

Step1 : Connecting to Indy Nodes Pool

Steward

Ledger



Create pool config

After pool configuration is created we can connect to the nodes pool that this configuration describes

Function :

```
pool.create_pool_ledger_config(pool_['name'], pool_['config'])
```

Open pool ledger

Function :

```
pool_['handle'] = await pool.open_pool_ledger(pool_['name'], None)
```

Pool Handle is returned , which is used to reference this opened connection in future libindy calls.

Step 2 : Getting Ownership of Steward's Verinym

Steward's Agent

Wallet



Create wallet
Function :

```
wallet.create_wallet(steward['wallet_config'], steward['wallet_credentials'])
```

Open wallet

```
wallet.open_wallet(steward['wallet_config'], steward['wallet_credentials'])
```

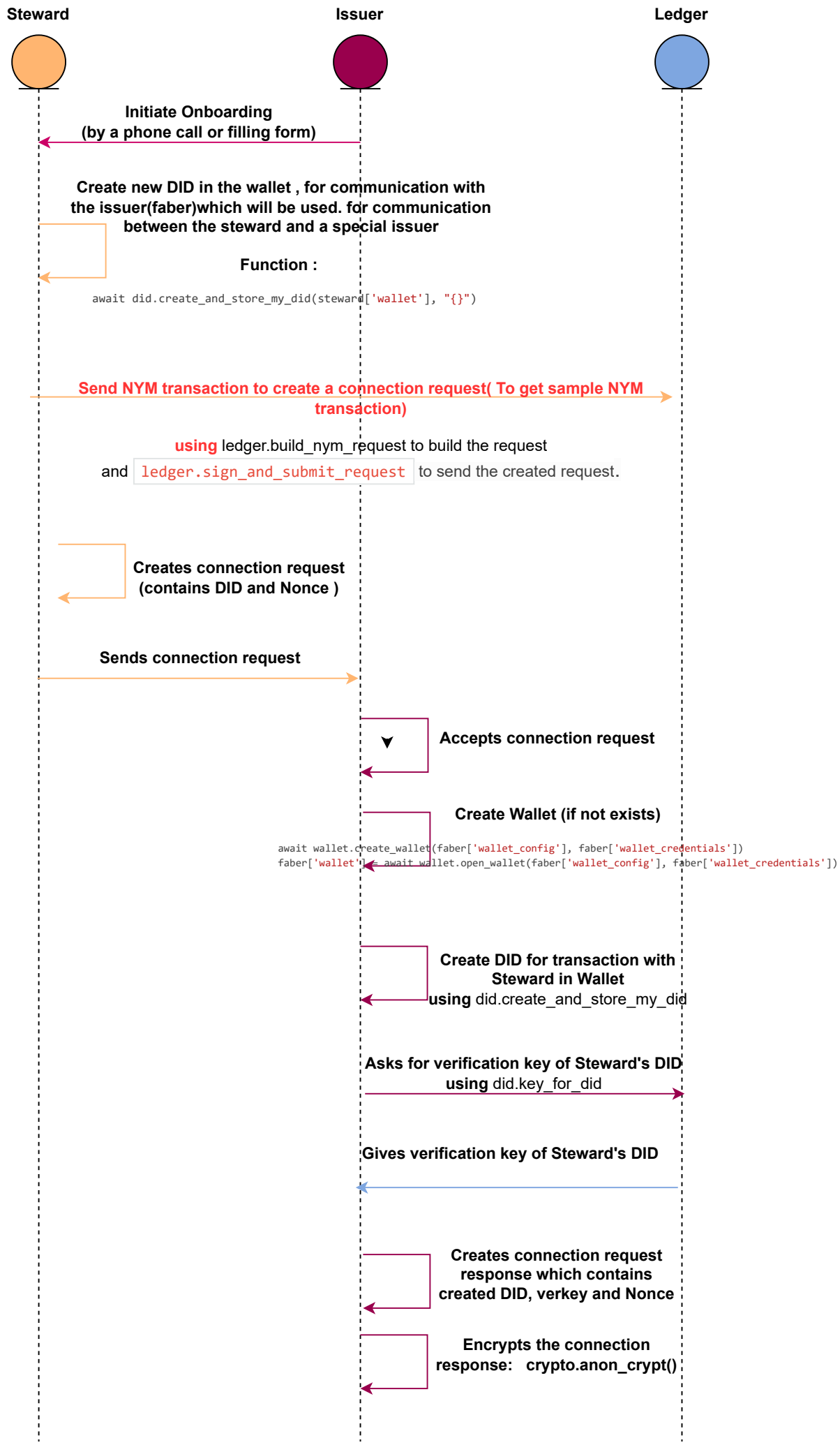
Wallet handle is returned by calling the function for open wallet , which is used to
reference this opened wallet for future calls

