

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

Service Provider Wallet

The **SKF's** wallet *creates* an **Escrow** to represent a marketplace.

new TimelockedEscrow(timeLimit)

EscrowCreated(escrowAddress, timeLimit)

The **SKF's** wallet *whitelists* a **Service Provider**

whitelist(serviceProviderAddress)

ServiceProviderWhitelisted(serviceProviderAddress)

The **SKF's** wallet *unwhitelists* a **Service Provider**

unwhitelist(serviceProviderAddress)

ServiceProviderUnwhitelisted(serviceProviderAddress)

Legend

- Wallet to Contract function call
- Inter-Contract function call
- Transfer of KEY
- Event emitted by contract

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

Service Provider Wallet

The **Identity Owner's** wallet checks an address to see if it's been whitelisted.

← *isWhitelisted(address) returns bool* →

Legend

○

Wallet to Contract function call

○

Inter-Contract function call

○

Transfer of KEY

◊

Event emitted by contract

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.

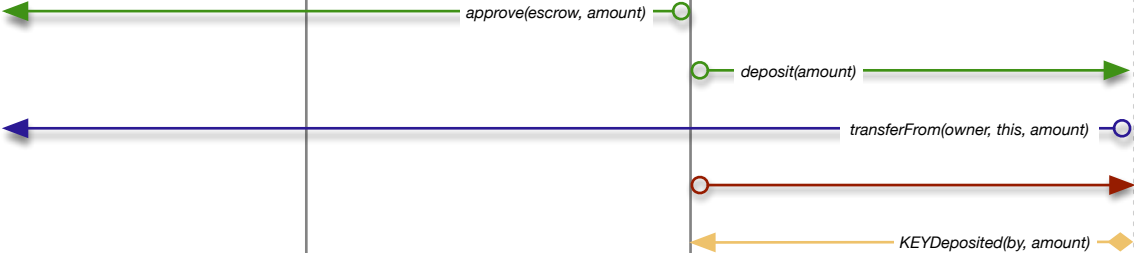


Timelocked Escrow Contract

Service Provider Wallet

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

The **Identity Owner's** wallet *approves* the **Escrow** to transfer an amount of **KEY** from the owner to itself.



Legend

- Wallet to Contract function call
- Inter-Contract function call
- Transfer of KEY
- Event emitted by contract

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

Service Provider Wallet

The **Service Provider's** wallet asks the **Escrow** if an address has sufficient funds on deposit.

← hasFunds(address, amount) returns bool →

Legend

○→

Wallet to Contract function call

○→

Inter-Contract function call

○→

Transfer of KEY

◊→

Event emitted by contract



SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

Service Provider Wallet

The **Service Provider's** wallet asks the **Escrow** if an address's deposit is still timelocked.

← areFundsTimelocked(address) returns bool →

Legend

○

Wallet to Contract function call

○

Inter-Contract function call

○

Transfer of KEY

◊

Event emitted by contract

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

Service Provider Wallet





The **Identity Owner's** wallet approves the **Escrow** to transfer funds to the **Service Provider's** address.



The **Identity Owner's** wallet *transfers* the approved funds to the **Service Provider's** address. This is only allowed if either the address has previously been whitelisted, or if the time lock has expired.



Legend

-  Wallet to Contract function call
-  Inter-Contract function call
-  Transfer of KEY
-  Event emitted by contract

SelfKey Foundation's CrowdSale Contract owns the **SelfKey Token** contract



SelfKey Token

The **SelfKey Foundation's** wallet creates **Escrow** contracts and adds and removes **Service Providers'** addresses from the **Escrow's** whitelist.

SKF Wallet

An **Identity Owner** is a party that wishes access to a service provided by a Service Provider. They will deposit some **KEY** into a **Timelocked Escrow**.

Identity Owner Wallet

The **Escrow** holds **KEY** deposited by the **Identity Owner** that can not be transferred to any address that is not in the **Escrow's** whitelist until the timelock has expired. It is owned by SKF.



Timelocked Escrow Contract

Service Provider Wallet

A **Service Provider** is a party, whitelisted by an **Escrow**, that wishes to provide a service to an **Identity Owner** who has deposited funds into that **Escrow**.

The **Identity Owner's** wallet *approves* the **Escrow** to transfer funds to itself.

approve(escrow, amount)

The **Identity Owner's** wallet *retrieves* their funds from the **Escrow** to itself. This is only allowed if the time lock period has expired.

retrieve()

transferFrom(this, identityOwnerAddress, amount)

Legend

- Wallet to Contract function call
- Inter-Contract function call
- Transfer of KEY
- Event emitted by contract