Raspberry Pi Indy agent
August, 2019

Raspberry Pi Indy agent

- **Introduction**
  - **Name**: Zeng Zixuan
  - **Location**: Hangzhou, China
  - **University**: Zhejiang University
  - **Mentor(s):** Adam Burdett
  - **Hyperledger project**: Hyperledger Indy

Raspberry Pi Indy agent

- **Project Description**:
- This project intends to develop an Indy agent running on Raspberry pi, producing a customized Raspbian image which have easy access to GPIO pins, enabling it to interact with external sensors, LED matrix, etc. With the new Hyperledger Aries project, our implementation is based on Aries-cloud-agent (previously indy-catalyst), that can interact with Indy pool and can create more interesting applications. This project also includes an Aries RFC defining the message format for interactions with SenseHat extension board as well as its messaging module implementation.

Raspberry Pi Indy agent

- **Project Objectives**: Obj 1: Build an Indy agent running on raspberry pi that can interact with other agents and have easy access to GPIO pins.
  - Obj 2: Produce a Raspbian image containing a generalized Indy Agent.

Raspberry Pi Indy agent

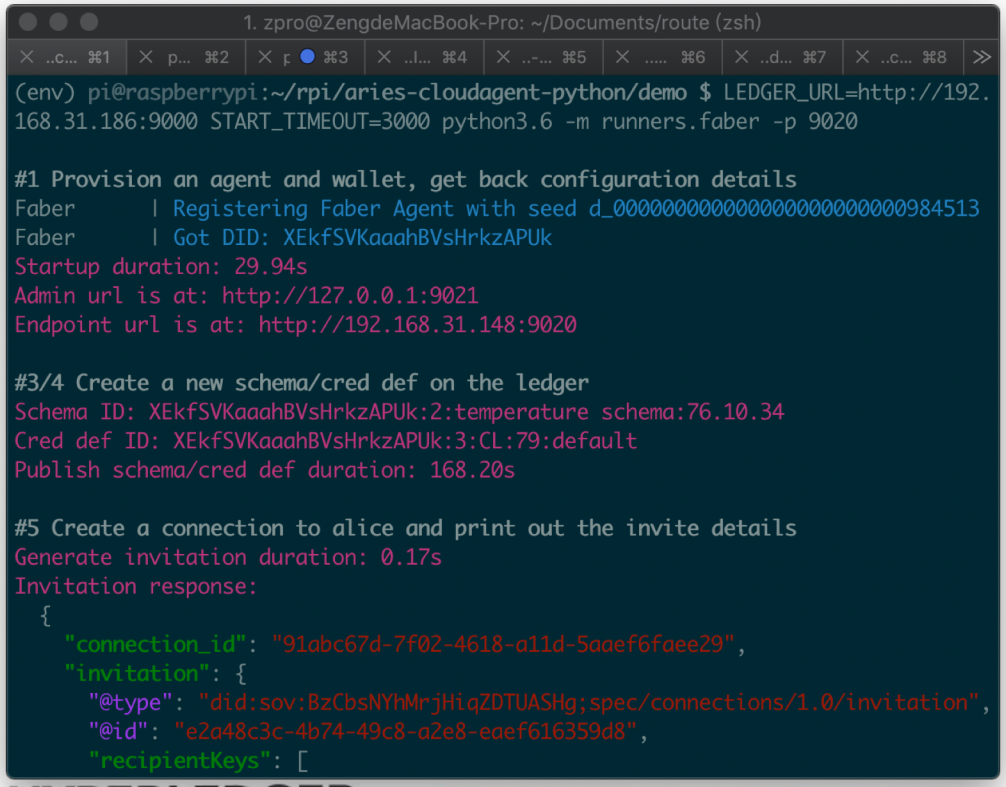
- **Project Deliverables:**
  - Deliverable 1: An Indy agent that can run on raspberry pi
  - Deliverable 2: Aries RFC and an implementation of it that describing the interactions with raspberry pi.
  - Deliverable 3: Image of Raspbian-Indy-Agent for easy deployment onto a Raspberry Pi.
  - Deliverable 4: A demo showing the functionalities of the agent.

