The wallet map exists before this work, and it is not left in a drawer as plain text. The map can watch the money. It cannot spend. Anyone who can read it can see the balance, so it is encrypted. The passphrase that opens the file is not a seed, and it is not a plate.

What is required

- The wallet map from the 2-of-3, on the data USB. It is called map.txt only while the work is happening.
- A passphrase that can be remembered and also written down. It is not one of the 24-word lists. It is not the decoy passphrase.
- Tails, with networking set to Disabled. This is not done on the everyday computer.

The steps

01. Tails is booted. Networking is set to Disabled. map.txt is copied to the Desktop. It is not opened on a phone.

02. Terminal is opened. The line below is run. It asks for a passphrase twice. If the two entries differ, it will not make the file.

03. The result is map.txt.gpg. That file is copied to two USB drives. map.txt is deleted from the Desktop and from the USB. The trash is emptied.

04. The passphrase is written on paper. That paper is kept away from the plates and away from the USB drives.

05. On a later offline boot, the file is decrypted once. The map has to open, and the first address has to match what was written down. If a plain copy was made, it is encrypted again. The plain copy is deleted.

```
$ gpg --symmetric --cipher-algo AES256 map.txt
```

Stop if

- The xpubs are not written on paper just in case.
- A letter code or a homemade cipher is not invented. The command above is the one that is used.
- The passphrase is not one of the seeds, and it is not stamped on a plate. A different passphrase is chosen, and the work is started over.

If this passphrase is lost, the coins are not gone. Two plates can rebuild the map. The path is m/48'/0'/0'/2'. Write that path on the inheritance sheet. It is not a secret.

Next. If I am gone. That page is for a spouse. It does not contain words.