- (3) **Open a wallet from file:** Choose this option to select a pre-existing wallet file from your computer (with the extension `.keys`).
- (4) **Restore wallet from keys or mnemonic seed:** Click here if you want to recover a pre-existing wallet using the mnemonic seed or the private keys, creating a new wallet file.
- (5) **Change wallet mode:** Click here to switch between simple, simple (bootstrap), and advanced modes.

Advanced options (requires wallet in advanced mode):

- (6) **Mainnet:** Select this from the dropdown list to use the main Monero network
- (7) **Testnet:** Select this option if you would like to use a development network instead of the main network. Testnet is designed to let developers test new features that are not available on Mainnet or Stagenet.
- (8) **Stagenet:** Select this option if you would like to use a network for staging instead of the main network. Stagenet mimics the features of Mainnet and is designed to let end users test Monero without the risk of losing funds.
- (9) **Number of KDF rounds:** Adjust the number of Key Derivation Function rounds in order to enhance the overall security of the generated keys. Any random large number will increase the security.

Create new wallet

Here you can create a new wallet file:

- (1) **Wallet name:** Give a name for your wallet (in this example `testname` is used).
- (2) **Mnemonic seed:** Write down your **mnemonic seed** and keep it safe. Your seed is the master key of your wallet, you can use it to recover your funds.
- (3) **Wallet location:** Select the destination folder of the wallet.

Add a password

Add a strong password to protect your wallet. If you lose your password, then only your mnemonic seed can recover your wallet.

Daemon settings

This page will open if your wallet is in advanced mode. Here you can choose if you are going to start a local node or connect to a remote node:

- (1) **Start node in background:** Check this box to start a local node (full node) in the background and download the blockchain.

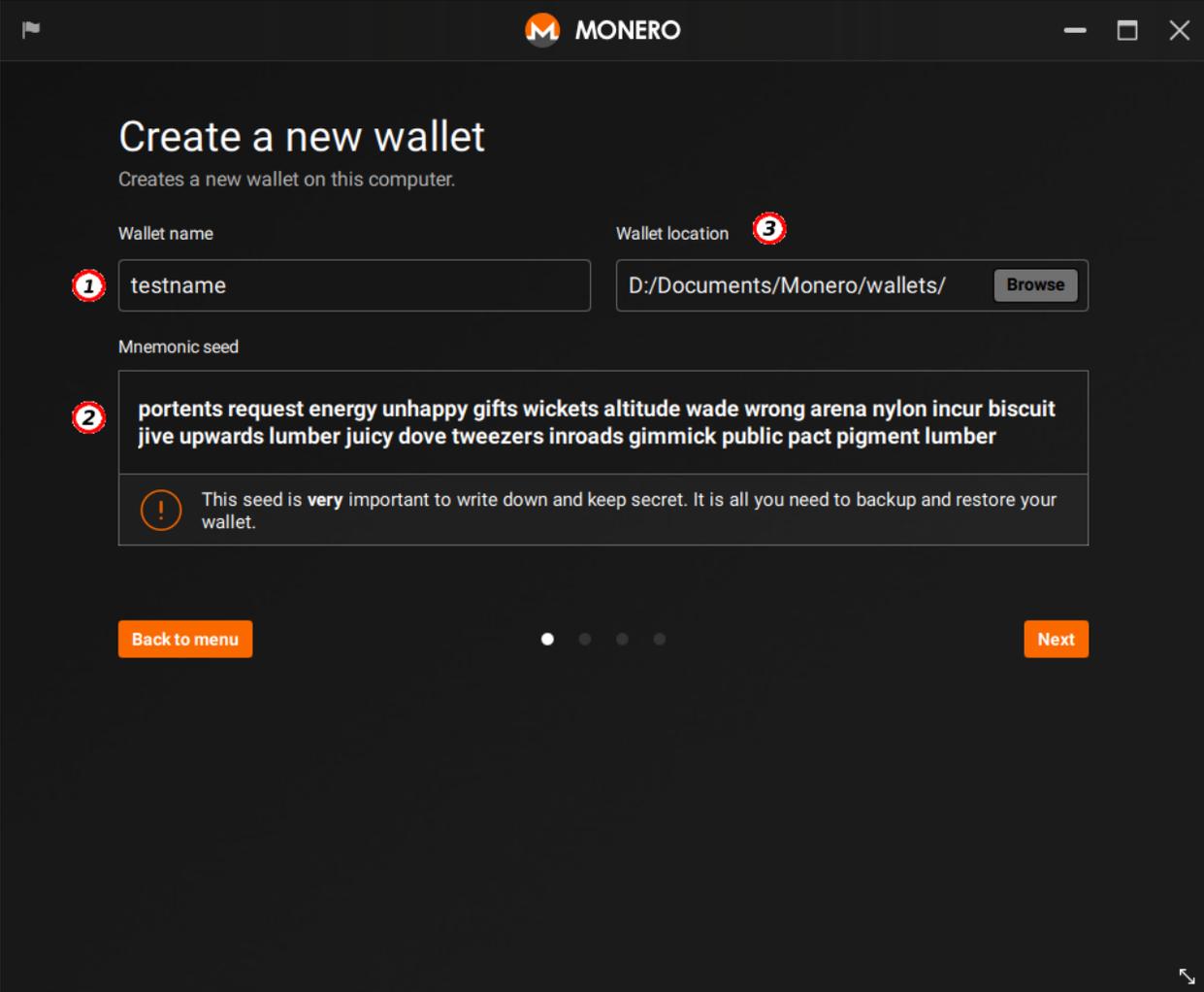


Figure 3.2: new

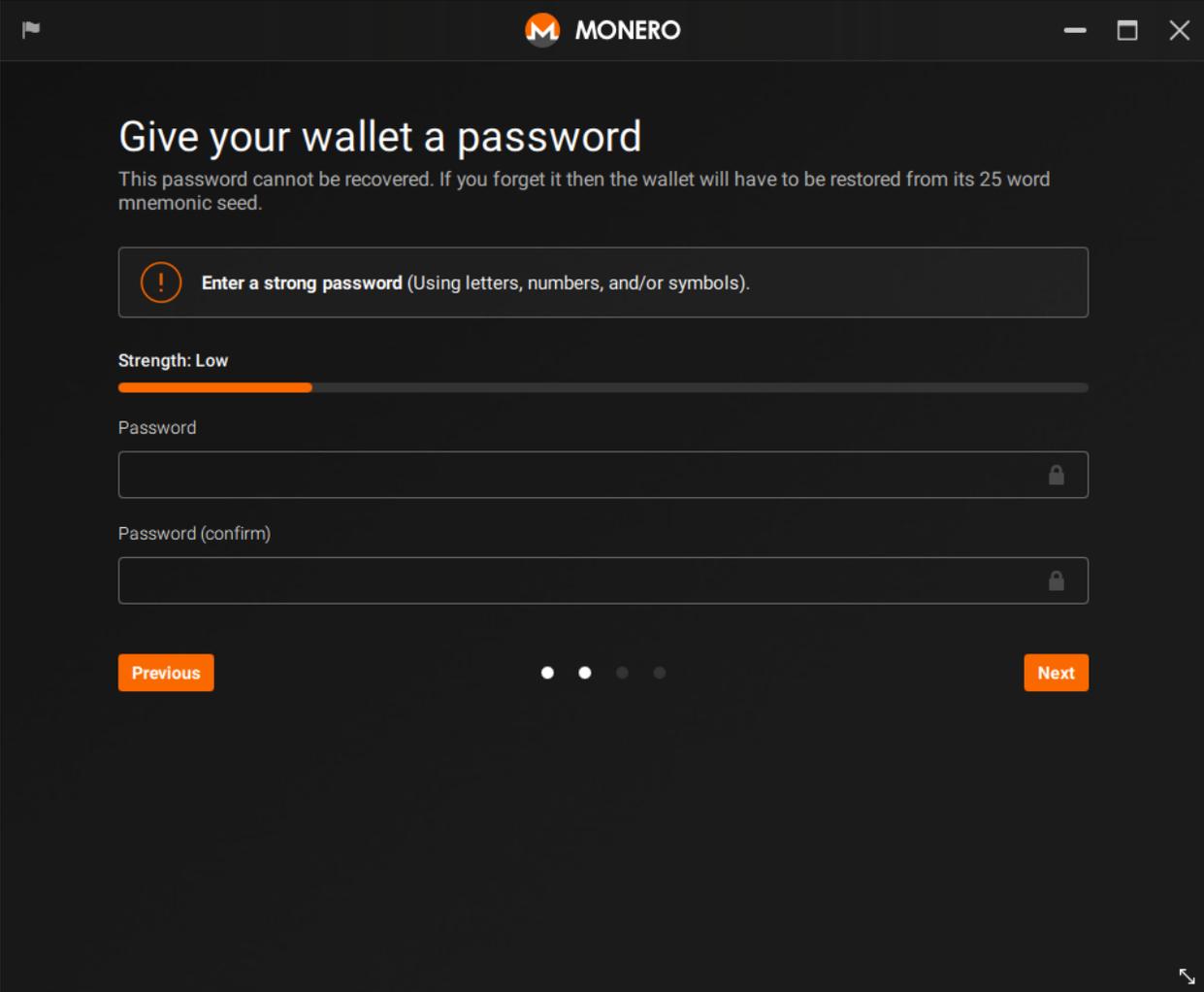


Figure 3.3: add password

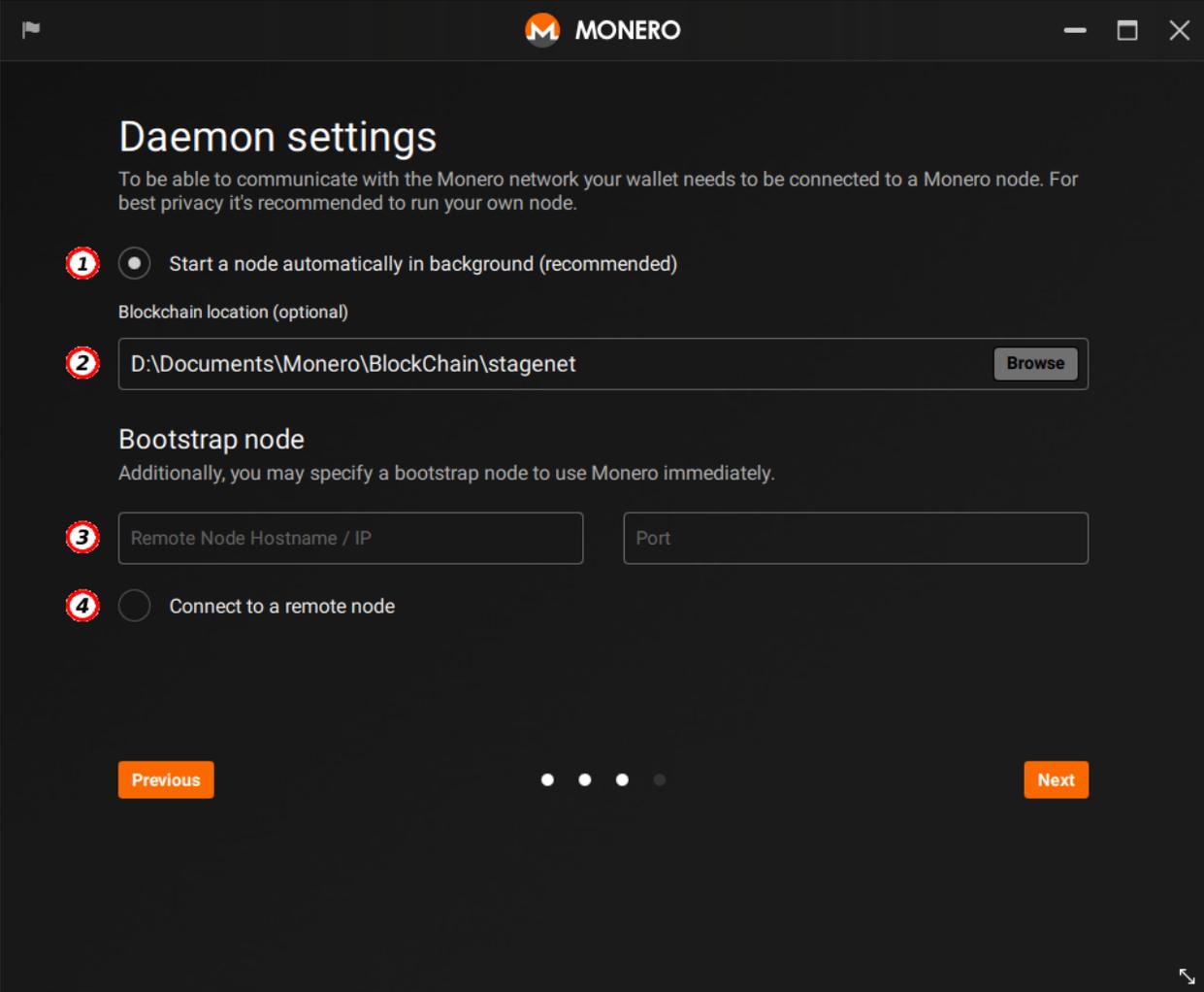


Figure 3.4: daemon settings

- (2) **Blockchain location (optional):** To store the blockchain somewhere other than default, enter that location here.
- (3) **Bootstrap node:** Here you can enter the host/IP address and port of a bootstrap node (remote node). Your local node will temporarily connect to this third-party remote node while the blockchain is being downloaded, allowing you to use your wallet immediately. For a list of available remote nodes and info about them, check the [About remote nodes](#) section of this guide.
- (4) **Connect to a remote node:** Check this box if you just want to use a remote node without downloading the blockchain. Enter the host/IP address and port of the remote node after checking the box.

Starting a local node (full node)

If you have selected to start a local node, every time you open your wallet you will see this pop up window:

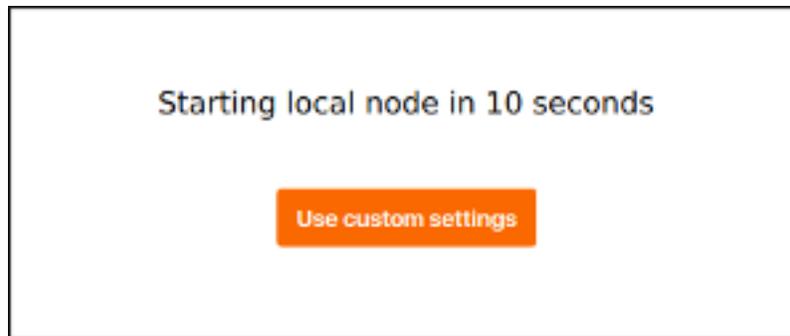


Figure 3.5: sync

It has a countdown that allows you to change some settings before your wallet connects to the local node. If you don't want to change any setting, just let the countdown finish and your local node will be started automatically.

