ComplyFirst

TECHNICAL COMPLIANCE GUIDE

Digital Asset | **Monero (XMR)** | **December 7, 2020**

Overview

This guide provides step by step technical guidance on how to request and handle the view key and key images for a specific Monero address. This will allow one to see both incoming and outgoing transactions to that address, in order to better ascertain risk level. This guide was last updated for Monero v0.17.1.4 and is meant for Windows 10, though similar steps also work on Linux and Mac.

NOTE: Because of the time and effort required, this procedure is only recommended to be used in very specific and limited circumstances, when standard evaluative efforts are found to be insufficient. Please do not rely on this for the standard course of interacting with users trading or otherwise interacting with XMR.

Table of Contents

Exporting data (for users)

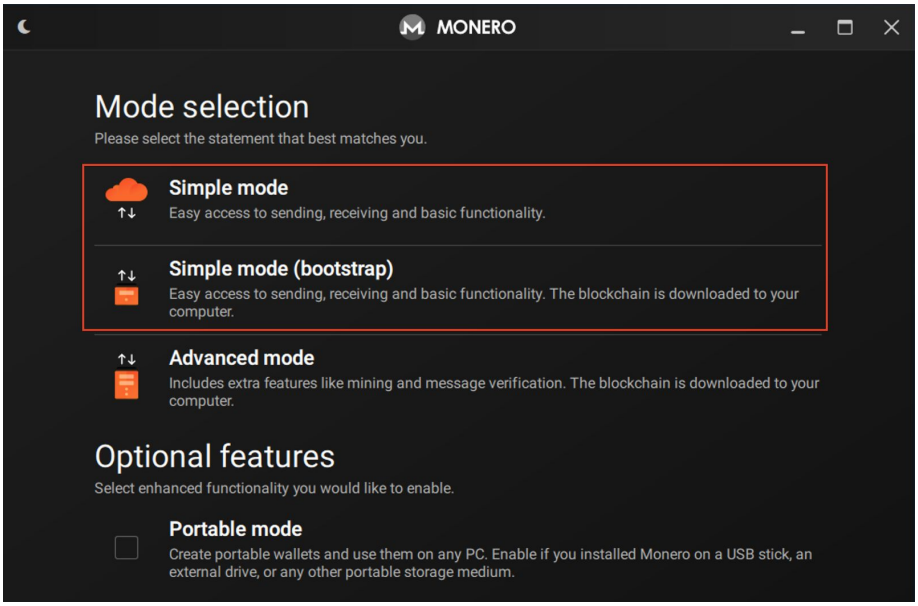
These steps are performed by the user in order to export view key and private key images from a user that is self-custoding their funds, using the official Monero GUI and command line (CLI) wallets.

NOTE: The user may be using wallet software other than the official Monero wallet to store their XMR. In that case, please check with the wallet developer as to whether an export of the viewkey and key images is possible. If not, it may be possible for the user to export their private key from their wallet software, import it into the official Monero wallet, and provide the requested information that way.

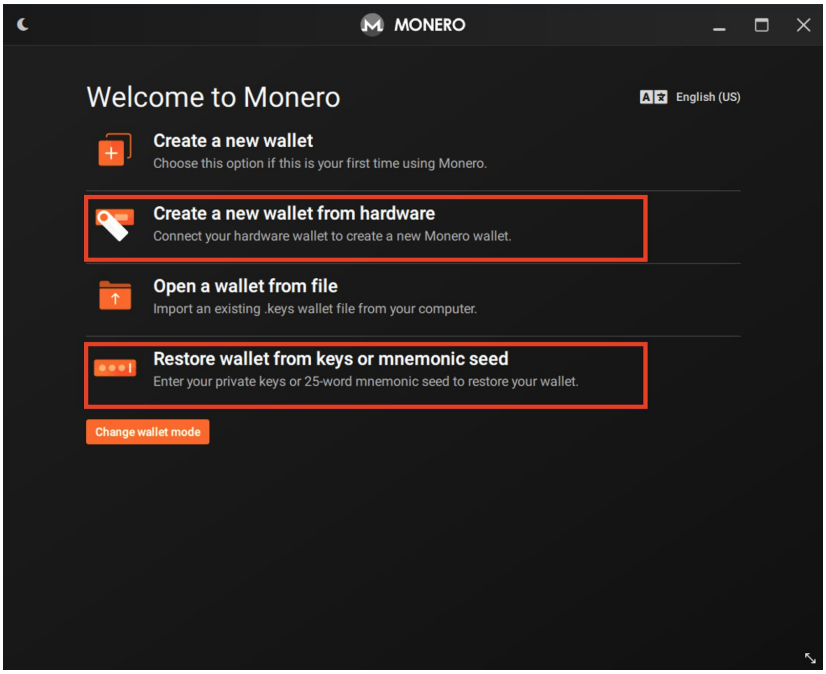
Baselining

This guide assumes that the user already has the official Monero wallet installed and functional. If this is not the case, the user should follow the instructions in this section.

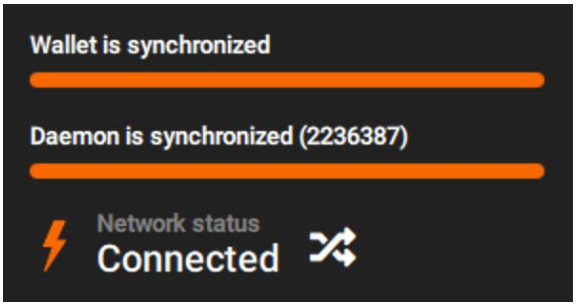
1. Download the Monero GUI Wallet for your platform from https://www.getmonero.org/downloads/#gui and install it.
2. Launch the GUI wallet and continue through language selection. On the **Mode selection** screen, choose **Simple mode** or **Simple mode (bootstrap)**, depending on whether you want to download the entire blockchain to your hard drive or not (note that downloading the blockchain will require a SSD drive with at least 100GB of space). Select **Portable mode** *only if applicable*. Then click through the warning message if one appears.



