Hi everyone

,

We’re excited to announce that the contract upgrade feature is now fully developed and tested. This feature will allow developers to upgrade their secret contracts without having to create a new contract and have users manually migrating their state. This can be useful for fixing bugs, adding new features, or updating the contract’s security.

The contract upgrade feature saves time and effort for developers, and increases the velocity and flexibility with which developers on Secret Network can build and deploy their contracts. This is because they no longer need to create a new contract and manually migrate user state, which can be a risky and error-prone process. The contract upgrade feature is a valuable tool for developers who want to keep their contracts up-to-date.

To upgrade a contract, the developer will need to specify an admin address when the contract is created. The admin address will be able to initiate the upgrade process by submitting a MsgMigrateContract

transaction. The transaction will specify the new contract code ID, and will invoke the migrate()

function on the new contract’s code. The new code can optionally perform state migrations inside the migrate()

function.

The upgrade process is as follows:

1. The new contract code is deployed to the network.

2. A MsgMigrateContract

transaction is submitted to the network, specifying the contract address, the new code ID and a message to be sent to the migrate()

function on the new code.

1. The network validates the message and checks that the admin address is authorized to upgrade the contract.

2. The state of the old contract is migrated to the new contract while keeping the same contract address.

The contract upgrade feature is backward compatible. This means that contracts that were created before the v1.10 upgrade can still be upgraded to the new version. However, the admins of these contracts will need to be hard-coded into the v1.10 upgrade proposal.

We’re currently in the process of collecting a list of contracts that developers want to upgrade. If you have a contract that you want to upgrade, please submit the contract address and the admin address in this form Upgradeable contracts.

We’ll be providing more details about the contract upgrade feature in the coming weeks. In the meantime, please feel free to ask any questions you have in the comments below.

Q&A:

- Q: What are the security implications of the contract upgrade feature on Secret Network?

- A:

The contract upgrade feature introduces some security risks, as the admin of a contract can upgrade it to have malicious code that can steal funds and private data. This is the same risk that exists on public blockchains, but it is unique to Secret Network in that the malicious code can also steal private data that is stored inside the contract.

To mitigate these risks, it is important to carefully choose the admin of a contract and to monitor the contract for any suspicious MsgMigrateContract

transactions.

Here are some additional steps that can be taken to mitigate the security risks of the contract upgrade feature: * Use a multisig wallet to control the admin keys.

- Require a super majority vote to approve a contract upgrade (DAOs / multisig).

- Implement a public code review process for all contract upgrades.

- Provide a way for users to reproduce the compiled WASM binary from source. This can be done by using the Docker contract optimizer image for compilation and specifying the exact git commit from which the contract was built.

- Use a multisig wallet to control the admin keys.

- Require a super majority vote to approve a contract upgrade (DAOs / multisig).
- Implement a public code review process for all contract upgrades.
- Provide a way for users to reproduce the compiled WASM binary from source. This can be done by using the Docker contract optimizer image for compilation and specifying the exact git commit from which the contract was built.
- Q: What are the benefits of upgrading a contract?
- A:

There are several benefits to upgrading a contract, including: * Fixing bugs

- Adding new features
- Updating the contract’s security
- Making the contract compatible with new network features
- Fixing bugs
- Adding new features
- Updating the contract’s security
- Making the contract compatible with new network features
- Q: How do I know if my contract needs to be upgraded?
- A:

There are various scenarios where you’d want to upgrade an old contract: * Older SNIP-20 tokens that: * Don’t support query permits

- Have the slower permits implementation (before the v1.3 upgrade on May 2022)
- Don’t record timestamps for transfers
- Don’t have the recent decoys feature
- Want to add MetaMask permit support in the future
- Don’t support query permits
- Have the slower permits implementation (before the v1.3 upgrade on May 2022)
- Don’t record timestamps for transfers
- Don’t have the recent decoys feature
- Want to add MetaMask permit support in the future
- Older SNIP-721 tokens
- Fixing bugs in highly used contracts
- Highly used contracts that already have state and that you can’t or don’t want to ask users to manually migrate their state
- Older SNIP-20 tokens that:
- Don’t support query permits
- Have the slower permits implementation (before the v1.3 upgrade on May 2022)
- Don’t record timestamps for transfers
- Don’t have the recent decoys feature
- Want to add MetaMask permit support in the future
- Don’t support query permits
- Have the slower permits implementation (before the v1.3 upgrade on May 2022)

- Don’t record timestamps for transfers
- Don’t have the recent decoys feature
- Want to add MetaMask permit support in the future
- Older SNIP-721 tokens
- Fixing bugs in highly used contracts
- Highly used contracts that already have state and that you can’t or don’t want to ask users to manually migrate their state
- Q: Would the contract code hash change after a contract upgrade?
- A:

Yes, the contract code hash will change after a contract upgrade. This means that any contracts that interacts with the upgraded contract will also need to be updated to use the new code hash. You may also need to update UIs that have the code hash hard-coded.

- Q: What are the security implications of upgrading a contract?
- A:

There are some security implications to upgrading a contract, such as the risk of introducing new bugs or vulnerabilities. It’s important to carefully review the upgrade process before proceeding. However, you can always perform an upgrade to the old code, provided that the new code didn’t make irreversible state migrations.

- Q: Can CosmWasm v0.10 contracts be upgraded as well?
- A:

Yes. Any contract can be upgraded to CosmWasm v0.10 or v1.

- Q: Can a contract be the admin of another contract?
- A:

Yes.

- Q: Can a contract be the admin of itself?
- A:

Yes.

- Q: Can a contract be upgraded once and then disable future upgrades?
- A:

For hard-coded admins via governance, only via a future governance proposal

. For contracts that were created with an admin, yes

, by sending a MsgUpdateAdmin

or MsgClearAdmin

transaction.

We hope this helps! Please feel free to ask any other questions you have in the comments below.

Onwards and upwards

,

The SCRT Labs Team