Create new wallet from hardware

On this page you can create a new wallet file from your hardware wallet device. This wallet file will be stored in your computer and every time you open it you will be asked to connect and unlock your hardware wallet. Monero GUI currently supports the hardware wallet models Ledger Nano S/X and Trezor Model T.

Create the wallet

- (1) **Wallet name:** Give a name for your wallet (in this example `ledger-test` is used).
- (2) **Wallet location:** Select the destination folder of the wallet.
- (3) **Create a new wallet from device:** Select this option if this is the first time you use a hardware wallet.
- (4) **Restore a wallet from device:** Choose this option if you already had a hardware wallet and want to restore it.

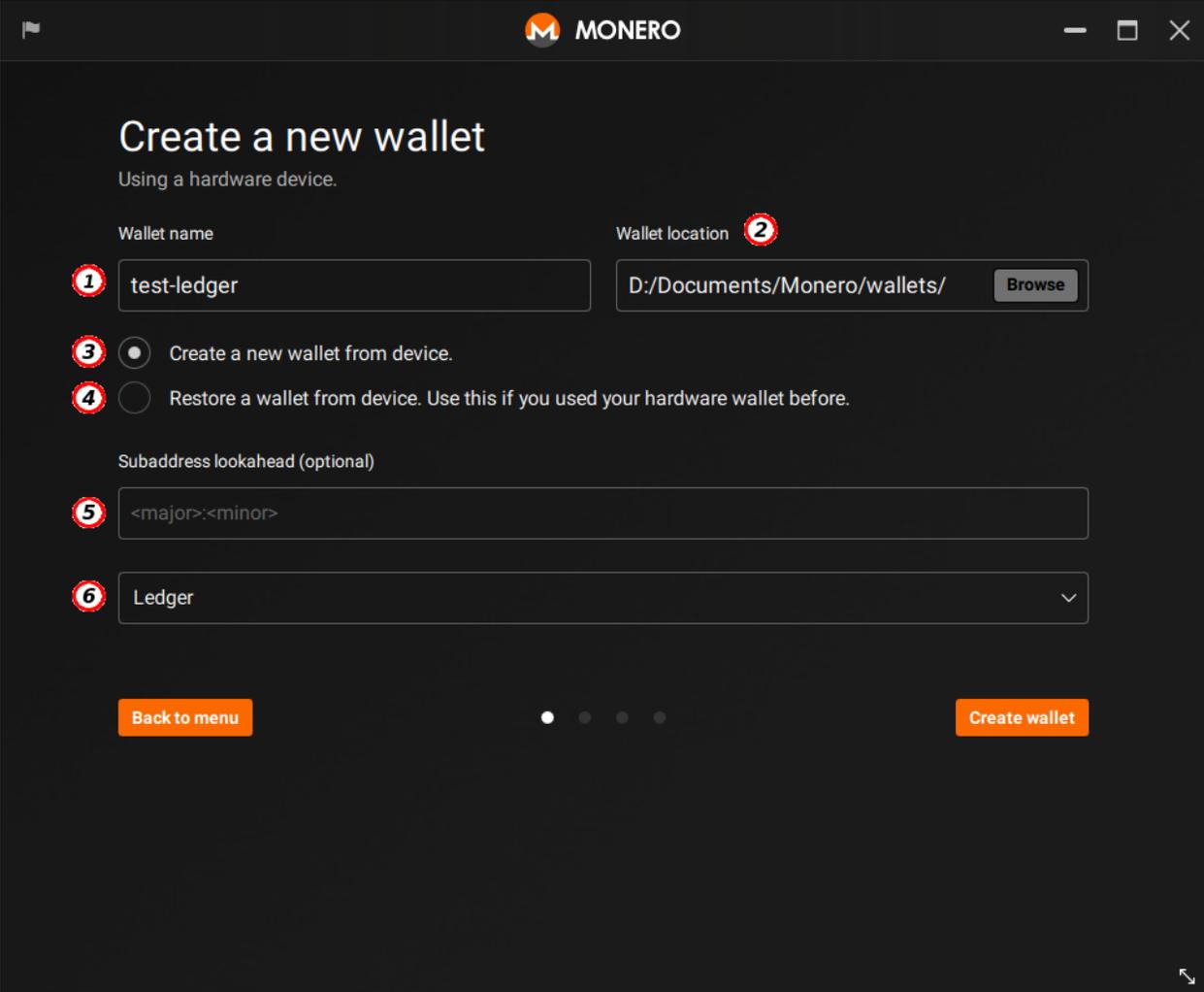


Figure 3.6: 2.4.1

Creation Date or Restore height (optional): *Restore only:* If you know the date or block height that your wallet was created at, you can specify it here so the wallet doesn't have to scan the entire blockchain looking for your funds. For example, if your first transaction was included on 2017-07-08 in block 1350000, you should put the previous day date (e.g. 2017-07-06) or a slightly lower height (e.g. 1330000) so the wallet will start scanning from there, saving you some time. More information about restore height on [StackExchange](#).

- (5) **Subaddress lookahead (optional):** pregenerate a number of accounts with a number of subaddresses each.
- (6) **Device name:** Select the hardware wallet you want to use. Monero GUI currently supports the hardware wallet models Ledger Nano S/X and Trezor Model T.

Step by step guides on how to generate a Monero wallet from a hardware wallet for all operating systems ([StackExchange](#)):

[How do I generate a Ledger Monero wallet with the GUI \(monero-wallet-gui\)?](#)

[How do I generate a Trezor Monero wallet with the GUI \(monero-wallet-gui\)?](#)

Add a password

Add a strong password to protect your wallet. If you forget the password of a wallet created from your hardware wallet, you will be able to create another wallet from your hardware device.

Daemon settings (Advanced mode)

Here you can choose if you are going to run a full node or use a remote one:

- (1) **Start node in background:** Check this box to run a full node and begin blockchain sync.
- (2) **Blockchain location (optional):** To store the blockchain somewhere other than default, enter that location here.
- (3) **Bootstrap node:** To use a bootstrap node enter the host and port. A bootstrap node allows you to use your wallet while you are downloading the blockchain by connecting to a remote node. For a list of available remote nodes and info about them, check the [About remote nodes](#) section of this guide.
- (4) **Connect to a remote node:** Check this box if you want to use only a remote node without downloading the blockchain. You will need to put the host and port of the remote node after checking the box.

Starting a local node (full node)

If you have selected to start a local node, every time you open your wallet you will see this pop up window:

It has a countdown that allows you to change some settings before your wallet connects to the local node. If you don't want to change any setting, just let the countdown finish and your local node will be started automatically.