3. On the next screen, import the XMR wallet data over as appropriate, by either:
    a. Exporting the wallet's mnemonic seed from the other wallet software (if not using a hardware wallet device) and clicking the **Restore wallet from keys or mnemonic seed** option
    b. Restoring a wallet from their Ledger or Trezor (if using a hardware wallet device) by clicking on the **Create a new wallet from hardware** option, and then choosing the **Restore a wallet from device** option on the next dialogue page.

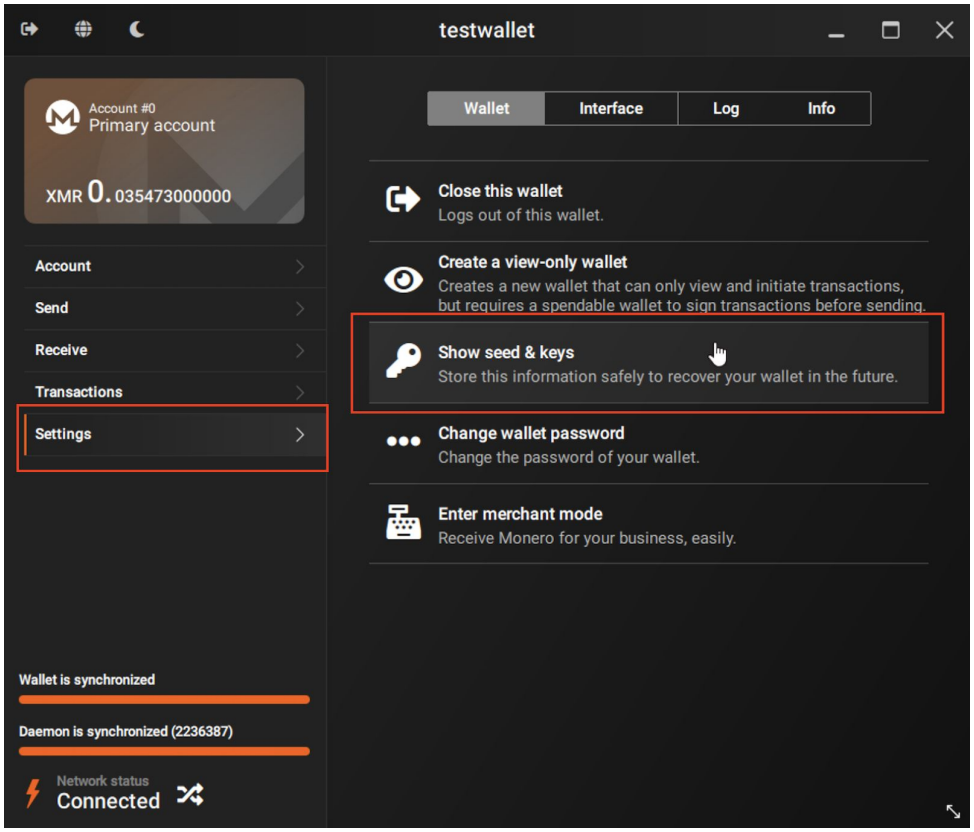


4. Before the user moves on, they should be at the main wallet screen in the Monero GUI and have it fully synchronized (which may take several hours if they had to import from seed or hardware wallet). The bottom left hand part of the GUI window should look similar to the below image before proceeding (specifically, note that both the wallet and daemon are “synchronized”):

