Overview

Tessera uses private and public keys pairs to provide transaction privacy. You can use existing key pairs or use Tessera to generate new key pairs .

You can configure Tessera to use one or more keys. Configure access to the keys by specifying keys in the Tessera configuration file .

!!! Example "Keys configuration"

"keys": { "passwordFile": "Path", "keyVaultConfigs": [{ "keyVaultType": "Enumeration: AZURE, HASHICORP, AWS", "properties": "Map[string]string" }], "keyData": [{ // The data for a private/public key pair }] } Configure the keyData object to access the key pair using any of the following methods:

Configuration method Description Direct Provide the key pair data in plain text. Inline Provide the key pair data in plain text with the private key in a config JSON object. File-based Provide the location of the key pair files. Azure Key Vault Provide the location of the keys in the configured Azure Key Vault . AWS Secrets Manager Provide the location of the keys in the configured AWS Secrets Manager . HashiCorp Vault Provide the location of the keys in the configured HashiCorp Vault . If using a vault to store your keys, use the keyVaultConfigs object to configure the details to access the vault.

Use multiple keys

You can configure multiple key pairs for a Tessera node. In this case, any one of the public keys can be used to address a private transaction to that node. Tessera tries each key to find one that can decrypt the payload.

!!! note

Multiple key pairs can only be configured within the configuration file.

View the keys registered for a node

You can use the ThirdParty API /keys endpoint to view the public keys of your Tessera node.

!!! example "/keys request"

request: :/keys { "keys" : [{ "key" : "oNspPPgszVUFw0qmGFfWwh1uxVUXgvBxleXORHj07g8=" }, { "key" : "ABn6zhBth2qpdrJXp98IvjExV212ALl3j4U//nj4FAI=" }] } You must configure the corresponding server .

Provide key passwords at runtime

Tessera displays a CLI prompt if it has incomplete password data for its locked keys . You can use this prompt to provide the required passwords for each key instead of providing them in the configuration file itself.

!!! example "CLI password prompt"

tessera -configfile path/to/config.json Password for key[0] missing or invalid. Attempt 1 of 2. Enter a password for the key

2019-12-09 13:48:16.159 [main] INFO c.q.t.config.keys.KeyEncryptorImpl - Decrypting private key 2019-12-09 13:48:19.364 [main] INFO c.q.t.config.keys.KeyEncryptorImpl - Decrypted private key

Tessera startup continues as normal

Update a configuration file with new keys

If you generate new keys , you can update the Tessera configuration file manually.

However, you can use the tessera keygen -configfile option to automatically update a configuration file. This is particularly useful for scripting. For example:

tessera -keygen -filename key1 -configfile /path/to/config.json --configout /path/to/new.json --pwdout /path/to/new.pwds The command prompts for a password and generates the key1 pair. The Tessera configuration /path/to/config.json is updated and saved to /path/to/new.json .

New passwords are appended to the existing password file defined in /path/to/config.json and written to /path/to/new.pwds .

If the --configout and --pwdout options are not provided, the updated JSON configuration prints to the terminal. Edit this page

Last updatedonOct 9, 2023 bydependabot[bot] Previous Keys Next Direct keys