# **Vaulting**

# **1 Basics**

A table with an overview of the support status and applicability.

|  |  |
| --- | --- |
| Status: | e.g. **Supported** |
| Architecture(s): |  |
| Component(s): |  |
| Hardware: | *n/a* |

# **2 Overview**

Offers the ability to make it impossible to remove funds from an account until certain conditions are met, such as a specific date passes. It could be set to be recurring also. For example, funds could only be available on the first day of the month. This makes it difficult for hackers who break into the wallet to remove the funds and also prevents extortion/coercion.

# **3 User details**

A user can choose an amount to allocate to their vault. The user can choose how long the allocated funds will be locked up. Periodic vaulting for both withdrawal and deposit can be set up easily to act similarly to allowance or savings account.

# **4 Technical details**

This feature crosses the separation of concerns once again between the wallet development team and the blockchain team. This is because internal transactions within a singular wallet will be necessary to complete this task, but the data needs to be rendered in the GUI.

Theoretically, users would make an internal transaction with themselves setting up a “pocket” or side-wallet that is secured with a smart-contract that releases funds at a specified date.

# **5 Limitations**

Developing a system that integrates smart contracts could be a bit heavier regarding workload than we are scoping on this release. If we can integrate other pre-existing frameworks, we may be able to solve this issue.

# **6 Testing**

Once again, the test network will be integral in testing the functionality of this feature.

# **7 Areas for improvement**

Additional scheduling features could be implemented for specific purposes that we may not be aware of at this time.

# **8 Known issues**

Not applicable at this time.

# **9 References**

Internal transactions - Ethereum.

Smart Contracts - Ethereum / Hyperledger