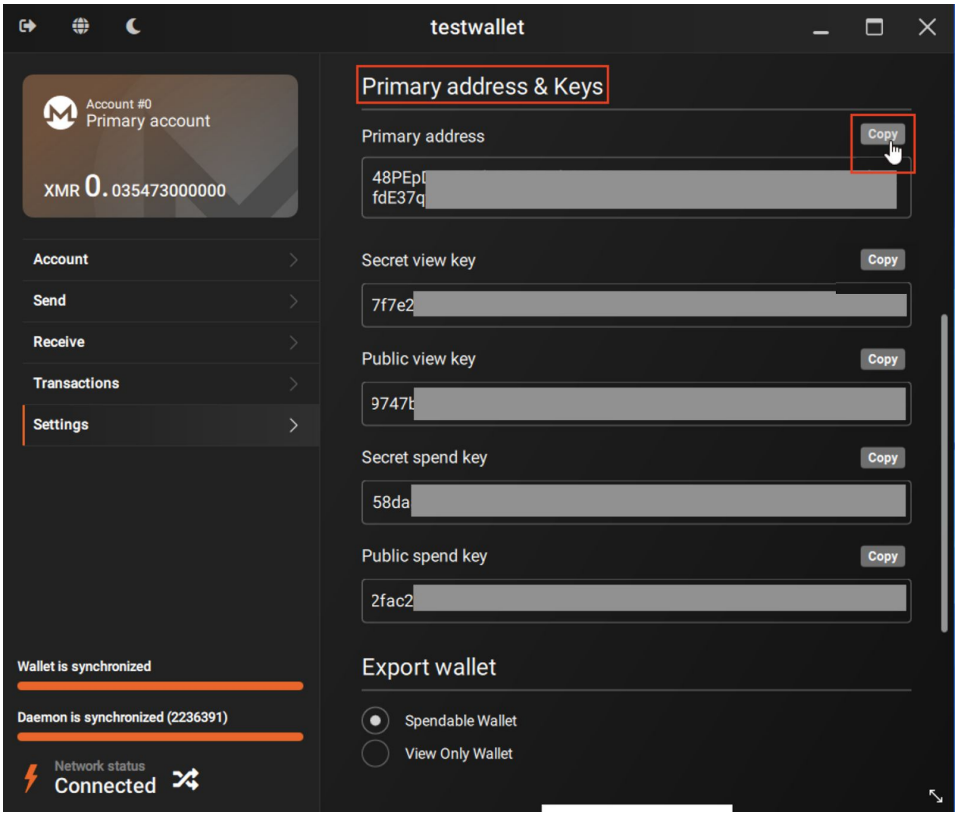


Obtaining the address and private view key

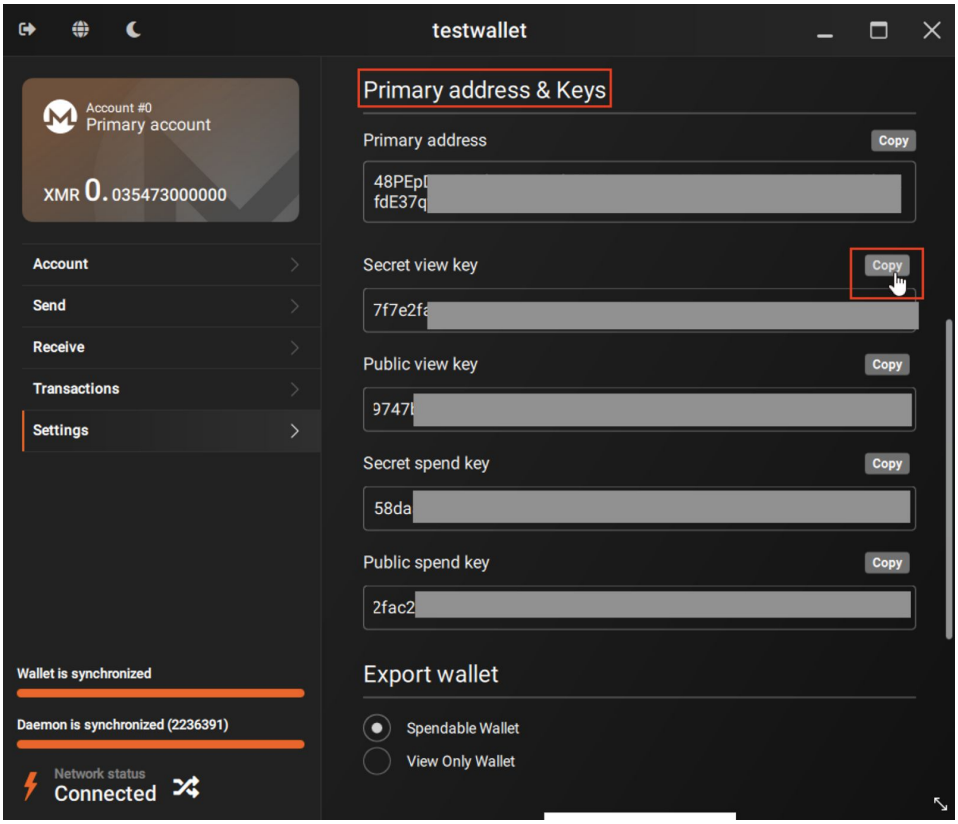
1. Click on **Settings**, then **Show seed & keys**:



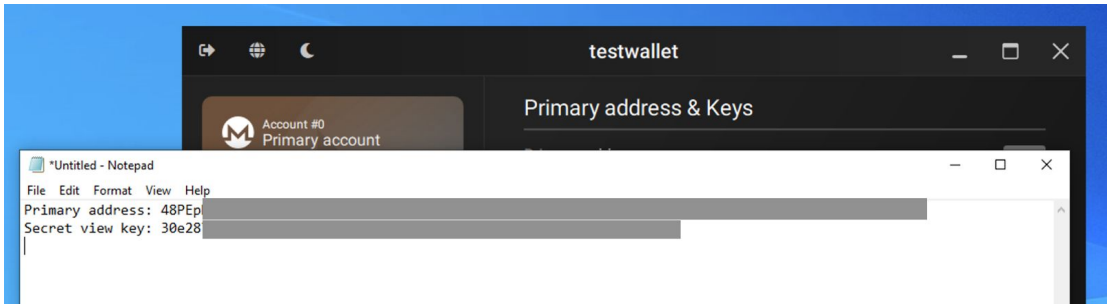
2. Enter the wallet password when prompted
3. On the page that appears, scroll down to the **Primary address & Keys** section, find the **Copy** button for the **Primary address** item and click it to copy the data to the clipboard:



4. Open up Notepad or another text editor (e.g. enter “notepad” on the Start menu search bar and click enter). Paste the address into the Notepad window.
5. Next, click the **Copy** button for the **Secret view key** and click it to copy it to your clipboard. NOTE: Ensure you copy only the **Secret view key** and ***NOT*** *any other key:*