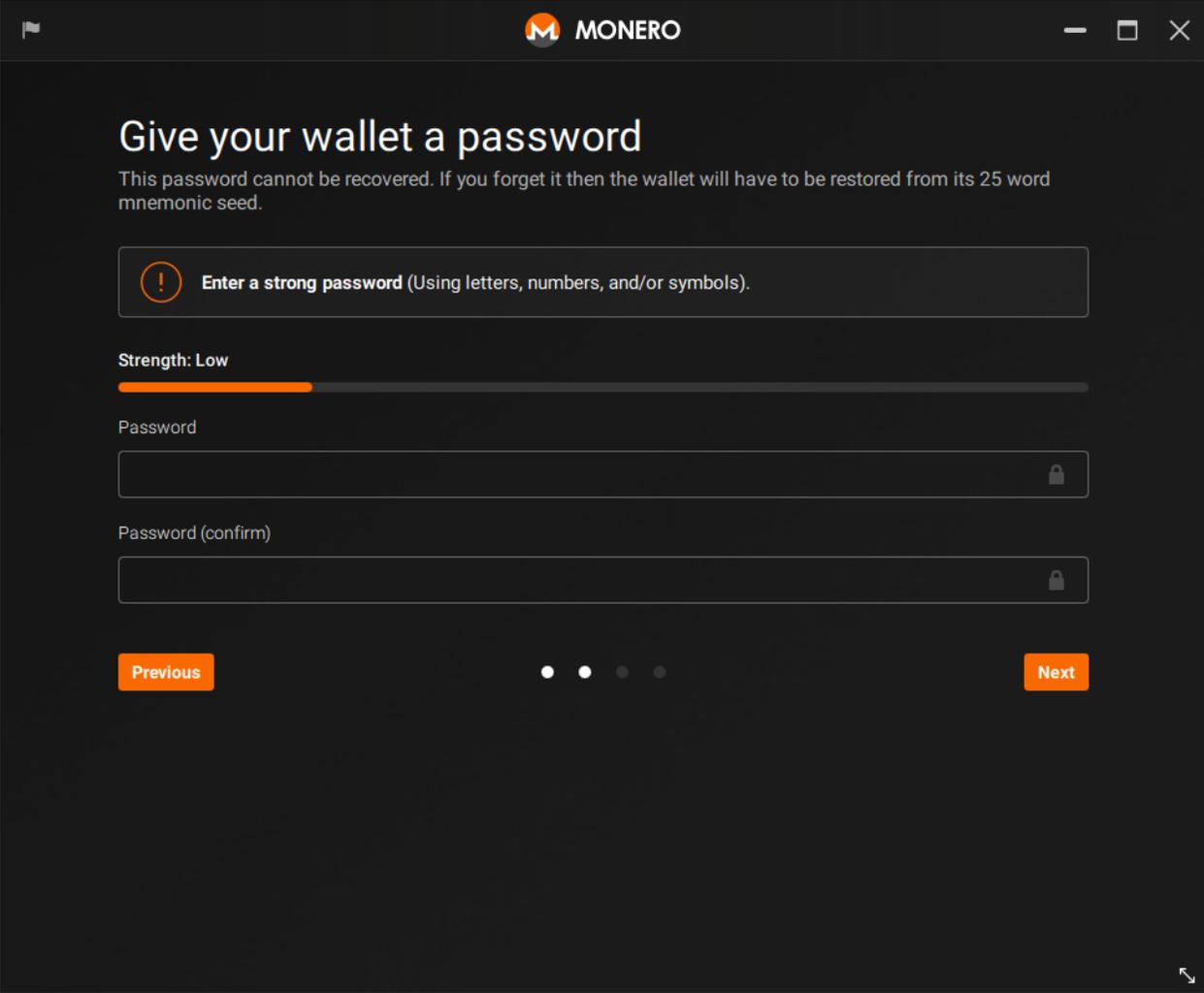


Figure 3.7: add password

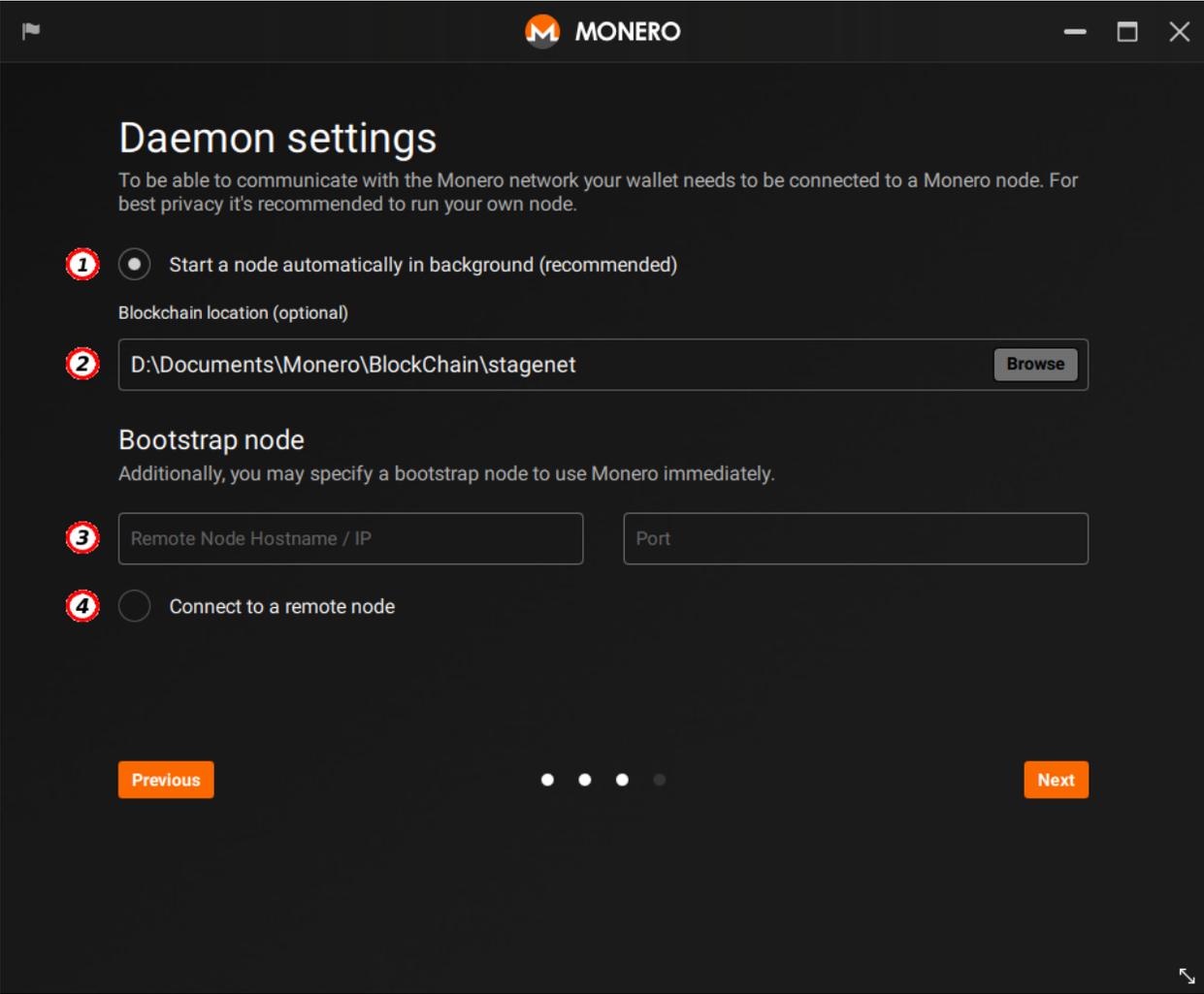


Figure 3.8: daemon settings

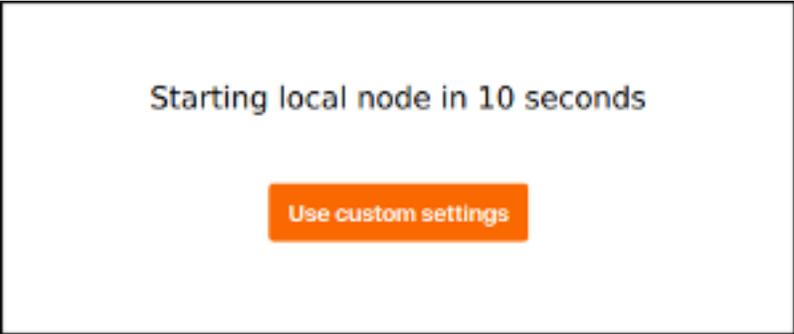


Figure 3.9: sync

Restore wallet from keys or mnemonic seed

Restoring from seed

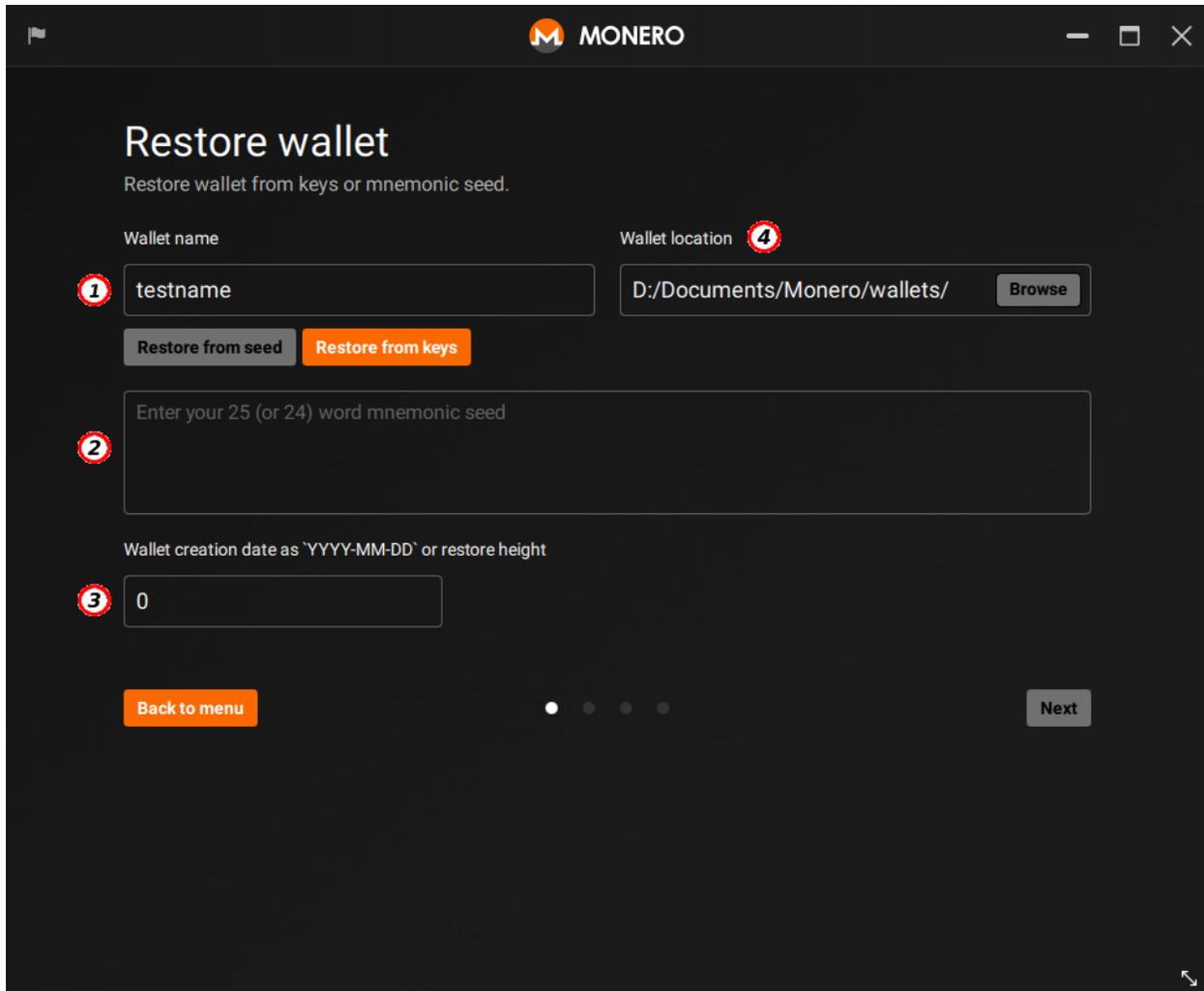


Figure 3.10: restore from seed

Restoring from your mnemonic seed is the easiest way to recover your wallet. You need to put the following information:

- (1) **Wallet name:** Give a name for your wallet (in this example `testname` is used).
- (2) **Mnemonic seed:** Paste your seed made of 25 (or 24) words.
- (3) **Creation Date or Restore height (optional):** If you know the date or block height that your wallet was created at, you can specify it here so the wallet doesn't have to scan the entire blockchain looking for your funds. For example, if your first transaction was included on 2017-07-08 in block 1350000, you should put the previous day date (e.g. 2017-07-06) or a slightly lower height (e.g. 1330000) so the wallet will start scanning from there, saving you some time. More information about restore height on [StackExchange](#).

(4) **Wallet location:** Select the destination folder of the wallet.

A detailed guide is available on getmonero.org: [‘How to restore your account’](#)

Restoring from keys

The screenshot shows the 'Restore wallet' interface in the Monero desktop application. The window title is 'MONERO'. The main heading is 'Restore wallet' with the subtitle 'Restore wallet from keys or mnemonic seed.' The form has five numbered steps: 1. 'Wallet name' field containing 'testname'. 2. 'Account address (public)' field. 3. 'View key (private)' field. 4. 'Spend key (private)' field. 5. 'Wallet creation date as `YYYY-MM-DD` or restore height' field containing '0'. There are also 'Restore from seed' and 'Restore from keys' buttons, a 'Wallet location' field with 'D:/Documents/Monero/wallets/' and a 'Browse' button, and 'Back to menu' and 'Next' buttons at the bottom.

Figure 3.11: restore from key

Restoring from keys is quite easy and can be extremely useful, especially if you are moving your wallet from an online service like MyMonero. You need to put the following information:

- (1) **Wallet name:** Give a name for your wallet (in this example `testname` is used).
- (2) **Account address:** The address of the wallet you are recovering.
- (3) **View key:** Your private view key (needed to be able to check your funds).
- (4) **Spend key:** Your private spend key (needed to spend your funds).
- (5) **Creation Date or Restore height (optional):** If you know the date or block height that your wallet was created at, you can specify it here so the wallet doesn't have to scan the entire blockchain

looking for your funds. For example, if your first transaction was included on 2017-07-08 in block 1350000, you should put the previous day date (e.g. 2017-07-06) or a slightly lower height (e.g. 1330000) so the wallet will start scanning from there, saving you some time. More information about restore height on [StackExchange](#).

(6) **Wallet location:** Select the destination folder of the wallet.

When everything is ready click the right arrow and then the **Use Monero** button. A detailed guide is available on getmonero.org: '[Restoring wallet from keys](#)'

Open a wallet from file

After clicking this option a window will pop up. Navigate to your file with the extension `.keys`, select it and click the right arrow.

Chapter 4

Monero Account

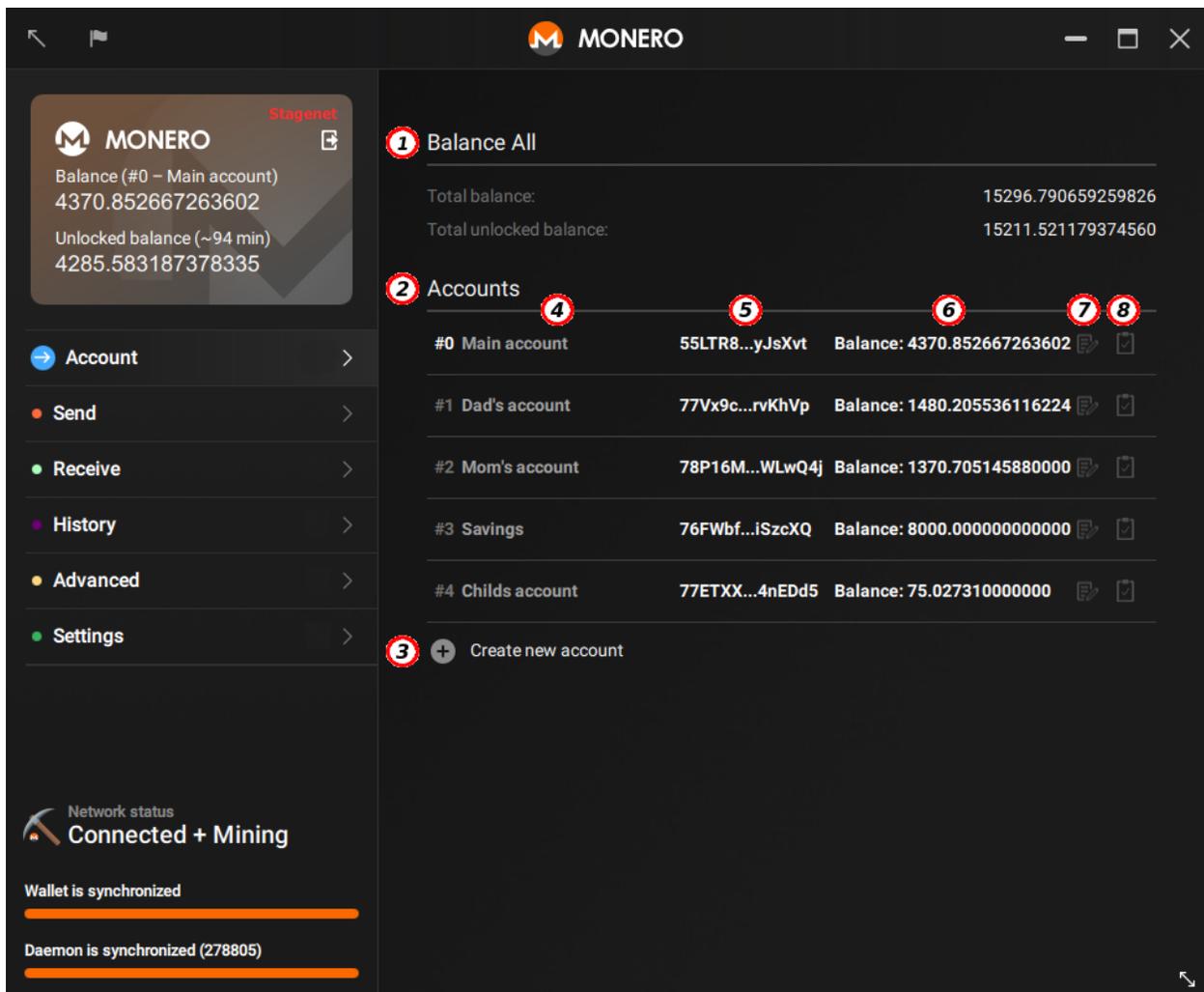


Figure 4.1: account

The Account tab provides tool to handle your different Monero accounts within your wallet. Each wallet can have multiple accounts, and each account has its own balance. Also, each account can have multiple

subaddresses associated with it. For convenience sake, the account address is considered the first address (or subaddress) generated for an account. Since accounts and subaddresses are deterministically derived from the mnemonic seed, you only need to know the seed in order to restore the account/subaddress structure when restoring a wallet (although any labels you assign to the accounts/subaddresses will need to be added again, since they are stored in your wallet, and not in your seed).

- (1) **Balance All:** Let you see the sum of all your accounts balances, unlocked or not.
- (2) **Accounts:** This is the list of all accounts already generated on your wallet and their corresponding account addresses.
- (3) **Create new account:** Click on this button to create a new account and set its *optional* label.

You can change the current account by simply clicking on it on the list.

Each of your accounts are filled with their own informations:

- (4) **Label:** A name or description of the account for convenience.
- (5) **Account Address:** This is the public address you should give to the sender when you want to receive money in this account. In account #0, it starts with 4 and corresponds to the wallet's primary address. In the remaining accounts it corresponds to the account's first subaddress and therefore starts with 8.
- (6) **Balance:** The account total balance. It cumulates the main address and all subaddresses for this account. *Note: Subaddresses are managed in the [Receive](#) screen*
- (7) **Edit Label:** Click on this button to edit this account label.
- (8) **Copy Address to Clipboard:** Click on this button to copy this account address to the clipboard.

Chapter 5

Send Monero

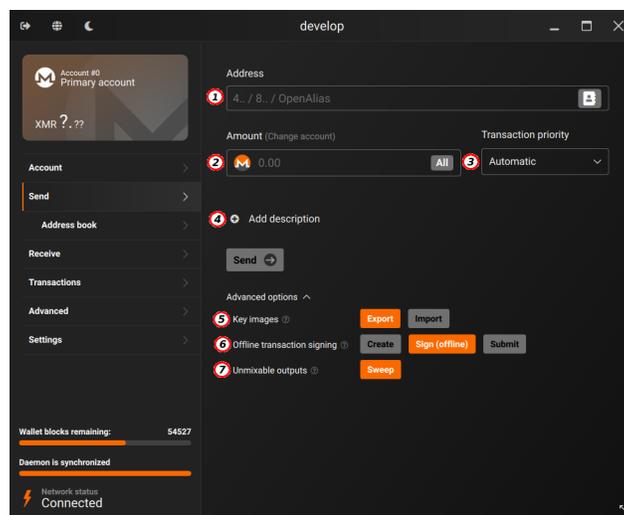


Figure 5.1: send

The **Send** tab provides tools for creating outgoing transactions.

- (1) **Address:** This is where you enter the destination address of the transaction. You can either enter a Monero address (a primary address starting with 4... or a subaddress starting with 8...) or an OpenAlias address (see [this page](#) for more information about OpenAlias). Best practice is to copy and paste the address to prevent errors. Since there are malwares that may replace the address of your clipboard with an address associated with the hackers, you should always verify the destination address after pasting it. Click on the *Address book* button on the right side to import an address from the address book of your wallet.
- (2) **Amount:** This is how much Monero (XMR) you want to send. Click on the *All* button on the right side to create a transaction spending all the spendable (unlocked) amount available in your wallet. If you have enabled the option to display the balance in other currencies in *Settings > Interface*, your wallet will also display the amount converted in fiat currency.
- (3) **Transaction priority:** This is the priority level your transaction will receive in the pool of transactions waiting to be confirmed. The higher the transaction priority, the faster your transaction will be confirmed (included in a block), but the higher you will pay in transaction fee.
Currently staying with the default or the slow option is likely to get you into the next block.