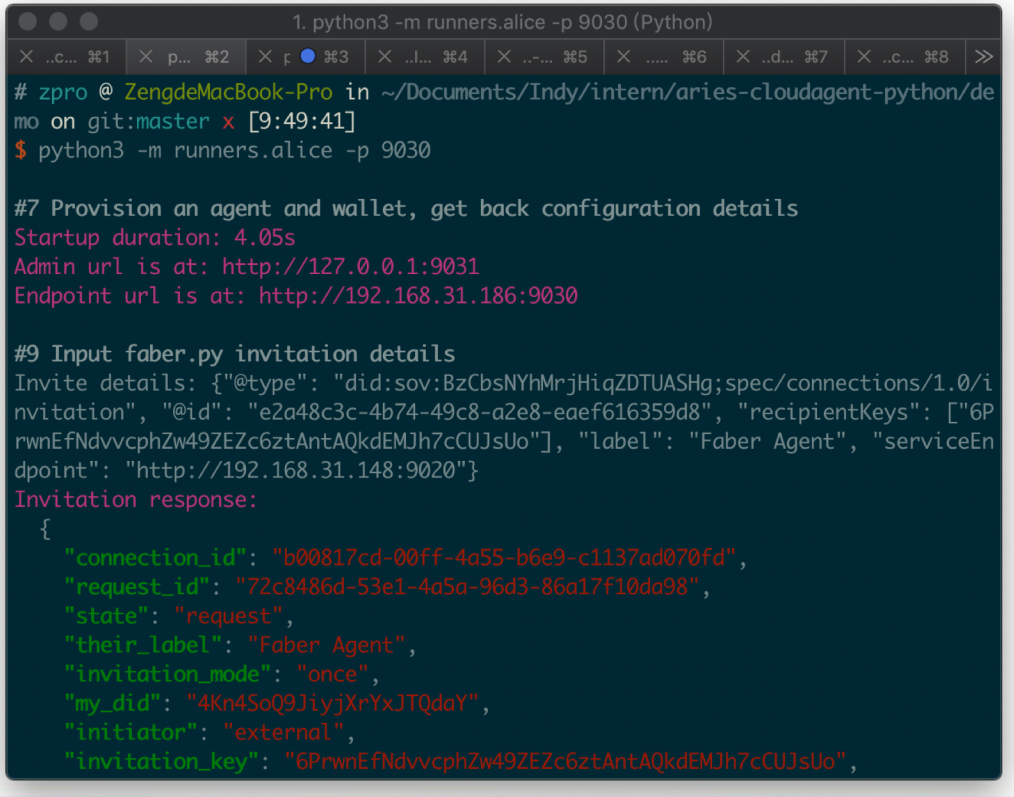
Raspberry Pi Indy agent

- **Aries RFC -** Raspberry Pi Interactions
- **Messages:**
  - ReadSensor: It consists of a list of sensors under the field "sensors”.
  - SensorValue: It has sensor values that "read" request asks for, depending on the contents within "sensors", it has different fields.
  - DisplayMessage: It displays a text message with the specified text colour and background colour.
  - DisplayLetter: It displays a single letter with the specified text colour and background colour.
  - SetPixels: It updates the entire LED matrix based on a 64 length list of pixel values.