6. Paste the secret view key into the Notepad/text editor window as well:

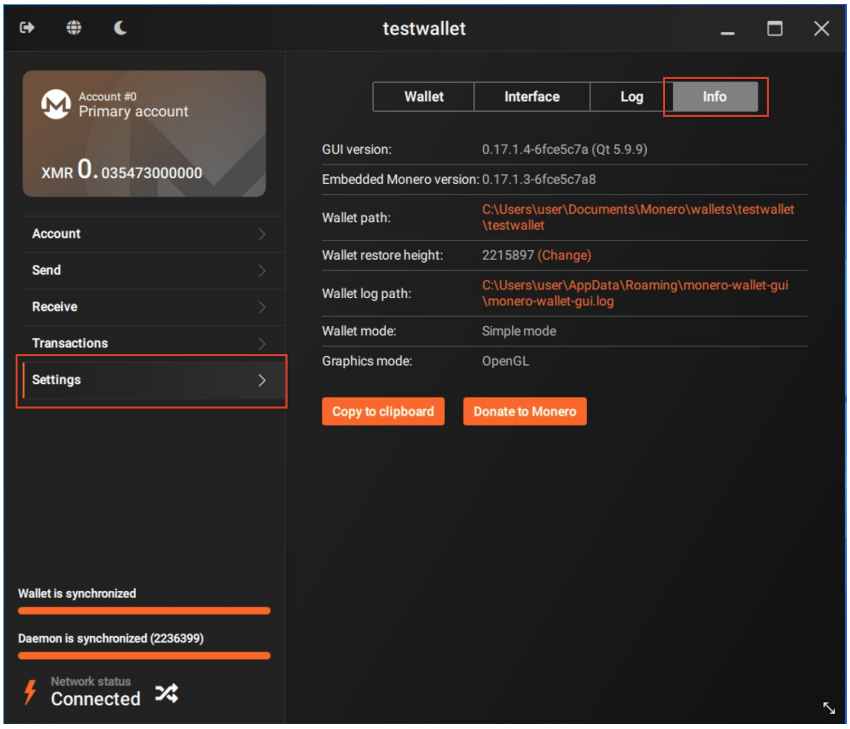


7. This information should be shared as requested with the compliance personnel. If compliance asks for the private key images file, include it using the steps below.

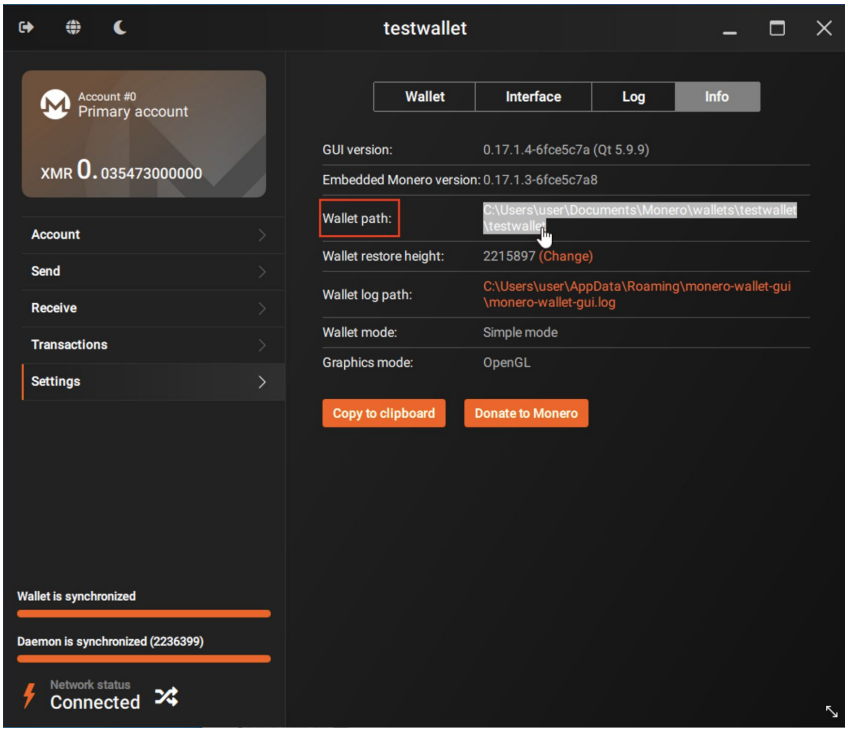
Obtaining the private key images

NOTE: Private key images are not the same as the private spend key. They only provide the ability to see outgoing transfers, and do not provide others the ability to spend funds. None of the data requested will allow the requestor to spend funds. Never share the secret spend key!

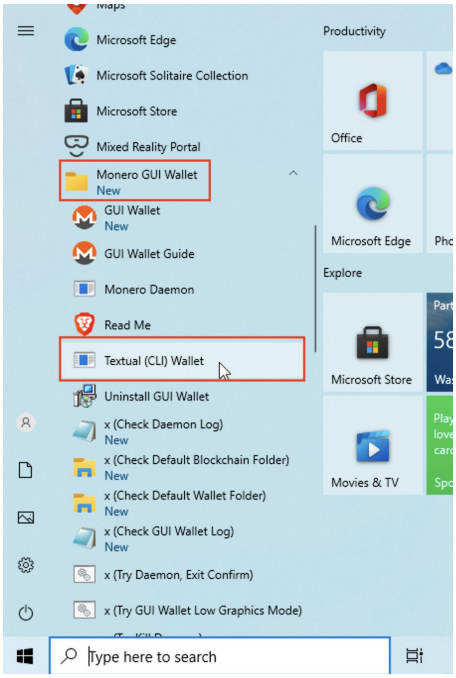
1. In the Monero GUI, find the file path of your wallet by going to **Settings | Info**:



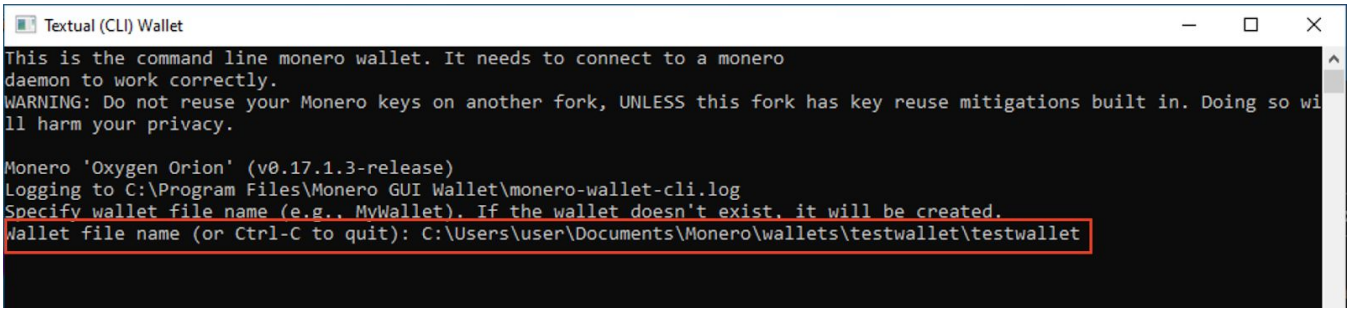
2. Highlight the file path to the right of **Wallet path** and press CTRL-C to copy it:



3. Launch the Monero command line wallet interface by going to **Start | Monero GUI Wallet | Textual (CLI) Wallet** (or by running **monero-wallet-cli.exe** in the downloaded folder):

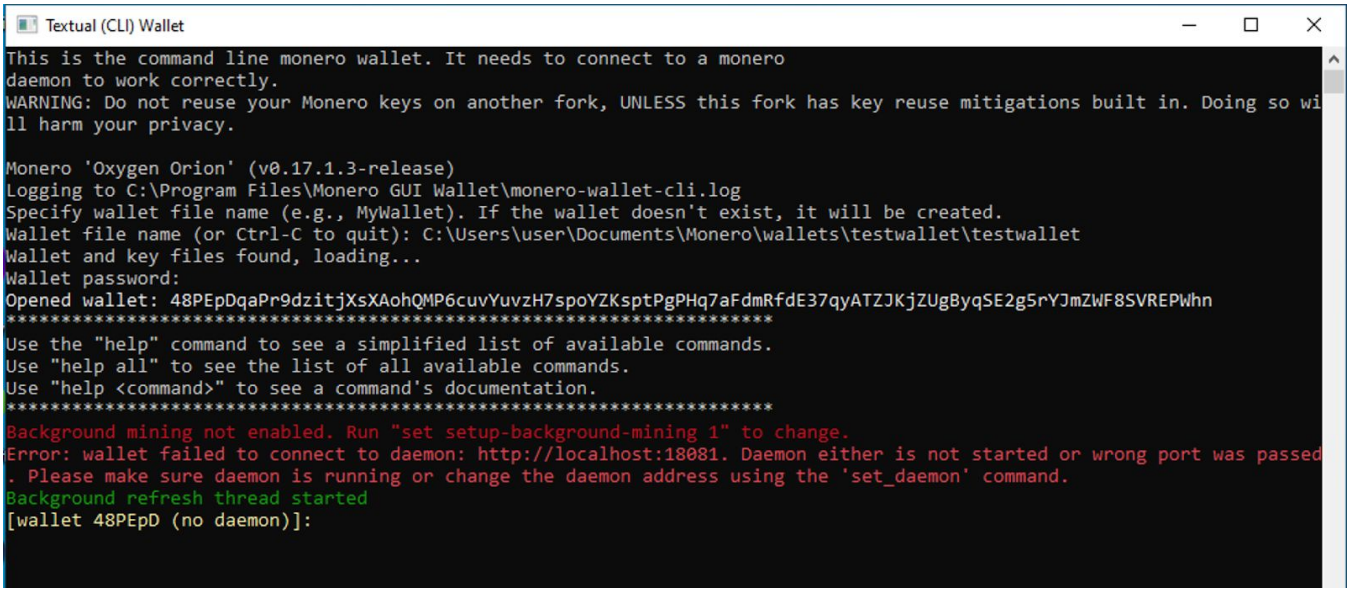


4. A command window prompting you for a Wallet file name should appear. Paste the wallet file path you copied in step 2 above into the command window and press enter:



NOTE: If the file path contains spaces, you will need to enclose the path in double quotation marks, for example "`C:\Users\user\My Wallet Folder`"

5. Close the Monero GUI wallet window before proceeding further and wait for it to disappear.
6. Back in the CLI wallet window, enter your wallet password and press enter. You will see a screen appear like the following (if prompted to enable background mining, enter "N" for no):



7. Enter the following command: **`export_key_images all C:\keyimages.dat`**. Press enter. You will be prompted for your wallet password. Enter it and press enter. The key images file should be created at the path specified.

8. Type **`exit`** to quit the CLI wallet.
9. Locate the keyimages.dat file you created and include it in your response to the compliance personnel. *To reiterate, this file just contains data of spend transactions. It does not give anyone access to your wallet to spend your funds.*

Importing data (for compliance personnel)

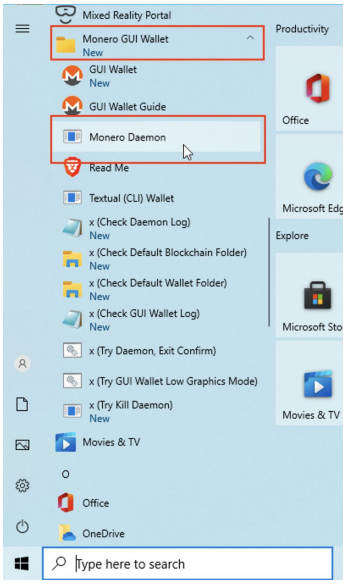
These steps are performed by the compliance professional, to get a view-only look at a specific Monero address, from a user's exported view key and private key images.