- (4) **Description (optional):** Here you can add some information regarding your transaction for future reference. This function is for your record keeping only. The description text will be stored in your wallet file and only you will be able to see it. It will not be sent to the receiver, nor stored on the blockchain.

Advanced options (requires wallet in advanced mode)

- (5) **Key images:** This section allows you to export the key images from your cold wallet into a file, as well as to import this key images file into your view-only wallet. When your view-only wallet has the key images for all outputs, it can display the true balance. In order to import the key images into your view-only wallet, you must use a local node or be connected to a trusted remote node (you can set a trusted remote node in “Settings” > “Node” page). Click on the question mark icon to see step-by-step instructions.
- (6) **Offline transaction signing:** This feature refers to the practice of keeping the private keys in a cold wallet, which is a wallet installed in an offline (“cold”) device. Since the offline wallet never gets connected to the internet, its stored private keys remain secure. The cold wallet is only used to sign individual transactions, which are later sent to the Monero network using a view-only wallet. Click on the question mark icon to see step-by-step instructions.
- (7) **Sweep unmixable:** This allows you to get rid of outputs in your wallet which have strange amounts like 0.000006839355 and can’t be spent without combining them with another output.
Most users will never need to use this feature.

Address Book

The **Address book** tab lets you save the name and Monero address of contacts that you frequently transact with. This is a convenient place to copy addresses from when creating transactions.

Chapter 6

Receive Monero

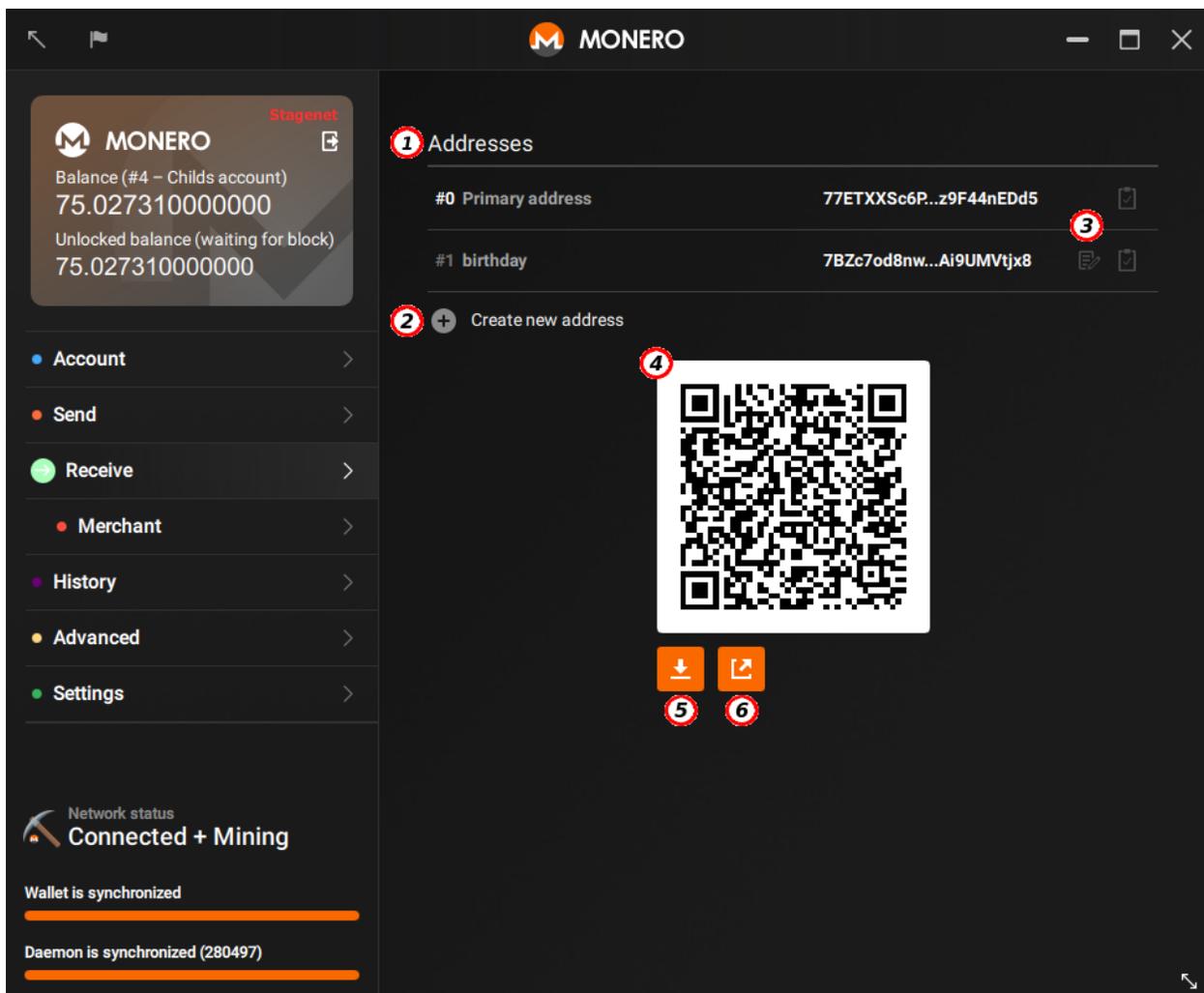


Figure 6.1: receive

The `Receive` tab provides tools for generating subaddresses.

- (1) **Addresses:** This list includes your wallet's primary address (starting with 4...) and all its subaddresses

(starting with 8...).

- (2) **Create new address:** This button allows you to create as many new subaddresses as you wish. Since Monero is private, you don't need to create a new subaddress every time you receive a transaction. However, to avoid off-chain linking, it's recommended that you create a new subaddress for every person, company or exchange you will receive Monero from. *Learn more about [subaddresses](#).*
- (3) **Set Label:** Click on this button to set this subaddress label. You can use this label to store the person, company or exchange you gave this subaddress.
- (4) **QR code:** This is a QR code that has your selected address embedded into it. It can be used as a way to give others your Monero address by scanning the code.
- (5) **Save:** Click here to save the QR code as a **png** image.
- (6) **Copy:** Click here to copy the QR code corresponding URL formatted as `*monero:`

•

Merchant view

The **Merchant** view is intending to offer professionals a nice app on their points of sell. It let you craft payment requests, and monitor incoming transactions

- (1) **QR code:** This is a QR code that has your selected address, and optionally the amount, embedded into it. It can be used as a way to give others your Monero address by scanning the code.
- (2) **Amount:** This is for creating a payment request, enter the amount of Monero you would like to receive.
- (3) **Payment URL:** This is the URL corresponding to the address and amount you have chosen. It could be copied by clicking on it and sent to a customer.
- (4) **Tracking:** If you tick this box, you'll see a list of incoming transactions.
- (5) **Incoming transaction:** Those are the transaction currently seen on the blockchain and the transaction pool, with associated confirmations.

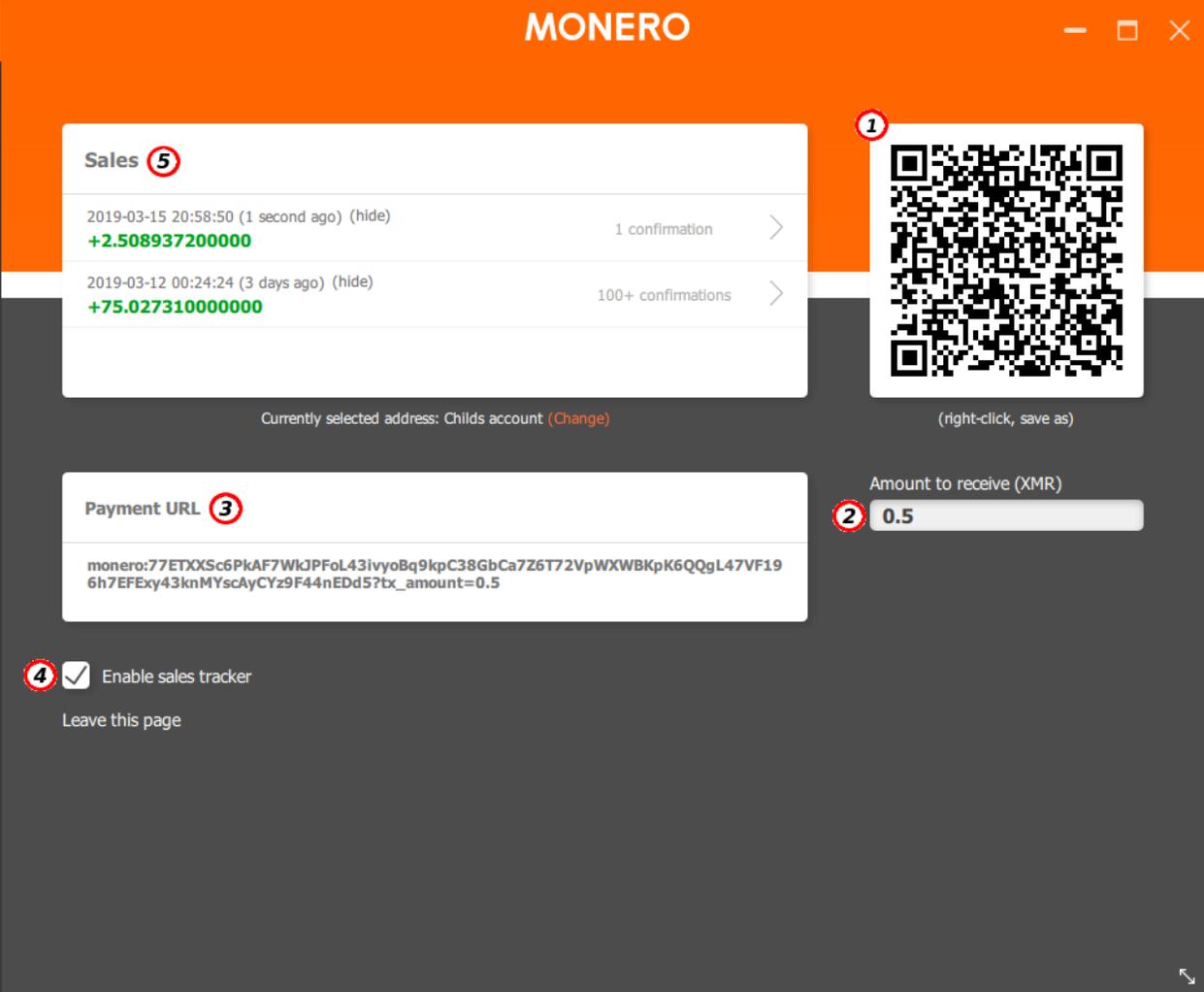


Figure 6.2: merchant

Chapter 7

Transactions History

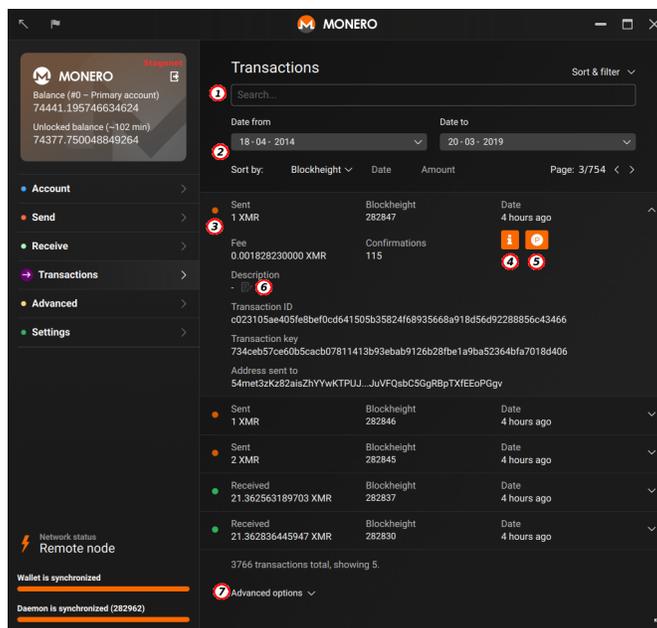


Figure 7.1: transactions

The **Transactions** tab let you walk through the transactions sent and received on the currently selected account of your Monero wallet.

- (1) **Search:** Quickly filter out transactions from the list searching by transaction ID, destination address, description, amount or blockheight (number of the block in which the transaction was mined).
- (2) **Sort & Filter:** Here you can filter out transactions between specific dates, or sort them by blockheight, date or amount.
- (3) **Transaction list:** This area displays the sent and received transactions of the currently selected account. Each transaction is displayed with the following information: **Amount**, **Blockheight**, **Date**, **Fee** (for outgoing transactions), **Confirmations**, **Description** (optional), **Transaction ID**, **Transaction key**, and **Destination address**. The transaction ID, the transaction key and the destination address can be used to prove that a payment was done in case of a dispute. The transaction keys and the descriptions (notes) are stored in the cache of your local wallet, therefore they cannot be recovered

from the blockchain.

- (4) **Details:** Clicking here will show you the following transaction details, when available: **Transaction ID, Date, Amount, Address, Payment ID, Integrated address, Transaction key, Description (note), Destinations and Rings.**
- (5) **Prove payment (P):** Clicking here will generate a payment proof, which you can use, in case of a dispute, to prove that you made a payment. More details on proving payments can be found [here](#)
- (6) **Set description:** Click on this icon to set a new description or edit an existing one.
- (7) **Advanced options:** Lets you choose in which format the date should be displayed.

Chapter 8

Advanced Features

Solo mining

The Mining tab provides a one click CPU miner that is embedded into the GUI.

- (1) **CPU threads:** Number of CPU threads to use for mining.
- (2) **threads auto-configuration:** Set automatically the number of threads either to the recommended number, or to the max number.
- (3) **Background mining:** Check this box to enable experimental background mining. This should allow you to use your computer normally while mining.
- (4) **Start mining:** Start the miner.
- (5) **Stop mining:** Stop the miner.
- (6) **Status:** Indicates the mining state and hashrate.

Prove - Check

The Prove/check tab provides tools for proving a payment or validating proof of a payment. This is necessary with Monero because these details are not available on the blockchain.