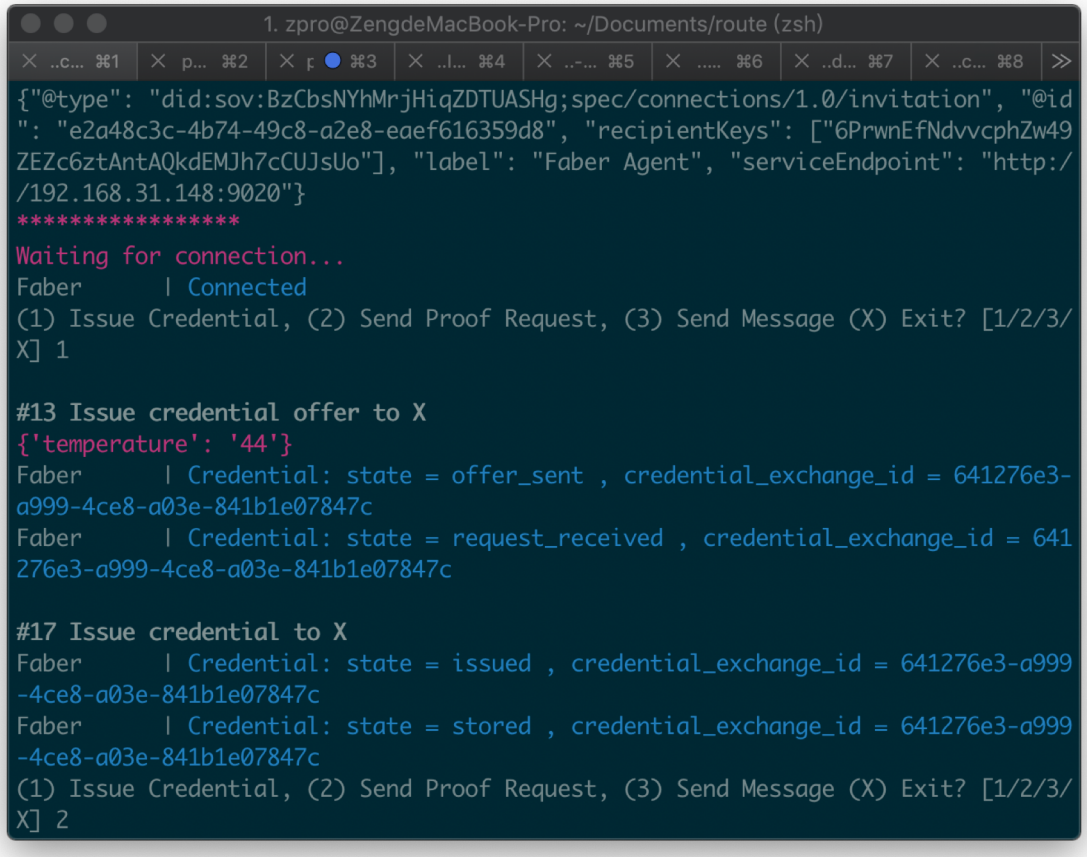
Raspberry Pi Indy agent

- **Demo for issuing credentials**
- **Ledger:** Indy Pool (PC)
- **Agent 1:** Faber - credential issuer (Raspberry Pi 3B+)
- **Agent 2:** Alice – request credential (PC)