System requirements

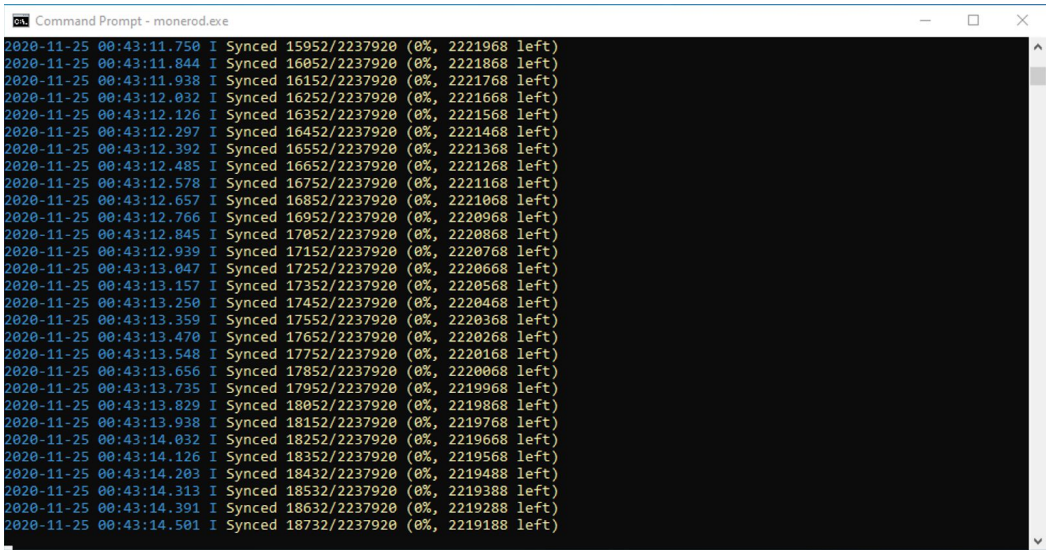
- Relatively recent CPU (4 or more cores preferred)
- SSD drive for the blockchain data (downloading the Monero blockchain to a non-SSD computer is very slow and definitely not recommended)
- At least 100GB of free space on the SSD

Installing and starting the Monero Daemon

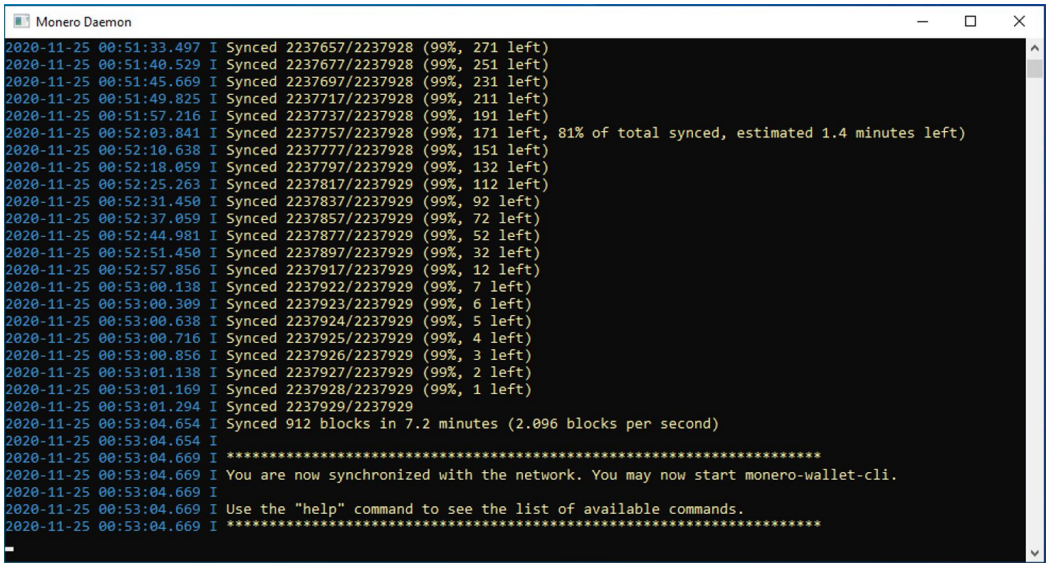
1. Download the Monero GUI Wallet for your platform from https://www.getmonero.org/downloads/#gui and install.
2. Launch monerod (the monero daemon) by clicking **Start | Monero GUI Wallet | Monero Daemon**. (Alternatively, you can launch a command prompt and type: `cd "C:\Program Files\Monero GUI Wallet" && monerod.exe`)



3. After a few minutes, synchronization will start:



4. Allow the daemon program to synchronize with the network. <u>This will take up to a day to finish, but only needs to be done once, so be patient.</u> When done, the daemon window state the synchronization is complete:



5. Once done synchronizing, you are now ready to create watch only user wallets, in order to verify transaction activity.
6. Create a directory on your computer to contain imported user wallets. We will use "*C:\walletdata*" in this tutorial.

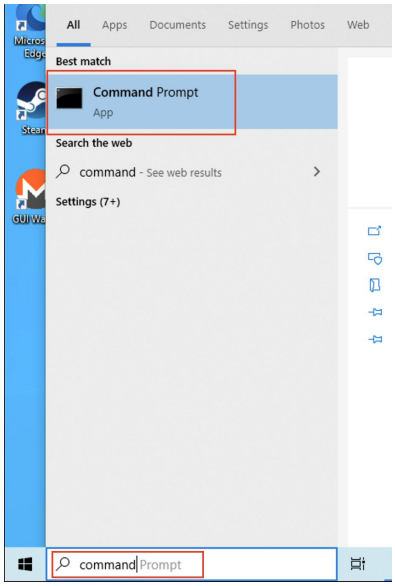
Importing user wallet data

This section presumes that you have obtained the following information from a user who followed the procedure detailed in <u>this section</u>:

- Monero address (base wallet address)
- Private view key

- Signed key images file (the examples assume you have placed this file in "*C:\walletdata*" and called it "*20201122-joe_jones-keyimages.dat*")