Prove Transaction

This will generate a proof that you made a payment to a certain address. You need to put the following information:

- (1) **Transaction ID:** This is the ID of the payment you are creating proof for. You can find the transaction details by selecting the History tab from the left menu.
- (2) **Address:** This is the address you are proving payment to.
- (3) **Message (optional):** This an optional message that will be signed with the transaction details. If you choose to include a message then the other party must also include the exact same message in

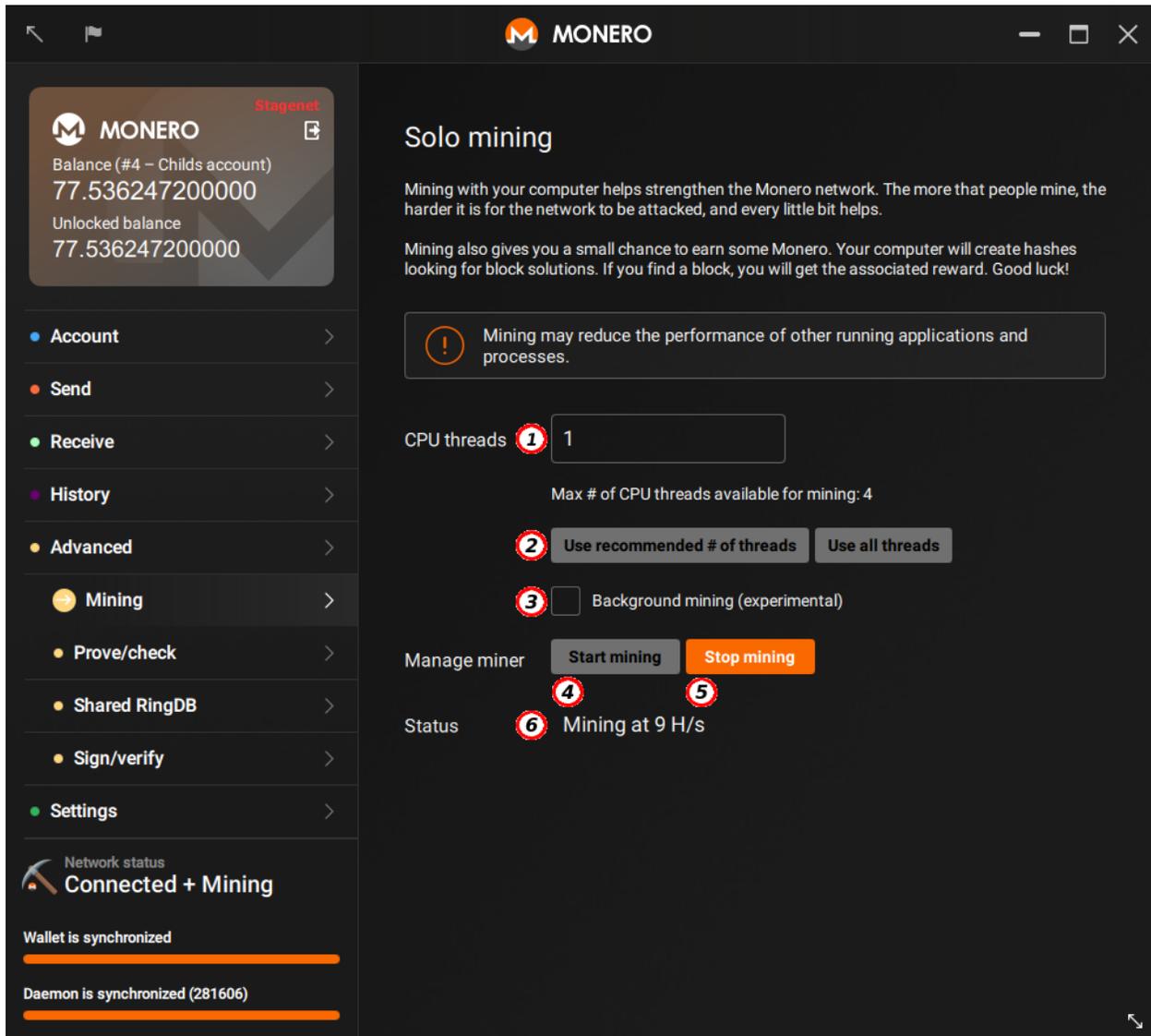


Figure 8.1: mining

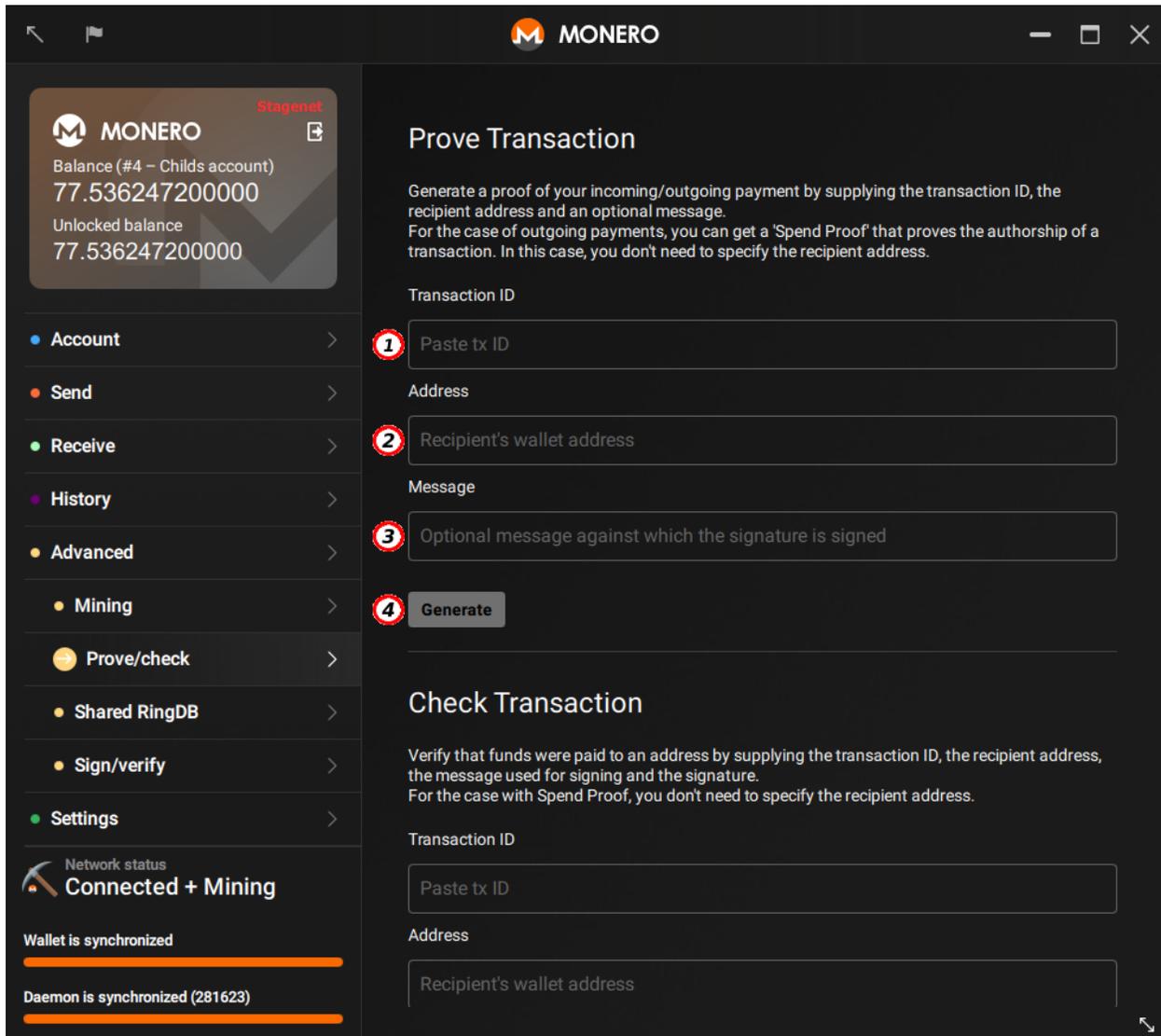


Figure 8.2: Prove payment

order to verify your proof.

(4) **Generate:** Click here once you've entered all the details to generate your proof.

A detailed guide is available on getmonero.org: ['How to prove payment'](#)

Check Transaction

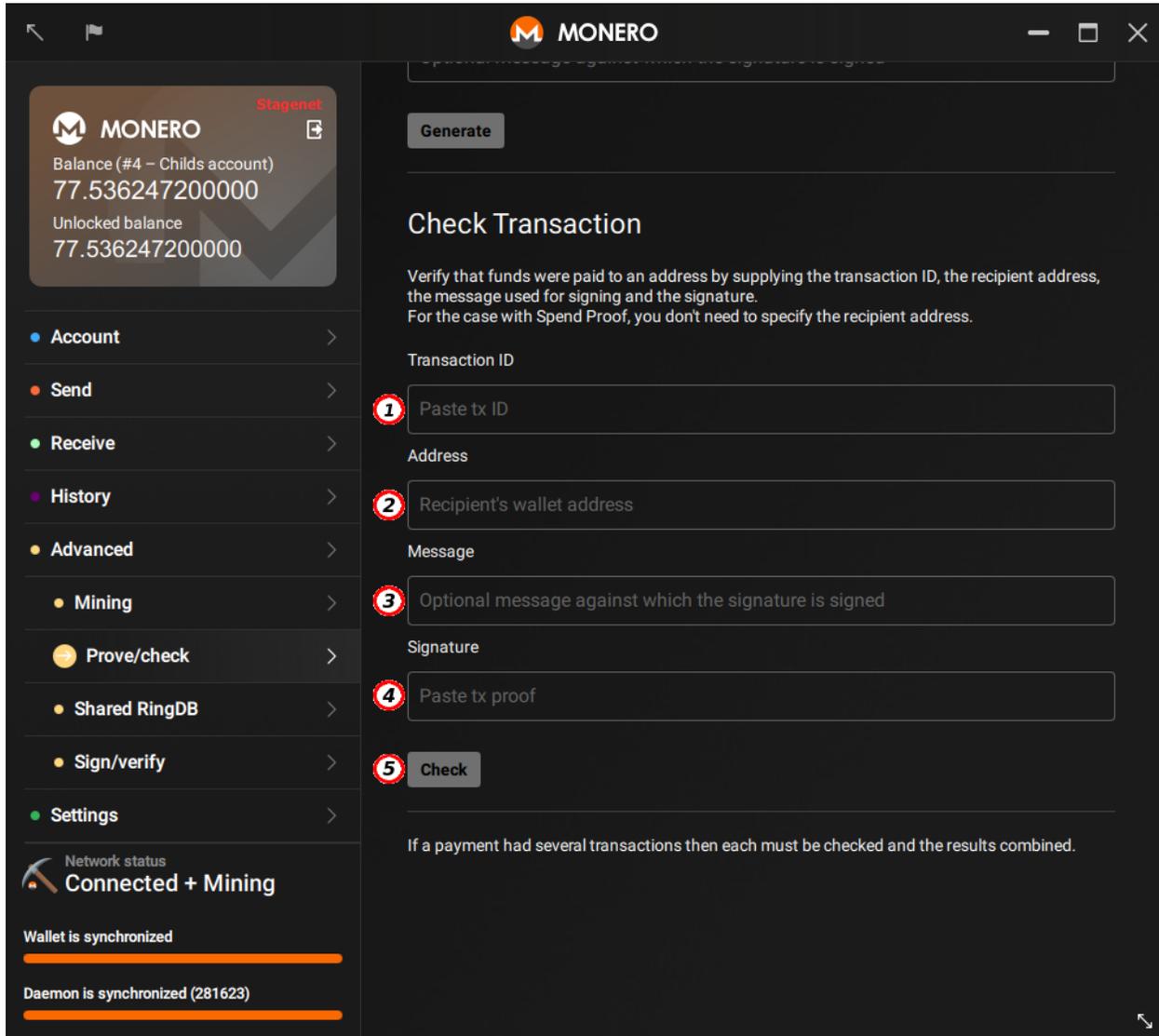


Figure 8.3: Check payment

This will verify that a payment was made. You need to put the following information:

- (1) **Transaction ID:** This is the ID of the payment you are attempting to verify.
- (2) **Address:** This is the receiving address of the payment you are attempting to verify.
- (3) **Message (optional):** This is the optional message that may have been included with the proof.

- (4) **Signature:** This is the signature generated to prove payment.
- (5) **Check:** Click here once you've entered all the details to check that the transaction proof is valid.

Shared RingDB

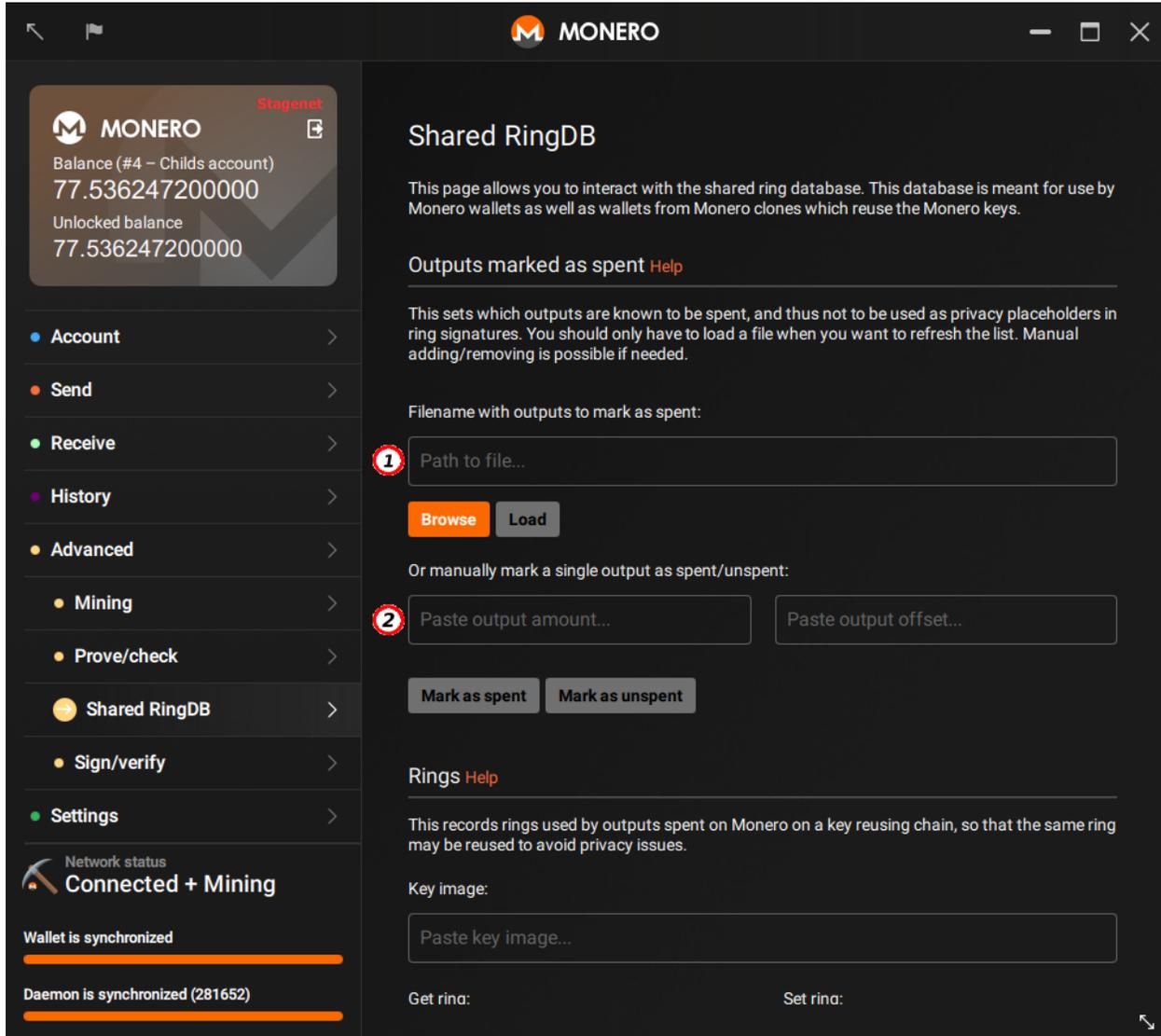


Figure 8.4: shared ringdb 1/2

This is an advanced tool that can be used to improve the privacy of ring signatures. The outputs used in ring signatures can be adapted to mitigate the privacy loss when using a key-reusing fork or to avoid outputs that could not be spent in this transaction.