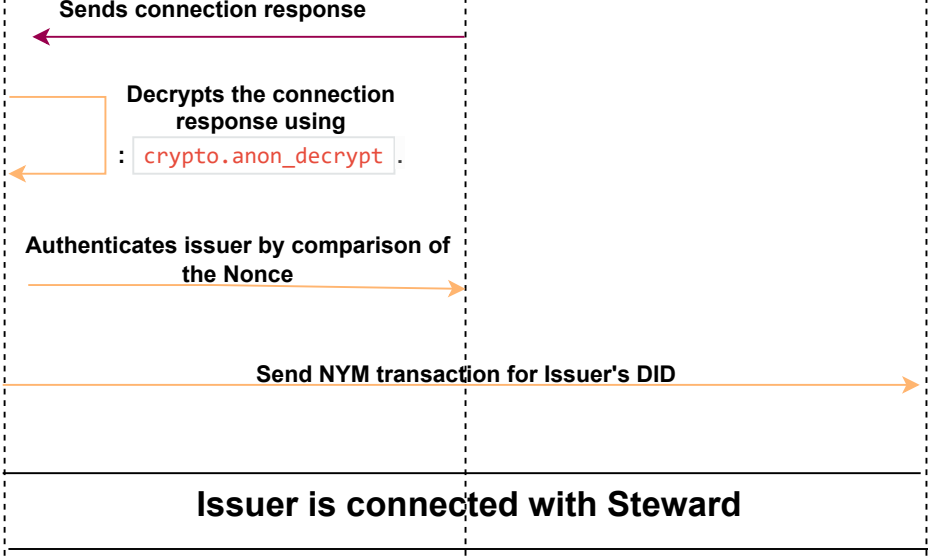
Create and store my DID

```
steward['did'], steward['key'] = await did.create_and_store_my_did(steward['wallet'], steward['did_info'])
```

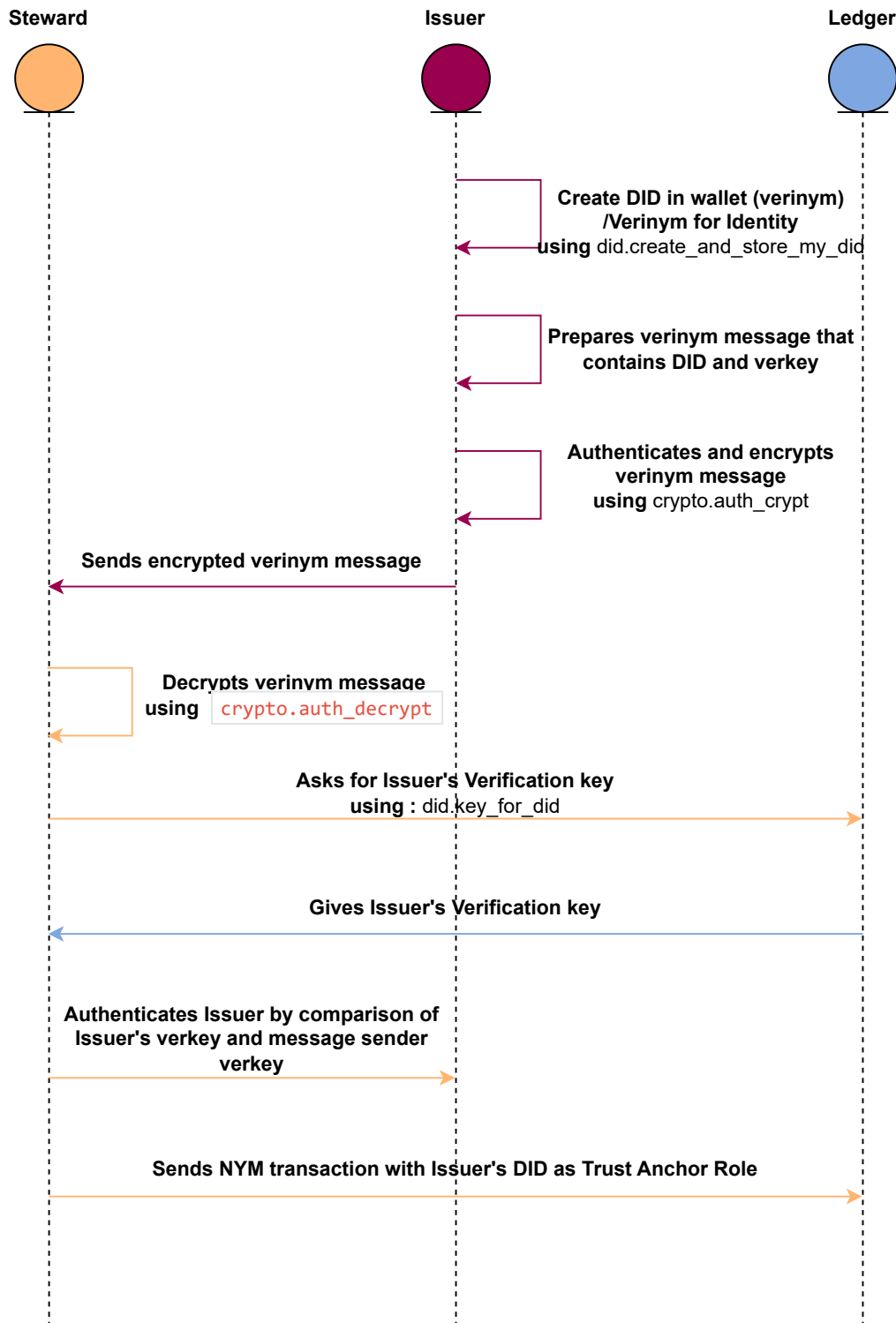
Store DID's signkey, verkey and
metadata