1. If you haven't already, launch the Monero Daemon (detailed earlier) and allow it to finish synchronizing to the blockchain.
2. Launch a command prompt. On Windows 10, this can be done by typing **command** in the start menu search bar, and choosing **Command Prompt** from the list of results that appear:



3. In the Command Prompt window that appears, change to the Monero GUI wallet directory. On Windows (assuming you installed to the default location), you would do this via the command:
   `cd "C:\Program Files\Monero GUI Wallet"`
4. Type the following command and press enter the following command:
   `monero-wallet-cli.exe --generate-from-view-key <WALLET_PATH>`
   Here, <WALLET_PATH> is the path to and the filename of the new wallet to create. It is good to use something like maybe the date as well as the user's ID in the filename. In the example below, we use "*C:\walletdata\20201122-joe_jones*":
5. Upon issuing the command successfully, you will be asked for the following:
   a. A "Standard address". Paste in the user's primary wallet address here and press enter.
   b. The "Secret view key". Paste in the user's secret view key and press enter.
   c. A wallet password. Using something basic like "123456" is fine here, as it is a read-only wallet and you can delete the wallet when done. You will be asked to confirm the password after entering it the first time.
   d. "Restore from specific blockchain height": Type **0** here and press enter.
   e. Whether to enable background mining or not. Type **N** here and press enter:

```
C:\Program Files\Monero GUI Wallet>monero-wallet-cli.exe --generate-from-view-key C:\walletdata\20201122-joe_jones
This is the command line monero wallet. It needs to connect to a monero
daemon to work correctly.
WARNING: Do not reuse your Monero keys on another fork, UNLESS this fork has key reuse mitigations built in. Doing so wi
ll harm your privacy.

Monero 'Oxygen Orion' (v0.17.1.3-release)
Logging to C:\Program Files\Monero GUI Wallet\monero-wallet-cli.log
Standard address: 48PEpDqaPr9dzitjXsXAohQMP6cuvYuvzH7spoYZKsptPgPHq7aFdmRfdE37qyATZJKjZUgByqSE2g5rYJmZWF8SVREPWhn
Secret view key:
Enter a new password for the wallet:
Confirm password:
Generated new wallet: 48PEpDqaPr9dzitjXsXAohQMP6cuvYuvzH7spoYZKsptPgPHq7aFdmRfdE37qyATZJKjZUgByqSE2g5rYJmZWF8SVREPWhn
Restore from specific blockchain height (optional, default 0),
or alternatively from specific date (YYYY-MM-DD): 0
The daemon is not set up to background mine.
With background mining enabled, the daemon will mine when idle and not on battery.
Enabling this supports the network you are using, and makes you eligible for receiving new monero
Do you want to do it now? (Y/Yes/N/No): : N
Background mining not enabled. Set setup-background-mining to 1 to change.
If you are new to Monero, type "welcome" for a brief overview.
Starting refresh...
Height 41959 / 2236426
```

6. The wallet refresh process then begins. This will take some time to complete (possibly an hour or more). Once done, you will be dropped into the wallet command prompt:

```
Command Prompt - monero-wallet-cli.exe --generate-from-view-key C:\walletdata\20201122-joe_jones
Secret view key:
Enter a new password for the wallet:
Confirm password:
Generated new wallet: 48PEpDqaPr9dzitjXsXAohQMP6cuvYuvzH7spoYZKsptPgPHq7aFdmRfdE37qyATZJKjZUgByqSE2g5rYJmZWF8SVREPWhn
Restore from specific blockchain height (optional, default 0),
or alternatively from specific date (YYYY-MM-DD): 2200000
The daemon is not set up to background mine.
With background mining enabled, the daemon will mine when idle and not on battery.
Enabling this supports the network you are using, and makes you eligible for receiving new monero
Do you want to do it now? (Y/Yes/N/No): : n
Background mining not enabled. Set setup-background-mining to 1 to change.
If you are new to Monero, type "welcome" for a brief overview.
Starting refresh...
Height 2236357, txid <4e934349bcbd5c10f186d449dfae7d193dee2f881f3b307bcd322747bbff07df>, 0.009900000000, idx 0/0
Height 2236360, txid <2f456e0b01acfbab2c2bbe24795d1169cf3d4c97e9085d504d1195599fca7051>, 0.013233000000, idx 0/0
Height 2236361, txid <7923a159e3d09e2d26873381a2b88005358dbd63c90f92ca743ffb5a4c490b81>, 0.012340000000, idx 0/0
Height 2236475, txid <6abcf9e78036a0b8116e1a10a62d374ad3a3c25720752077e52424ec7d2f91d5>, 0.002120420000, idx 0/0
NOTE: this transaction uses an encrypted payment ID: consider using subaddresses instead
Refresh done, blocks received: 36962
Untagged accounts:
         Account              Balance      Unlocked balance                Label
 *       0 48PEpD       0.037593420000        0.037593420000       Primary account
-------------------------------------------------------------------------------
         Total        0.037593420000        0.037593420000
Currently selected account: [0] Primary account
Tag: (No tag assigned)
Balance: 0.037593420000, unlocked balance: 0.037593420000 (Some owned outputs have missing key images - import_key_image
s needed)
Background refresh thread started
[wallet 48PEpD]:
```

7. Assuming you have received the user's type the following command: **`import_key_images <KEY_IMAGE_FILE_PATH>`**, where <KEY_IMAGE_FILE_PATH> is the path where you saved the signed key images file the user sent you. In this example, we use *"C:\walletdata\20201122-joe_jones-keyimages.dat"*. Press enter when done:

```
[wallet 48PEpD]: import_key_images C:\walletdata\20201122-joe_jones-keyimages.dat
Height 2236475, txid <6abcf9e78036a0b8116e1a10a62d374ad3a3c25720752077e52424ec7d2f91d5>, spent 0.013233000000, idx 0/0
Signed key images imported to height 2236475, 0.013233000000 spent, 0.024360420000 unspent
[wallet 48PEpD]:
```