- (1) **Mark as spent filename:** This tool will mark outputs that are known to be spent. After running `monero-blockchain-mark-spent-outputs`, import the resulting file to avoid using these outputs as decoys in constructed ring signatures. This file is stored in the `.shared-ringdb` folder by default.

- (2) **Mark as spent output:** This will mark or unmark as spent a chosen single output. Outputs are represented by 64-character strings. The outputs added in this field will not be used as decoys in constructed ring signatures. Unmarked outputs may (but will not necessarily) be used as decoys.

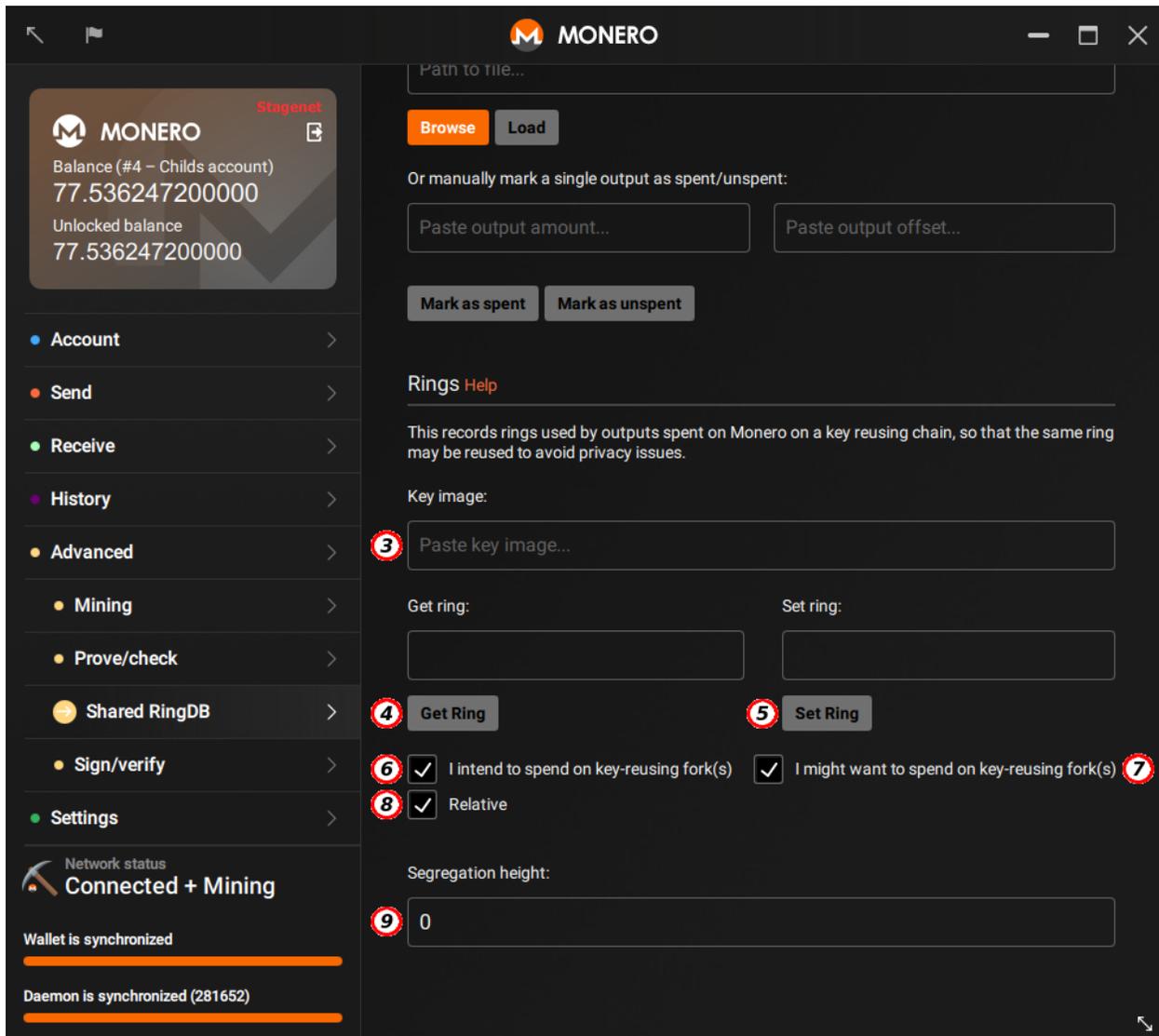


Figure 8.5: shared ringdb 2/2

- (3) **Key image input:** Add the key image that was used on the key-reusing fork.
- (4) **Get ring:** Press the “Get Ring” button to get the ring members for the given key image in 3.
- (5) **Set ring:** Press the “Set Ring” button to set the ring members for a transaction. Copy the ring members from 4 to get those for the key image, or manually type in your own.
- (6) **Intent to spend:** Select this if you are certain that you will spend Monero on a key-reusing fork. This will aggressively modify the input selection algorithm to give you the greatest plausible deniability.
- (7) **Possibility to spend:** Select this if you may spend Monero on a key-reusing fork. This will modify the input selection algorithm. Uncheck this only if you are certain you will not use a key-reusing fork.

- (8) **Relative:** When selected, the offsets are encoded relative to the previous, as opposed to absolute (transactions use relative offsets).
- (9) **Segregation height:** The block height at which the key-reusing fork splits.

Sign - verify Message

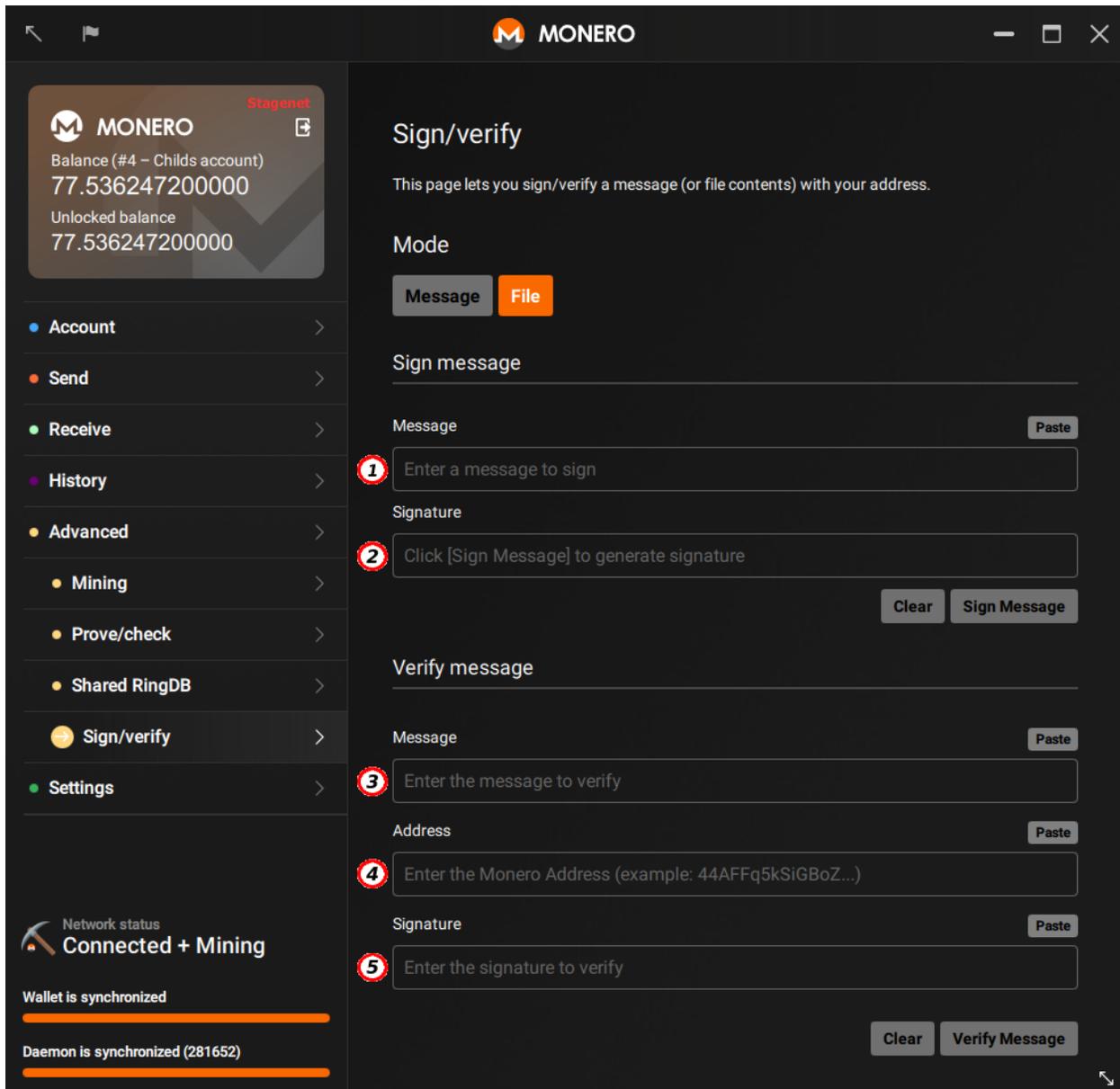


Figure 8.6: sign/verify

The Sign/verify tab provides tools for signing a message or file with your private key or verifying the authenticity of a signed message or file.

Sign

- (1) **Message:** This is where you can enter a message to be signed.
- (2) **Signature:** This is where your unique signature will appear once you click the Sign button. This is linked to your private key and the message you entered. It will be given as proof along with the message which was signed.

Verify

- (3) **Verify message:** This is where you will put a message that has been signed.
- (4) **Address:** This is where you will enter the public Monero address of the signer.
- (5) **Signature:** This is where you will enter the signature you are verifying. Once all the required information has been entered click the Verify button. A pop-up will tell you if the signature is valid.

Sign - verify File

The **Sign/verify** tab provides tools for signing a message or file with your private key or verifying the authenticity of a signed message or file.

Sign

- (1) **File:** This is where you can choose a file to be signed. Click Browse to navigate the file system.
- (2) **Signature:** This is where your unique signature will appear once you click the Sign button. This is linked to your private key and the file you entered. It will be given as proof along with the file which was signed.

Verify

- (3) **File:** This is where you enter the path to a file that has been signed. Click Browse to navigate the file system.
- (4) **Address:** This is where you will enter the public Monero address of the signer.
- (5) **Signature:** This is where you will enter the signature you are verifying. Once all the required information has been entered click the Verify button. A pop-up will tell you if the signature is valid.

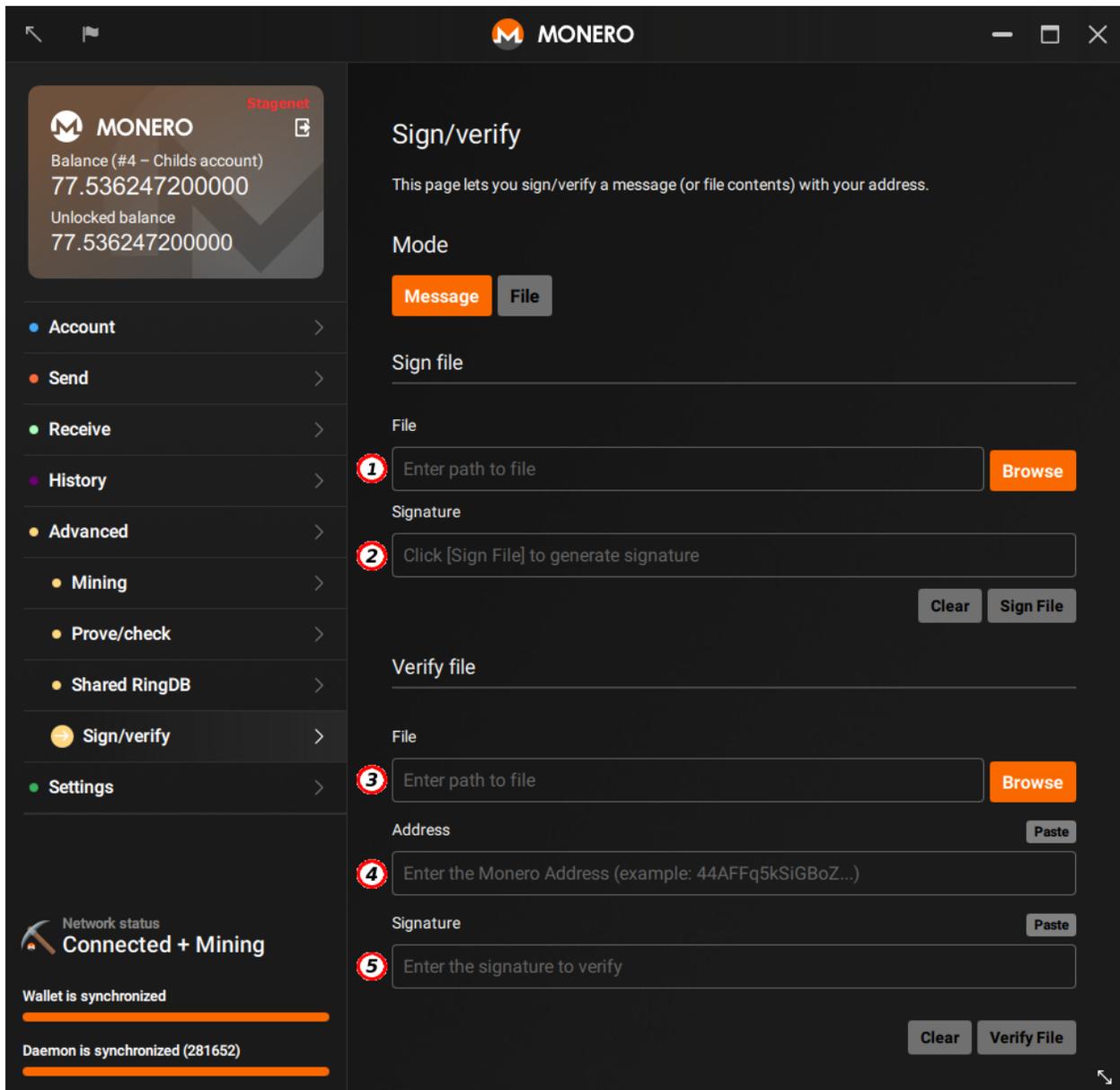


Figure 8.7: sign/verify

Chapter 9

Settings

The **Settings** tab provides tools for customizing configuration options.

