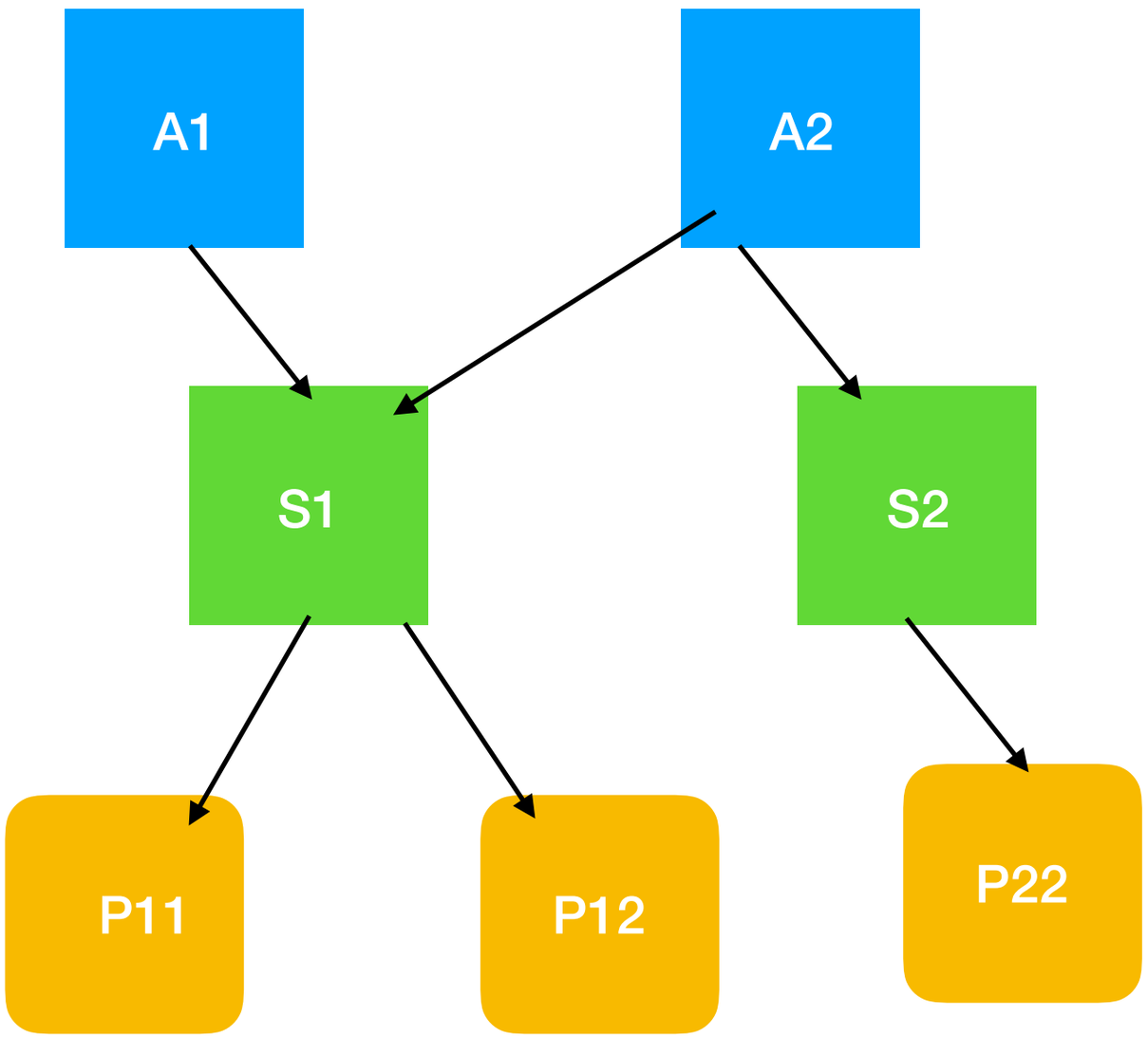
Assignment B
Identity Management

- Pulak Chakraborty

Use Case - Supply Chain

- For the sake of constructing a use case, we have assumed that the Ethereum based private blockchain is used by the companies involved an Electrical Manufacturing Supply chain

- There are assemblers and suppliers, who supply products



Use Case - Supply Chain

- In the previous slide, we have diagram of a over simplified supply chain where each supplier has some products, which might be of interest from an assembler's perspective.

- In order to keep it simple, we have omitted multiple other parties (sub-suppliers, auditors etc.)

Identity and Access Management - Considerations

- assumption - the involved companies are using private blockchain
- considerations:

1. payment after delivery
2. only registered users/companies will be able to participate in buy/sell
3. trading amounts should be hidden from suppliers so that they can’t see total market size or involvement of other suppliers
4. system needs to have different types of access rights (create/view etc.) depending on the functionality of the system

Identity and Access Management

- the points mentioned in the previous slide makes it a very important factor for the need of a proper Identity and access management solution for the decentralized application in our platform.

- For that purpose, a very simplistic solution would be to implement functionalities which involve 1. define the identities of the system users and 2. defining their access rights

- For the users of the system, there should be an infrastructure to facilitate CRUD operations

- Users should be categorized based on how they are going to access the system and according to that, they should be given certain access rights, thus authorizing them on how to use the system

- A super simplistic demo code has been created to demonstrate the purpose

On the Demo Code - Write up

- UserManagement.sol - in this solidity contract we create the blueprint of the CRUD operations. please note that only creation and deletion of users/nodes has been implemented; comments are included in the code to facilitate the ease of understanding

- User.sol - in this code, the permissions for certain users are created and simple functions were created to demonstrate how those access rights come into play

  - Records can be thought of as products supplied by a seller/supplier; the structure is kept super simple

  - hasPermission1 is one such right which enables a party to add a record in the system

  - access rights for a particular user can be applied/revoked by the contract owner

Going further

- as mentioned before, the code and the use case is kept super simple to demonstrate the sample Identity Management scenario.

- a few things which can be further implemented are as follows:

- instead of storing raw record data not he blockchain, data could be hashed and then stored; update, delete, retrieve functionalities on the records can be added and respective access rights could be created

- identity systems such as Uport could be used verify digital identities on the blockchain