Step 3 : Onboarding of Steward and Issuer

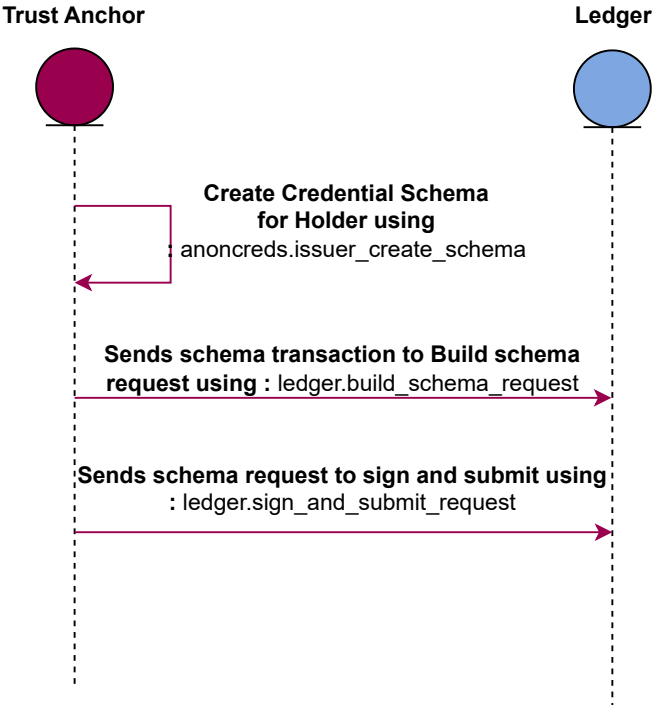




Step 4 : Getting Verinym and Trust Anchor Role for



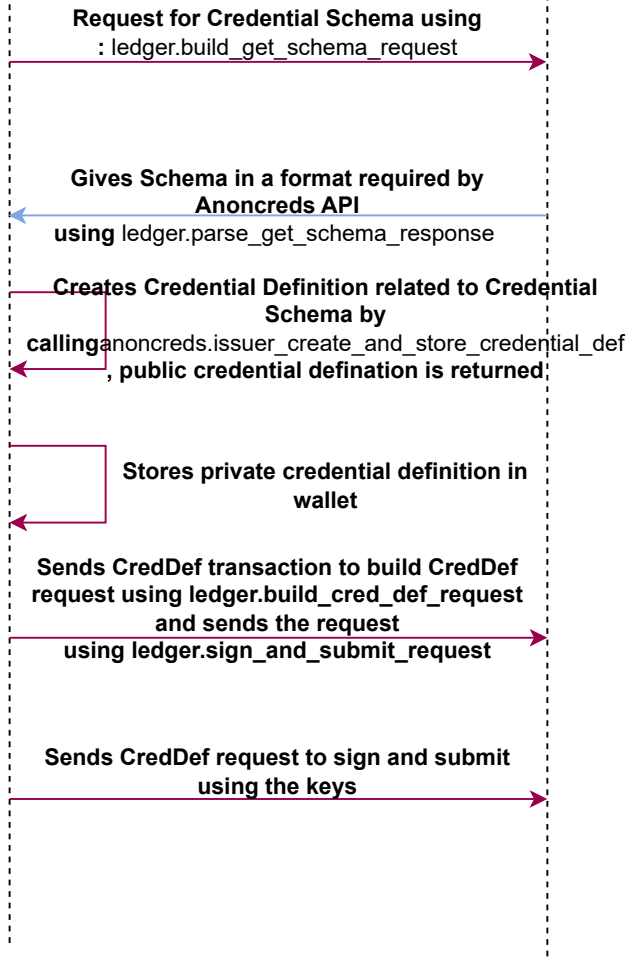