Wallet

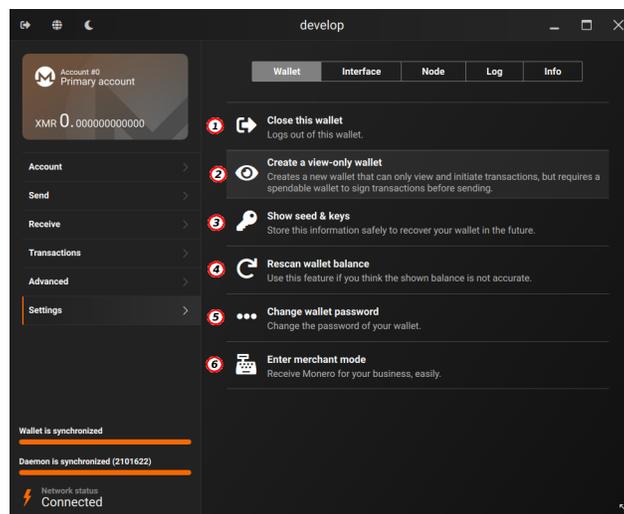


Figure 9.1: settings-wallet

- (1) **Close this wallet:** Click here to safely close the current wallet and open the main menu.
- (2) **Create a view-only wallet:** Click here to generate a view-only (or audit) wallet file, which is capable to see all incoming transactions. In order to display the real balance, you must also import the key images from your normal wallet into your view-only wallet. The view-only wallet will be created with the same password as the current wallet. See this [user-guide](#) for more information.
- (3) **Show seed & keys:** This button will redirect you to the [Seed and keys](#) tab.
- (4) **Rescan wallet balance:** Click here to rescan all spent outputs from your wallet, updating your wallet balance. If your balance is still incorrect, you probably will have to resync your wallet after changing the wallet restore height to a restore height before the height of the first transaction to your

wallet. To do this, go to the Settings > Info page, Wallet restore height, and click on the (Change) link.

(5) **Change wallet password:** Click here to change the password of your wallet file.

(6) **Enter merchant mode:** Click here to open Monero GUI's business-friendly mode.

Interface

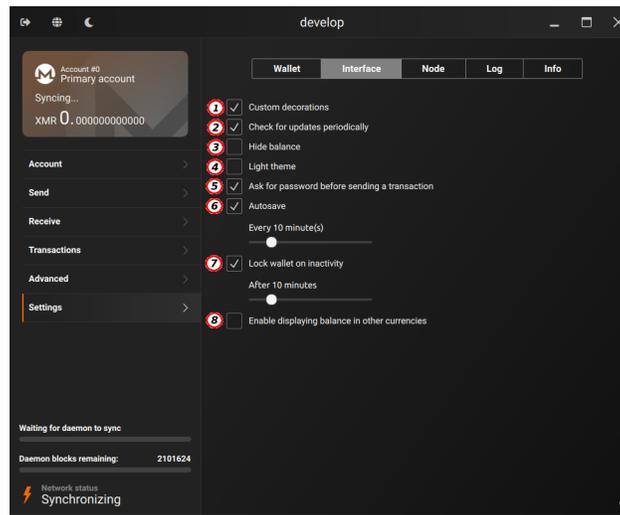


Figure 9.2: settings_interface

The **Interface** tab is where you can adjust appearance and security settings of the GUI wallet. The options are:

- (1) **Custom decorations:** Switch custom window decorations.
- (2) **Check for updates periodically:** Enable this option if you want your wallet to periodically check if a new version is available.
- (3) **Hide balance:** Will display your balance as `?.??` in the balance card. It is a useful privacy feature if you are running your wallet in a public place, as a point of sales system, or if you have to take a screenshot or record your screen to someone.
- (4) **Light theme:** Switch between light and dark themes.
- (5) **Ask for password before sending a transaction:** Disable this option if you don't want to type your password every time a transaction is sent. This option is enabled by default.
- (6) **Autosave:** When enabled, your wallet file will be saved automatically, every 10 minutes (default) or at an interval of your choice. This option is enabled by default.
- (7) **Lock wallet on inactivity:** Set this to automatically lock your wallet after N minutes of inactivity. This ensures no one is able to use your wallet while you are away from keyboard.
- (8) **Enable displaying balance in other currencies:** Gives you the possibility to also display your balance, the transaction amount and transaction fee in other currencies. You can select the price source and the currency you prefer (currently only *USD* and *EUR* are supported).

Local Node

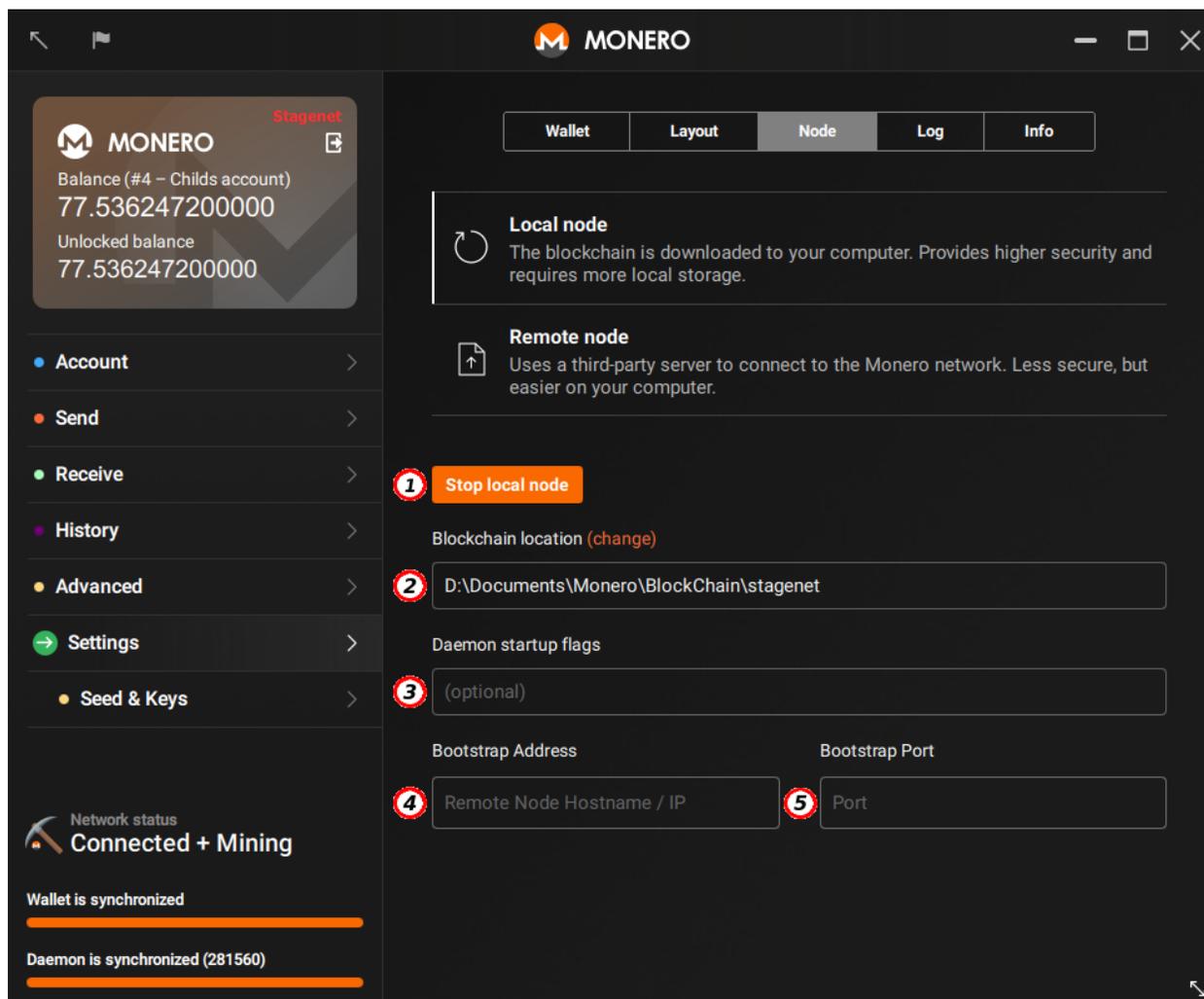


Figure 9.3: settings_local-node

- (1) **Start/Stop Local Node:** Depending on its current state, either start or stop the local node.
- (2) **Blockchain location:** Manually enter a non-default path to the blockchain, or click **change** to choose a new path.
- (3) **Startup flags:** When using a local node, this will be where you enter additional command line options.
- (4) **Bootstrap Address:** Enter the hostname or IP address of the bootstrap remote node. See section [Bootstrap nodes](#) for a brief explanation of what a bootstrap node is.
- (5) **Bootstrap Port:** Enter the port of the bootstrap remote node.

Remote Node

Use a remote node, do not download the blockchain. Check the '[About remote nodes](#)' section of this guide.

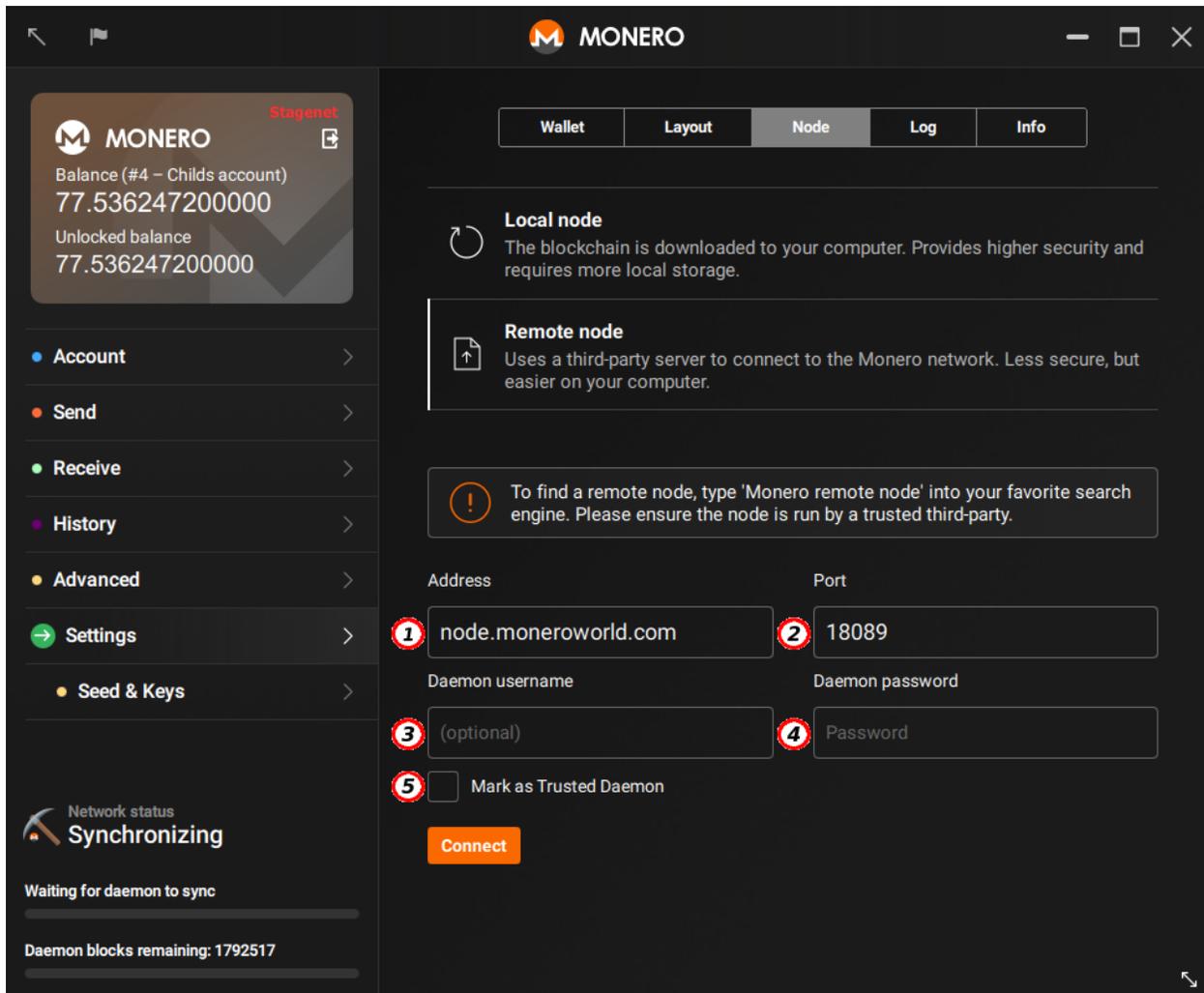


Figure 9.4: settings_remote-node

- (1) **Address:** Enter the hostname or IP address of the remote node.
- (2) **Port:** Enter the port of the remote node.
- (3) **Daemon Username:** Enter a username in case authentication to the remote node is required.
- (4) **Daemon Password:** Enter a password in case authentication to the remote node is required.
- (5) **Trusted Daemon:** By default, data requests to remote nodes are obfuscated, and this process requires more resources. If your remote node is under your control (i.e. a dedicated server of your own) you can mark it trusted so that data requests will not be obfuscated anymore (just like a local node).

