

# Multisignature accounts on Namada

Multisignature accounts (multisigs) are accounts on Namada that allow for multiple signers. There are many benefits of having multisigs, including but not limited to

- Increased security
- Ability to share wallets
- Better recovery options

For this reason, all accounts on Namada are multisignature accounts by default.

## Initialising a multisignature account

Before creating an account, a user must generate at least one cryptographic key, that will be used to sign transactions.

The following method will generate such a key:

```
namadaw
```

```
gen \ --alias my-key1 A second key can be generated as well (which will be useful for multisigs):
```

```
namadaw
```

```
gen \ --alias my-key2 Accounts on Namada are initialized through the following method:
```

Non-multisig account (single signer)

```
namadac
```

```
init-account \ --alias my-multisig-alias \ --public-keys my-key1 \ --signing-keys my-key1 Multisig account (at least 2 signers)
```

```
namadac
```

```
init-account \ --alias my-multisig-alias \ --public-keys my-key1,my-key2 \ --signing-keys my-key1,my-key2 \ --threshold 2
```

## Submitting a multisignature transaction

In order to submit a multisignature transaction, an offline transaction must first be constructed.

### Constructing an offline transaction

As of Forv0.23.0 there are certain limitations to the offline transaction construction. Please be wary of any bugs that may arise. The `--dump-tx` argument allows a user to do this. A folder is required to be specified where the transaction will be dumped.

```
mkdir
```

```
tx_dumps This can be done through the following method:
```

```
namadac
```

```
transfer \ --source my-multisig-alias \ --target some-established-account-alias \ --token NAM \ --amount 100 \ --signing-keys my-key1 \ --dump-tx \ --output-folder-path tx_dumps What this means is that the transaction has been constructed, and is ready to be signed.
```

Within the specified folder, a `tx` file will be created. This file contains the hexadecimal representation of the transaction bytes. This file can be used to sign the transaction.

### Signing the transaction

The next step is to sign the transaction. `my-key1` can sign the transaction through the following method:

```
namadac
```

```
sign-tx \ --tx-path "" \ --signing-keys my-key1 \ --owner my-multisig-alias Note that any number of --signing-keys could sign the transaction in this stage. This will produce multiple signatures, which can be used to submit the transaction. Which means that the signature has been saved to this file (located in the current directory).
```

Let's save this as an alias:

```
export SIGNATURE_ONE =  
"offline_signature_FB7246E3FC43F59D8AEEC234EBFDB9DF1AC9BB7B14E536D05A7E2617CA41D4CD_0.tx" Ensure to  
sign the transaction with at least k-of-n keys, where k is the minimum number of signatures required to submit a transaction,  
and n is the total number of keys. In this example, k=2 and n=2.
```

Then let's say this signing produces another signature which we save to the alias SIGNATURE\_TWO .

## Submitting the transaction

The final step is to submit the transaction. This can be done through the following method:

```
namadac
```

```
tx \ --tx-path "tx_dumps/a45ef98a817290d6fc0efbd480bf66647ea8061aee1628ce09b4af4f4eed1c2.tx" \ --signatures  
SIGNATURE_ONE SIGNATURE_TWO \ --owner my-multisig-alias \ --gas-payer my-key1 Note the lack of commas used in  
the--signatures argument. This is because the argument is a list of files, not a list of signatures. Also note the tx_dumps  
folder. This is the folder where the transaction was dumped to as specified in--output-folder-path in the previous step.
```

## Changing the multisig threshold

It is possible to change the multisig threshold of an account. This can be done through the following method:

```
namadac
```

```
update-account \ --address my-multisig-address \ --threshold 1 \ --signing-keys my-key1,my-key2 One can check that the  
threshold has been updated correctly by running:
```

```
namadac
```

```
query-account \ --owner my-multisig-address Which will yield the threshold of 1, together with the two public keys.
```

## Changing the public keys of a multisig account

It is possible to change the public keys of a multisig account. This can be done through the following method:

```
namadac
```

```
update-account \ --address my-multisig-address \ --public-keys my-key3,my-key4,my-key5 \ --signing-keys my-key1,my-key2  
Which will change the public keys of the multisig account from my-key1 and my-key2 to the keys my-key3 ,my-key4 and my-  
key5 (assuming they exist in the wallet).
```

The public-keys provided to the argument--public-keys will become the new signers of the multisig. The list must be a list of public keys, separated by commas, and without spaces. There must be at least 1 public key in the list, and the length of list must be at least the threshold of the multisig account.

## A video tutorial

Skip all the boring reading and watch a video tutorial instead:

[Transparent accounts Send and Receive Nam Tokens](#)