Step 5: Creating Credential Schema



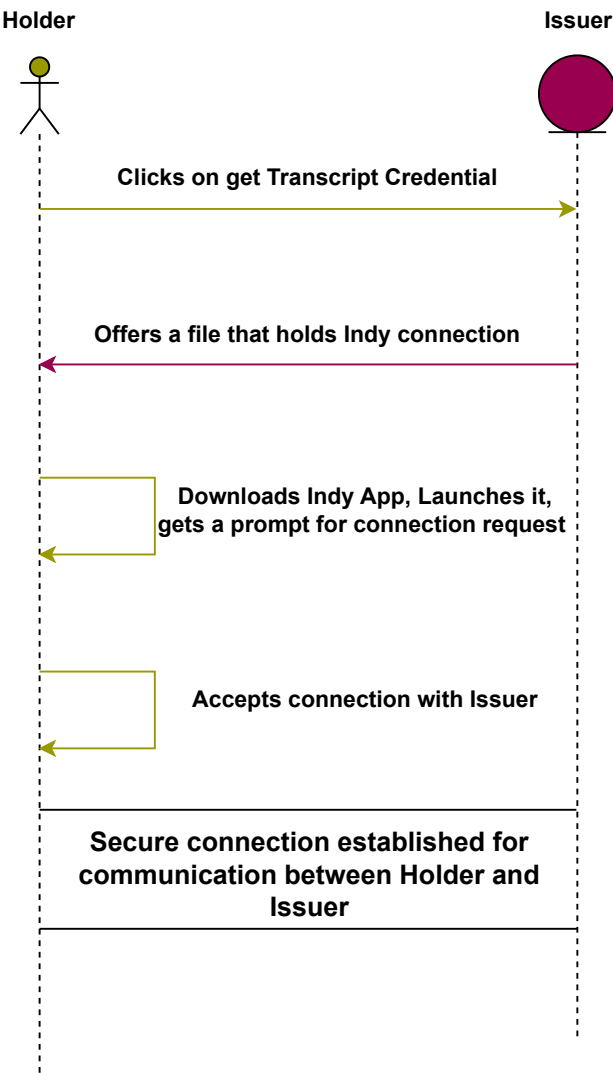
Step 6: Creating Credential Definition

Trust Anchor

Ledger