8. The CLI wallet will quickly scan and apply the key images file. If everything went properly, you should now have a view-only copy of the user's wallet, allowing you to see the incoming and outgoing transfer information.
9. Issue the command **`balance`** to view the wallet's balance.

```
[wallet 48PEpD]: balance
Currently selected account: [0] Primary account
Tag: (No tag assigned)
Balance: 0.024360420000, unlocked balance: 0.024360420000
[wallet 48PEpD]:
```

NOTE: If you see the text "*(Some owned outputs have missing key images - import_key_images needed)*" after the balance printout, it may mean that:

a. the key image file you imported was missing key images. Specifically, if the user recreated the wallet and used a restore height after the earliest N transactions, there may be missing outgoing transactions in the provided private key image data. You can request that the user rescan the wallet from block height 0 and then re-export the signed key image data; or
b. the user did not specify the "all" keyword after the "export_key_images" command. Not specifying this keyword may lead to some key images not being included in the export file; or
c. the user is purposefully withholding specific key images.

Once you obtain an updated key images export file, you can rerun the **`import_key_images`** command, followed by **`balance`**, to see if the message about missing key images disappears. It is safe to rerun this command as often as necessary (e.g. if it takes numerous tries to get the correct data from the user).

10. Proceed to the Viewing transaction activity section below to view and verify transaction information.

Viewing transaction activity

Type **`show_transfers`** to show all incoming and outgoing transfers to the wallet. A transaction log will be output to the screen (you may want to maximize the CLI wallet window to aid readability):

```
[wallet 48PEpD]: show_transfers
 2236357     in unlocked       2020-11-22 19:08:23       0.009900000000 4e934349bcbd5c10f186d449dfae7d193dee2f881
f3b307bcd322747bbff07df 0000000000000000 0.000000000000 48PEpD:0.009900000000 0 -
 2236360     in unlocked       2020-11-22 19:29:47       0.013233000000 2f456e0b01acfbab2c2bbe24795d1169cf3d4c97e
9085d504d1195599fca7051 0000000000000000 0.000000000000 48PEpD:0.013233000000 0 -
 2236361     in unlocked       2020-11-22 19:33:04       0.012340000000 7923a159e3d09e2d26873381a2b88005358dbd63c
90f92ca743ffb5a4c490b81 0000000000000000 0.000000000000 48PEpD:0.012340000000 0 -
 2236475    out        -       2020-11-22 23:38:12       0.011100000000 6abcf9e78036a0b8116e1a10a62d374ad3a3c2572
0752077e52424ec7d2f91d5 0000000000000000 0.000012580000 - 0 -
[wallet 48PEpD]:
```

Examine the resulting transfer log. The columns in the output are space delimited.

If you are investigating a large number of transactions, you can export the list to a CSV using the command **`export_transfers`**. Both outputs are organized in the following fashion:

For incoming or mined transactions:

1. Block Number
2. "Block" or "in" ("in" for standard incoming transfers, "block" for mined coins)
3. "unlocked" if the coins from the transfer are available for use at the current block
4. Time (YYYY-MM-DD HH:MM:SS in UTC)
5. Amount
6. Transaction Hash

7. Payment ID (or zeros if not available)
8. Fee (shown with outgoing tx, with incoming tx this will be zero)
9. Wallet destinations (in the form of walletID:amount, where walletID is the first 6 characters in the Monero address that received the funds [which may be a subaddress], and the amount is the amount that that address/subaddress received)
10. Subaddress index
11. "-"
12. Note (normally blank)

For outgoing transactions:

1. Block Number
2. "out"
3. “-”
4. Time (YYYY-MM-DD HH:MM:SS in UTC)
5. Amount (excluding change and fee)
6. Transaction Hash
7. Payment ID (or zeros if not available)
8. Fee
9. Destination address(es) (in the form of address:amount)
10. Input addresses (Set of address indices used as inputs in this transfer, e.g. “0” for the wallet’s primary/first address)
11. "-"
12. Note (normally blank)

Examine the amounts of the incoming and outgoing transfers and sum the incoming tx and outgoing transfers.

ABOUT COMPLYFIRST

This report reflects an independent analysis by ComplyFirst and is intended as a tool to help law enforcement, regulators and industry participants understand and evaluate information that may be relevant to AML compliance. ***The report does not reflect a legal conclusion and is no indication of qualitative value of an asset or suitability for investment or any other purpose and is solely for use by ComplyFirst and not for reliance by any other party.***

ComplyFirst's analytical framework is based on relevant federal law, including FinCEN Guidance issued on May 9, 2019 (FIN-2019-G001). Neither the report nor our framework constitute an exhaustive treatment of the legal and regulatory issues relevant to conducting an analysis of AML compliance and ComplyFirst does not analyze other laws or regulations which may apply. The analysis concerning AML compliance may evolve over time as the nature of digital assets, applicable precedent and FinCEN statements and interpretations change and evolve. ComplyFirst's framework has not been endorsed by FinCEN or any other government authority.

The report is based on a limited review of factual information publicly available or otherwise made available to ComplyFirst. Not all potentially relevant factual information has necessarily been reviewed and no independent investigation or analysis, apart from ComplyFirst's own efforts, has been taken to confirm information on which this analysis is based. We do not assume any responsibility for the completeness of the information upon which our analysis and determination is based. It is possible that if additional facts where known or assumed or understood facts prove to be incorrect, the analysis would be materially different.

DOCUMENT HISTORY

Date	Description
11/25/2020	Initial public release
12/7/2020	Minor formatting changes