13th October 2023

CS61065 Theory and Applications of Blockchain

Assignment 3: Hyperledger Indy

Date of Submission: November 02, 2023 EOD

You need to submit the assignment as a group of maximum 4 members. Only one member of each group should submit the assignment. Clearly mention your group details in the submission.

Rajesh is a graduate of IIT Kharagpur. He always wanted to be a commercial pilot. Now, he has been selected by the National Aviation Academy (NAA). Now he wants to apply for an educational loan from CBDC Bank. The bank requires the proof of an asset, and proof of admission to a reputed institute as evidence for the creditworthiness of the loan.

In this assignment you need to implement a verifiable credential and verifiable presentation the flow using Hyperledger Indy, where there are 4 parties:

- NAA
- Rajesh
- CBDC Bank
- Government

Assume Government and NAA will be issuing the credentials for proof of property, and student certificate respectively to Rajesh. Rajesh will present the credentials to CBDC Bank that will validate his claims.

Submission Instructions

Create a directory and name it as A3_Indy_(#ROLLNUMBERs separated by underscore). You need to create a single file which will execute the entire flow, and place it inside this directory. If you are implementing it in python, then name the file as **indyassignment.py**. Similarly for nodejs, name it as **indyassignment.js**, and so on. In the first line of the file, write your roll number as a comment. Compress the folder as a zip (with .zip extension). Upload the compressed file in moodle. Make sure you DO NOT include node_modules or similar library dependency files in your zip.

Part A **(15 Marks)**

Launch Indy pool by starting the docker image mailtisen/indy_pool:latest

You may also choose to run it from the indy_pool repository.

2. Connect to the indy pool.

3. Configure one steward.

4. Register Verinyms for Trust Anchors - Government and NAA

Part B **(15 Marks)**

Setup the credential schemas and credential definitions for PropertyDetails, and BonafideStudent. The Government creates both the schemas in the indy ledger. NAA registers a credential definition for BonafideStudent, and the Government registers a credential definition for PropertyDetails.

The schema for PropertyDetails and BonafideStudent are as follows:

```
{
'name': 'PropertyDetails',
'version': '1.2',
'attributes': ['owner_first_name', 'owner_last_name',
'address_of_property', 'residing_since_year', 'property_value_estimate',
realtion_to_applicant]
}

 {
'name': 'BonafideStudent',
'version': '1.2',
'attributes': ['student_first_name', 'student_last_name',
'degree_name', 'student_since_year', 'cgpa']
}
```

Part C **(10 Marks)**

Once the schema and credential definition setup is done, the issuers issue credentials to Rajesh

1. Government issues 'PropertyDetails' credential.
2. NAA issues 'BonafideStudent' credential.
3. Rajesh saves both credentials to his wallet.

Use the following claims to create the verifiable credentials:

First Name: “Rajesh”

Last Name: “Kumar”

Address of Property: “Malancha Road, Kharagpur”

Estimated Value of Property: 2000000

Owner Since: 2010

Degree Name: “Pilot Training Programme”

Student Since: 2022

CGPA: 8

Part D **(10 Marks)**

CBDC Bank requests a “loan_application_proof_request”, where the proofs for the following are

required:

- first_name
- last_name
- **degree_name**
- **student_since_year [>=2019 and <= 2023]**
- **cgpa [> 6]**
- **address_of_property**
- **property_value_estimate [> 800000]**
- **residing_since_year**

The claims in **red** must be from a credential issued by NAA.

Claims in **blue** must be from a credential issued by the Government.

For student_since_year, cgpa, and property_value, the values are not requested, instead the zero knowledge proof is requested to validate them (use 'requested_predicates' for it).