Log

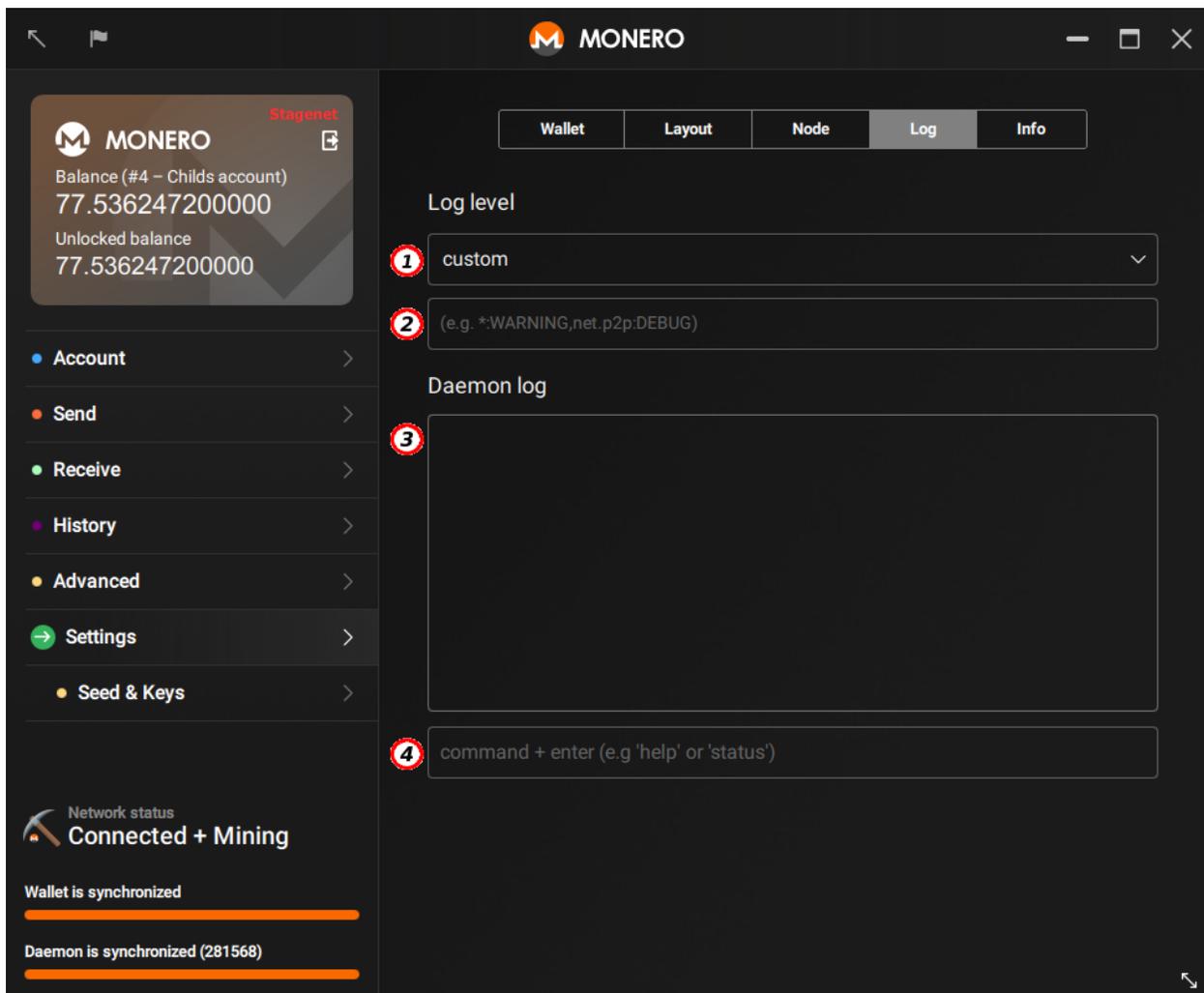


Figure 9.5: settings-log

- (1) **Log level:** Change the verbosity of the debug logs.
- (2) **Log categories:** Add specific categories to the debug logs.

- (3) **Daemon log:** Real time output of the log.
- (4) **Command line:** Interact with the daemon.

Info

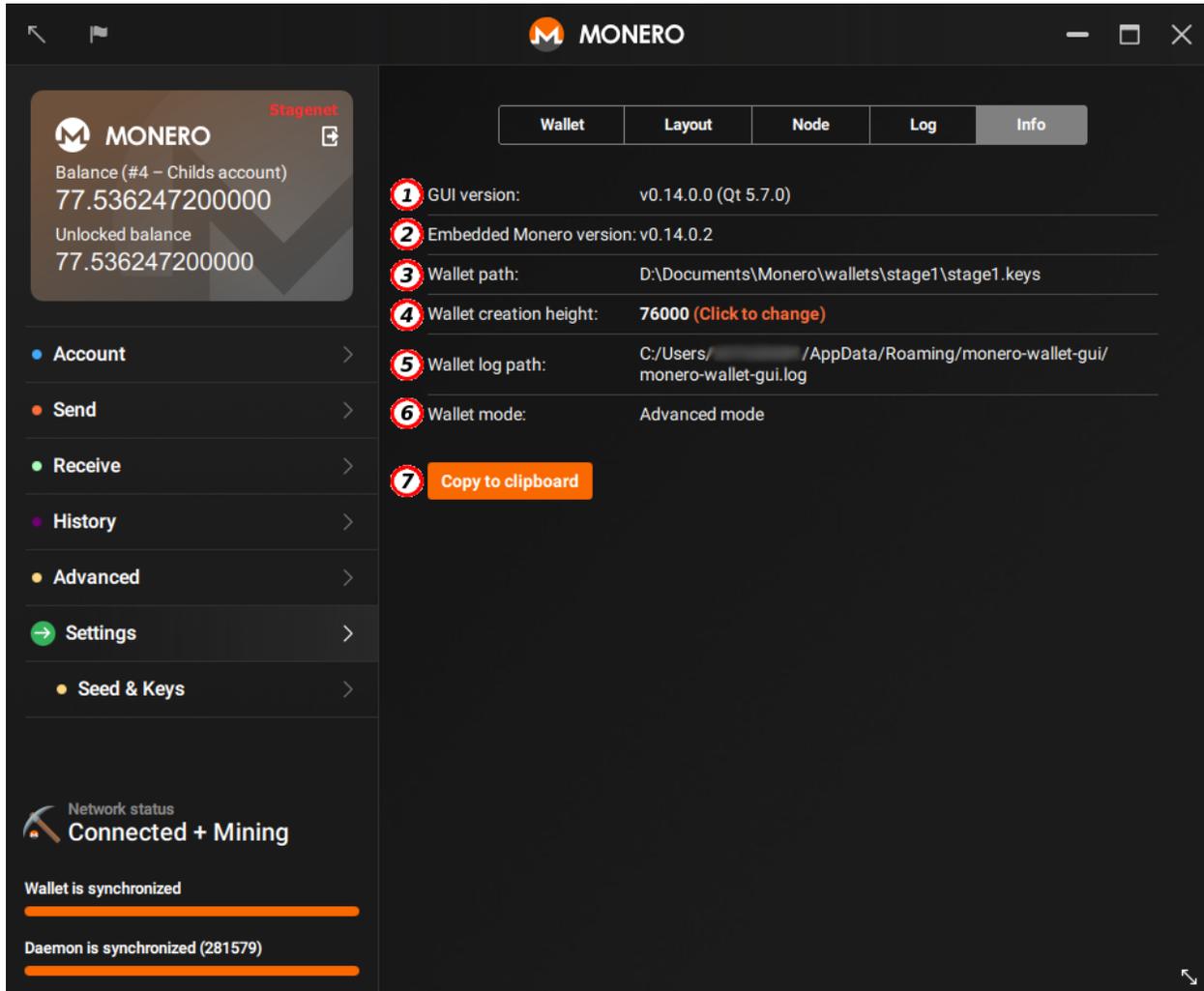


Figure 9.6: settings-info

- (1) **GUI Version:** Version of the GUI wallet installed.
- (2) **Embedded Monero Version:** Version of the embedded daemon in use.
- (3) **Wallet path:** Where the wallet is located on your computer.
- (4) **Wallet restore height:** Your wallet synchronization starts at the block defined in wallet restore height. Therefore, you must set the restore height before the height of the first incoming transaction of your wallet. If you click on the (Change) link, you can set a blockheight or a date. After changing it, your wallet will rescan the blockchain starting from the blockheight you set. *Note: when you change your wallet's restore height, you will lose all information that is stored in the wallet software*

and cannot be recovered from the blockchain, which includes destination addresses, transaction keys, and descriptions (notes).

- (5) **Wallet log path:** Where the logs for this wallet will be saved.
- (6) **Wallet mode:** Display the wallet mode chosen in the **wizard**.
- (7) **Graphics mode:** Display the graphics mode.
- (8) **Copy to clipboard:** Copy all information to clipboard.

Seed and keys

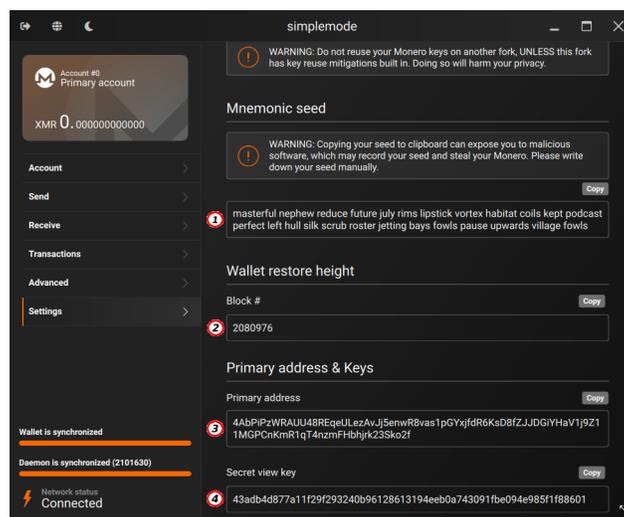


Figure 9.7: seed-keys

The **Seeds & Keys** tab displays your wallets mnemonic seed as well as your secret view key, public view key, secret spend key, and public spend key.

- (1) **Mnemonic seed:** **DO NOT share your mnemonic seed with anyone. Store a copy securely.** The mnemonic seed is a 25 word phrase that contains all the information needed to view and spend funds. *Learn more about [mnemonic seeds](#).*
- (2) **Wallet restore height.** This is the blockheight of the first incoming transaction of your wallet. This information is useful when you restore your wallet, because you will only have to synchronize the blockchain for the period that is needed.
- (3) **Primary address:** This is the main address of your wallet and starts with 4. All other addresses from your wallet are subaddresses, and start with 8.
- (4) **Secret view key:** Secret (private) view keys allows the holder to view your wallets incoming transactions, but not outgoing. It is sometimes useful for auditing purposes to give your secret view key to a third party.
- (5) **Public view key:** The public view key is used for stealth address creation. *Learn more about [view keys](#).*

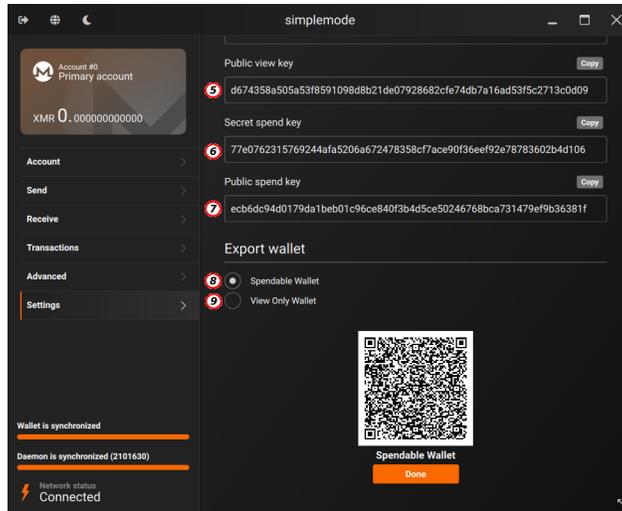


Figure 9.8: seed-keys-2

- (6) **Secret spend key:** **DO NOT** share your secret spend key with anyone. The secret (private) spend key is used to sign transactions and should be regarded with the same security as your mnemonic seed.
- (7) **Public spend key:** The public spend key is used by the network to verify the signature of the key image you generate when you make a transaction. This is what prevents double-spends as the network enforces the rule that a key image can be spent only once. *Learn more about [spend keys](#).*
- (8) **Export Spendable Wallet:** **DO NOT** share your spendable wallet QR code with anyone. This can be used like a mnemonic seed for recovering your wallet. This creates a qr code that contains all of your keys.
- (9) **Export View Only Wallet:** This creates a QR code that contains only the keys for viewing the transactions that this wallet sends or receives, but cannot create transactions.

Chapter 10

Binaries Verification

Verify that the files you downloaded match the official ones. You can use [this step-by-step guide](#) with pictures (easy, for Windows user).

Chapter 11

About remote nodes

Remote nodes are useful if you are not able/don't want to download the whole blockchain, but be advised that malicious remote nodes could compromise some privacy. They may track your IP address, track your "restore height" and associated block request data, and send you inaccurate information to learn more about transactions you make. Please make sure to use a node you trust (hopefully your own) and remain aware of these limitations. Users who prioritize privacy should use the wallet in advanced mode and start a local node without setting a bootstrap node. A number of public remote nodes are listed at moneroworld.com and [node-o-matic](http://node-o-matic.com)

Bootstrap nodes

Remote nodes can also be useful if you are running a local node and want to use Monero before the blockchain download has finished. In this situation, the remote node is known as a bootstrap node. Your local node will temporarily connect to it while the blockchain is still being downloaded. When the blockchain is synced locally, your local node will disconnect from the bootstrap node. It is a reasonable trade-off for most people who care about privacy but also want the convenience of an automatic fallback option. Be aware that when using remote nodes (including a local node with a bootstrap node), nodes may track your IP address, your "restore height", and associated block request data, as well as send you inaccurate information to learn more about transactions you make. Please make sure to use a remote node you trust (hopefully your own) and remain aware of these limitations. Keep the Monero full node software operating on your computer even when you are not using the wallet to reduce the amount of leaked data. More information at <https://getmonero.org/resources/moneropedia/bootstrap-node.html>

Chapter 12

Common issues and solutions

The following are only some of the most common issues. For a more extensive list visit the [FAQ](#) page on [getmonero.org](#).

- **How do I generate a Ledger Monero wallet with the GUI?**
- **I am missing (not seeing) a transaction to (in) the GUI (zero balance)**
- **I am using the GUI and my daemon doesn't start anymore**
- **Transaction stuck as “pending” in the GUI**
- **My GUI feels buggy / freezes all the time**
- **My name contains a special (non-ASCII) character (e.g. é, ø, â, Ö) and I can't create a wallet with the GUI**
- **The GUI uses all my bandwidth and I can't browse anymore or use another application that requires internet connection**
- **How do I move the blockchain (data.mdb) to a different directory during (or after) the initial sync without losing the progress?**
- **How do I change the language of the 25 word mnemonic seed in the GUI or CLI?**