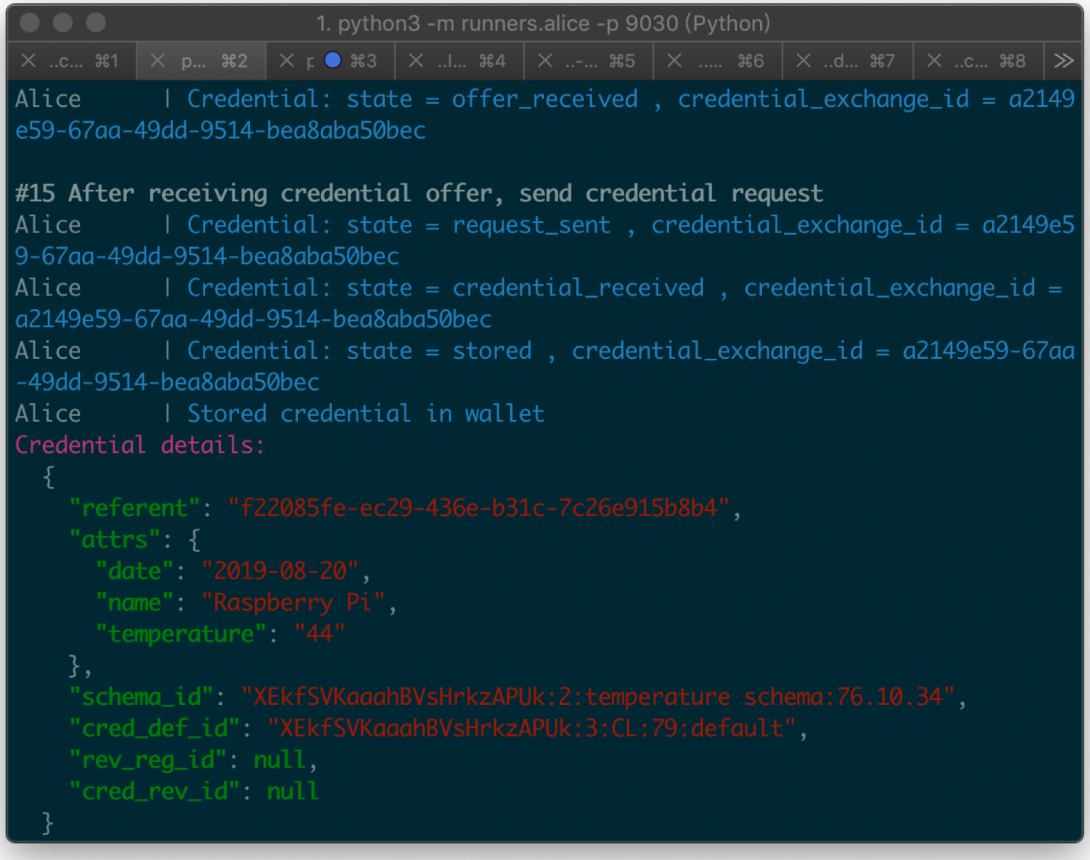




Raspberry Pi Indy agent

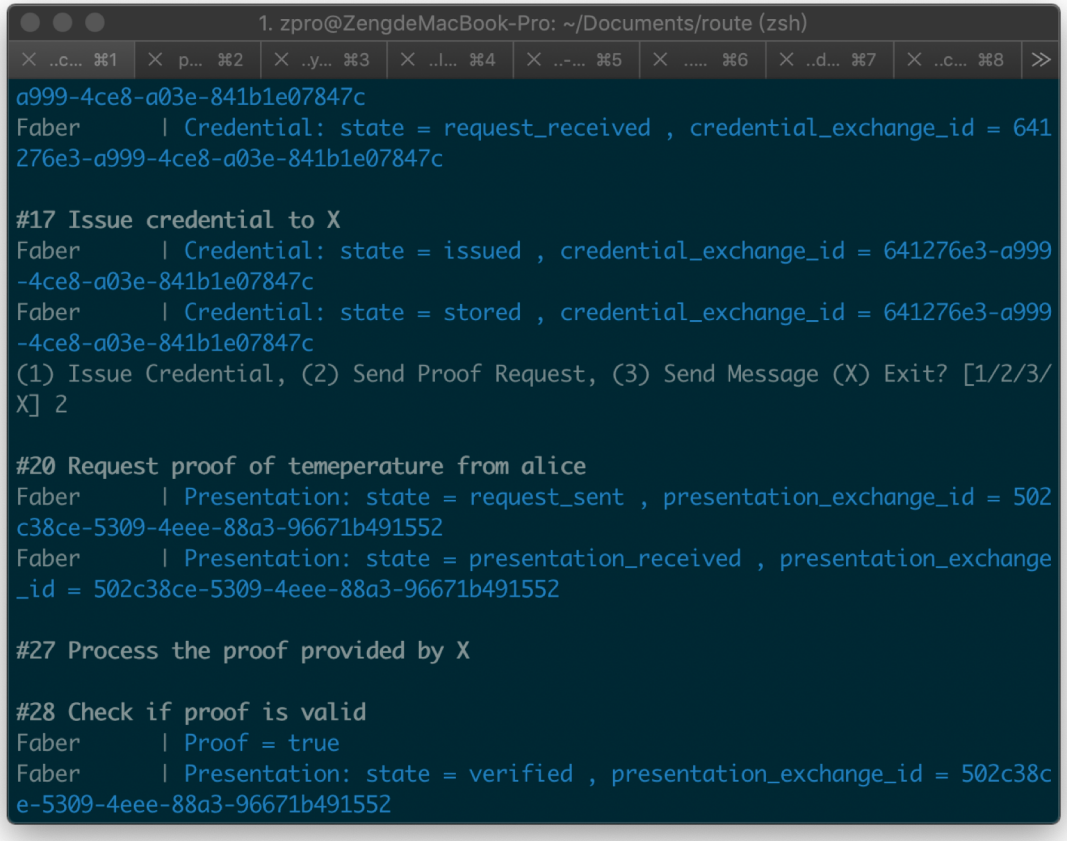


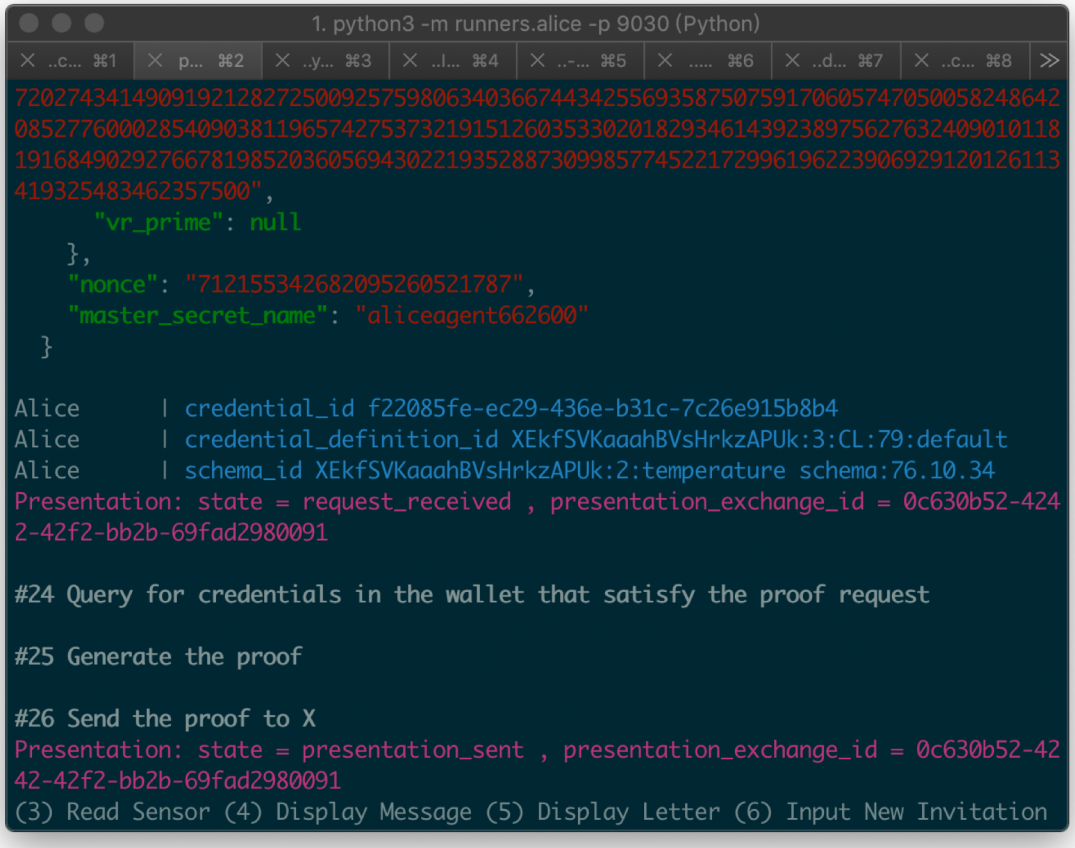
Issue temperature credential (Faber)



Received and saved credential (Alice)

Raspberry Pi Indy agent





Request proofs from Alice and it is verified (Faber)

Sent presentation to Faber (Alice)

Raspberry Pi Indy agent

- **Sending Messages:**
  - ReadSensor

```
(3) Read Sensor (4) Display Message (5) Display Letter (6) Input New Invitation
(X) Exit? [3/4/5/6/X]: 3
Enter sensor names: temperature humidity
Alice      | Sensor Value: {'temperature': 45.05219268798828, 'pressure': None,
'humidity': 45.33596420288086, 'orientation': None, 'accelerometer': None, 'comp
ass': None, 'gyroscope': None, 'stick_events': None, 'pixels': None}
```

  - DisplayMessage

```
(3) Read Sensor (4) Display Message (5) Display Letter (6) Input New Invitation
(X) Exit? [3/4/5/6/X]: 4
Enter your message: Hello world
Alice      | Received message: DisplayMessage message received
```

  - DisplayLetter

```
(3) Read Sensor (4) Display Message (5) Display Letter (6) Input New Invitation
(X) Exit? [3/4/5/6/X]: 5
Enter your letter: A
Alice      | Received message: DisplayLetter message received
```

Raspberry Pi Indy agent

- **Project Execution & Accomplishments**:
- Still in progress: Make indy agent on raspberry Pi able to issue credentials related to the data from its sensors.
- Most proud of: Build the Indy SDK on (Raspberry pi) ARMv7 with no sufficient tutorials. Tried many approaches and finally work.
- Most challenging: Learning the agent deployment with docker.
- Bugs documented: Timeout issue for DetectProcess() in Aries cloud agent demo directly running on pi. Opened an issue and it was fixed in later commit.

Raspberry Pi Indy agent

- **Recommendations for future work:**
- Extend to other IoT devices
- Add support for more add-on board
- Add support for more messaging type