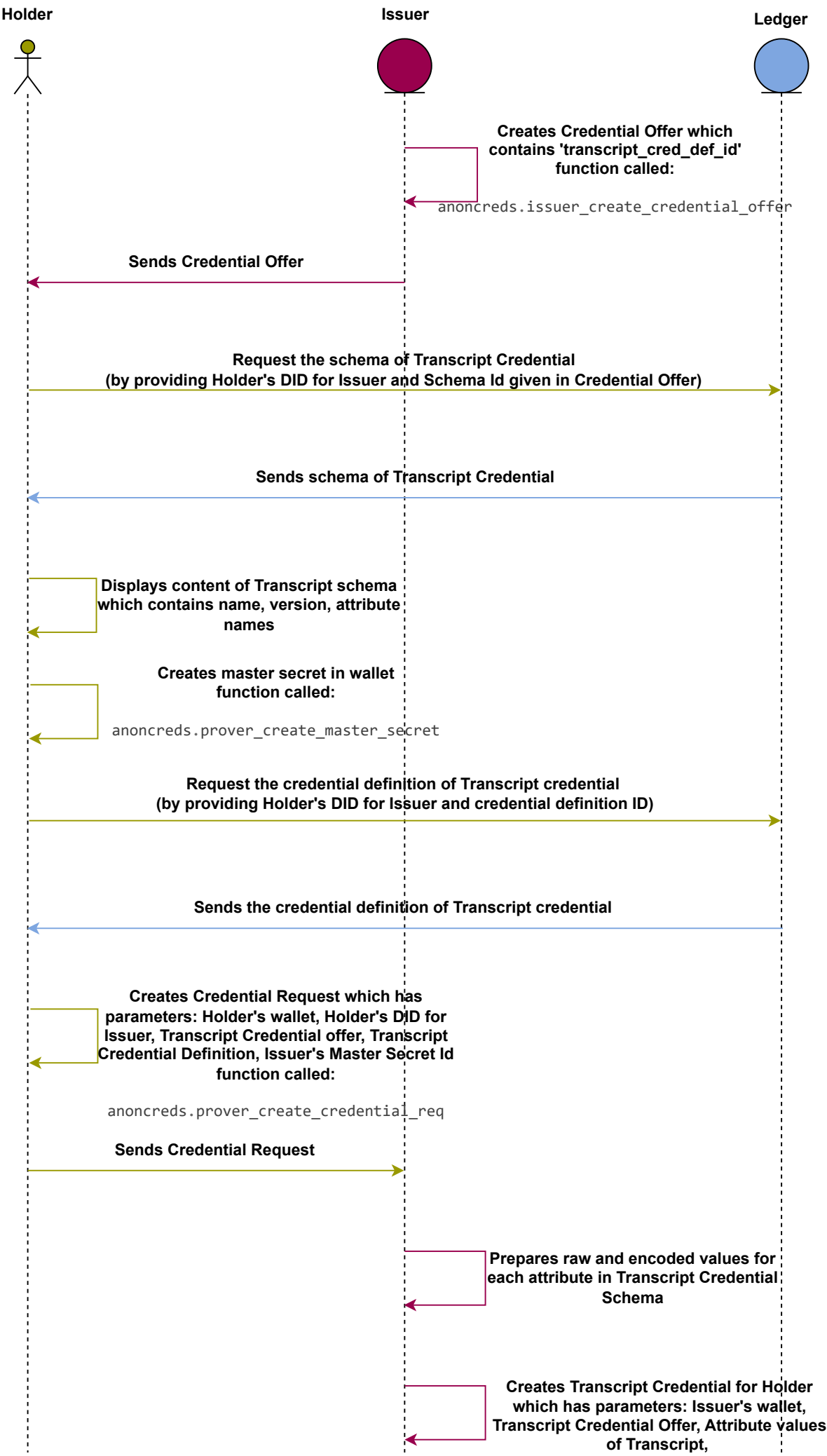


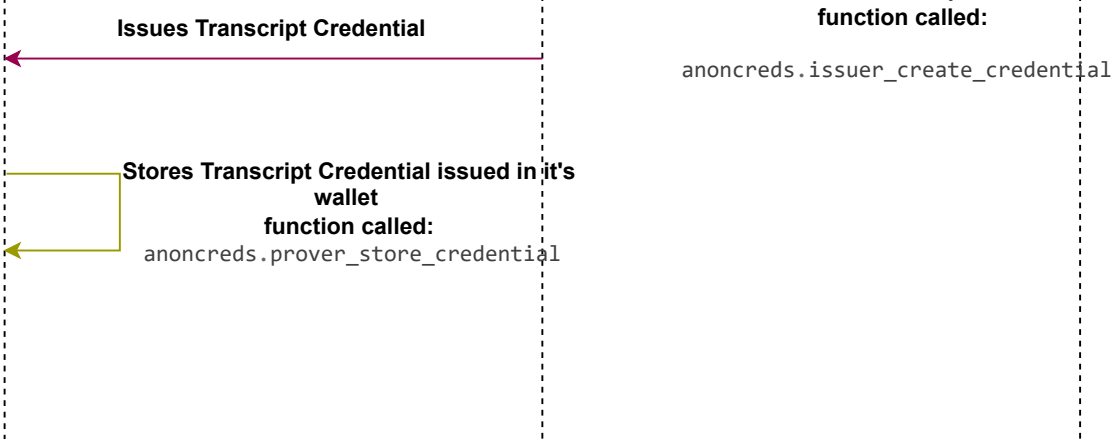
Connecting Holder with Issuer



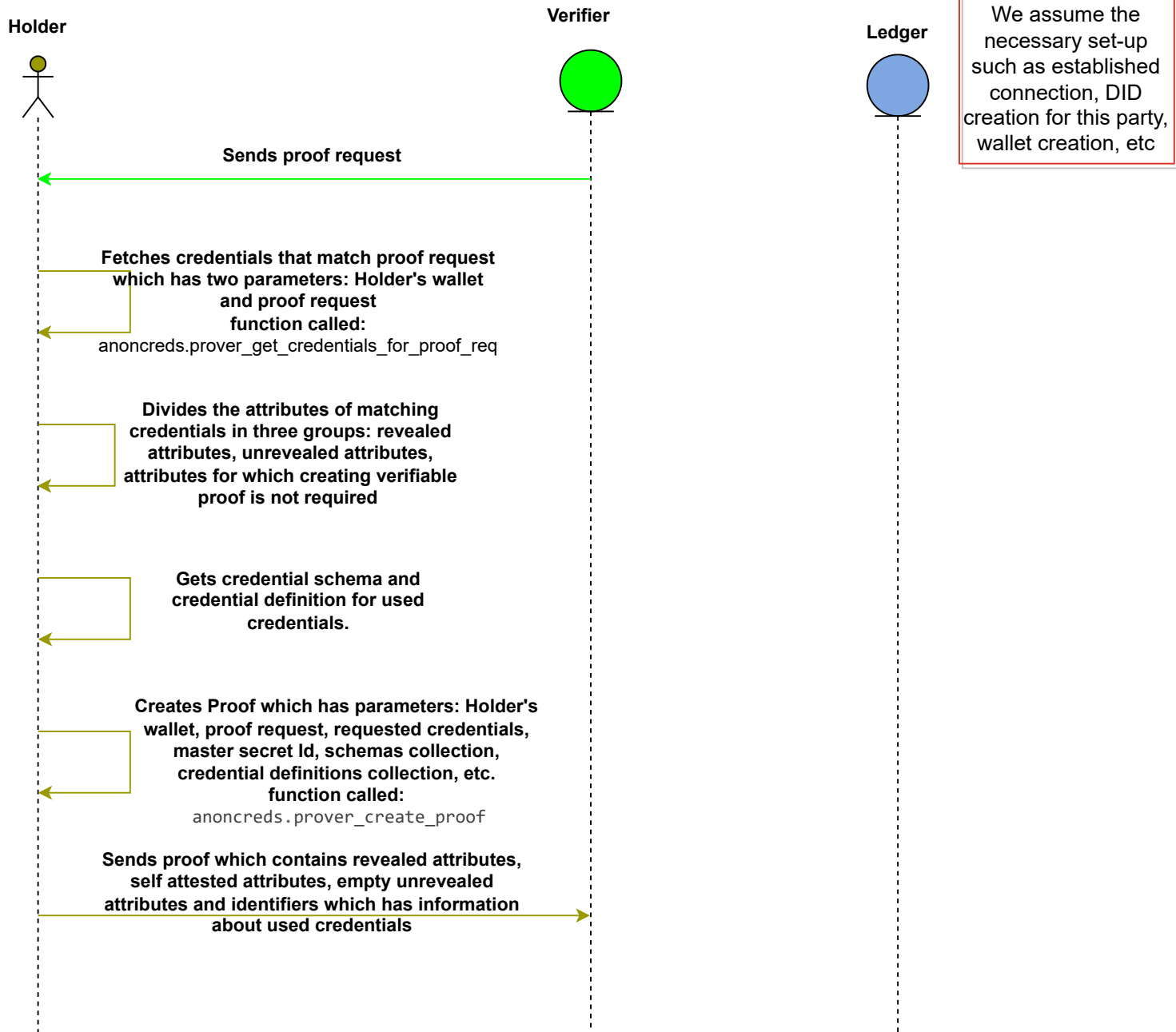
We assume the necessary set-up such as DID creation for holder, wallet creation, etc

Credential Issuance





Holder Generating Proof



Verifier Checking Validity of Proof

