



I. Feather Wallet with Monerod Isolation Setup (Qubes-Whonix)

1. Objective
2. Setup
 - a. Prerequisites
 - b. Setup user policy in domo
 - c. Port forward to your feather qube
3. Conclusion

Feather Wallet with Monerod Isolation Setup (Qubes-Whonix)

[Bu sayfanin TURKCE cevirisi var!](#)

Objective

In this guide I will show you how to isolate your Feather Wallet in an offline Qubes-Whonix qube, and yet, have it sync with a monero node on another Qubes-Whonix qube on the same QubesOS.

At the end of you following this guide, you will have:

- a feather wallet running in a Qubes-Whonix qube that has no netqube (ie, "offline")

- a feather wallet running in a Qubes-Whonix qube that has a TCP port-forwarding from another Qubes-Whonix qube that runs your monero node
- a feather wallet that can sync its wallet state using a monerod running in another qube

For more information about separating your monero wallet and your monero node inside separate Qubes-Whonix qubes, refer to the following resources:

- <https://forum.qubes-os.org/t/monero-wallet-daemon-isolation-with-qubes-whonix/1121>
- https://www.getmonero.org/resources/user-guides/cli_wallet_daemon_isolation_qubes_whonix.html
- https://www.whonix.org/wiki/Monero_Wallet_Isolation

This guide is highly inspired from these resources. However, keep in mind that the official whonix wiki and getmonero dot org resources are slightly convoluted. Here, we will use a very similar but somewhat simpler way to achieve our goal.

In this guide QubesOS 4.2 (which runs Whonix 17) is used for demonstration.

Setup

Prerequisites

You need to have

1. A monero node, in a Qubes-Whonix qube, running and synced with the Monero blockchain
2. A Feather Wallet, in a Qubes-Whonix qube, downloaded and its digital signatures verified

To prepare these, follow the following guides first:

1. [Basic monero node setup \(Qubes-Whonix\)](#)
2. [Feather Wallet setup \(Linux\)](#)

The second guide is originally written for Lubuntu Linux distribution. However, you can equally follow that guide for setting up your feather wallet in a new Qubes-Whonix qube.

I show how to create a new Qubes-Whonix qube in the first guide I list above. Follow it for creating a Qubes-Whonix qube named **anon-feather-who17**. In the [new Qube creation window](#), insert the followings:

1. **Name and label:** anon-feather-who17.

2. Color: Pick red for color coding your anonymous qubes.

3. Type: Keep it as **AppVM (persistent home, volatile root)** option.

4. Template: Select **whonix-workstation-17** option.

5. Networking: Select **(none)** option.

6. Launch settings after creation: Check this option.

Click **OK**. This will be your offline feather wallet qube.

In order to setup the feather wallet in an offline qube, you should download the feather wallet (and its digital signatures) in a non-offline qube. And then use `qvm-move` command to move the feather wallet files to your offline feather qube.

Once you have the **anon-monerod-who17** and **anon-feather-who17** qubes ready, you can continue with this guide.

Setup user policy in dom0

Open a terminal window in dom0. Enter the following command:

```
$ sudo vim /etc/qubes/policy.d/30-user.policy
```

Add the following line into the file:

```
qubes.ConnectTCP +18081 anon-feather-who17 anon-monerod-who17 allow
```

Here, `qubes.ConnectTCP` is the name of the RPC service we use. This service is located in dom0 `/etc/qubes-rpc/qubes.ConnectTCP` file. The `+18081` is the port argument we pass to `qubes.ConnectTCP` service. Port 18081 is the RPC port for our monero node that local wallets can use to query the blockchain state. This is exactly what we want, from a distant qube apart.

Port forward to your feather qube

Open a terminal window in **anon-feather-who17**. Enter the following command:

```
$ qvm-connect-tcp 18081:anon-monerod-who17:18081
```

From then on, your feather wallet will be able to connect to `127.0.0.1:18081` as a monero node.

Start your feather wallet. File -> Settings -> Network -> Node. **Uncheck** the **Let Feather**

manage this list option and then click on **Add custom node(s)**, insert your node's address as 127.0.0.1:18081. On the **Proxy** tab, select Proxy: None. If you try to add your local node address without first disabling the tor proxy of feather wallet, it will give an error.

After this, you should see your feather wallet quickly syncing its wallet state with the Monero blockchain. In my experience, syncing my wallet this way is much more faster than connecting to a remote node over Tor and syncing it.

A few important points:

1. The port forwarding command we use, `qvm-connect-tcp 18081:anon-monerod-who17:18081`, is active only for the current feather wallet qube session. If you restart the qube, you have to first repeat this command, and then start your feather wallet.
2. In order to update your feather wallet in the future, you should do how you did during initial install: download the updated feather wallet program in another qube and move the downloaded program (and its digital signatures) to your offline **anon-feather-who17** qube.

Conclusion

In this guide I showed you how to isolate your feather wallet and your monero daemon yet get them co-operate and update your monero wallet's state. For this, the prerequisites were a running monerod qube, and a feather wallet installation. You can follow my previous guides that I linked above to prepare them. Once they were ready, we created a policy file in domo authorizing the port forwarding request between our monerod and feather wallet qubes. After that, while the monerod qube is synced, ready, and running in the background, we entered the port forwarding command in the feather wallet qube. From then on, feather wallet is able to sync with the monero's blockchain, even though the qube it's in has no networking qube.

If you find this guide useful you can send some **Monero (XMR)** to this address:

84dAAMSmxmUW2FzE3Bm6wxLNYCYC4DzLkAPfaK7UqsHCWBfEgHnvA4GNgeGMcRDPwdwVvDrtZsTMzc5BV



Donate XMR!

Last edited Fri 11 Oct 2024 21:28:50