

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)

```
1. zpro@ZengdeMacBook-Pro: ~/Documents/route (zsh)
(env) pi@raspberrypi:~/rpi/aries-cloudagent-python/demo $ LEDGER_URL=http://192.168.31.186:9000 START_TIMEOUT=3000 python3.6 -m runners.faber -p 9020

#1 Provision an agent and wallet, get back configuration details
Faber | Registering Faber Agent with seed d_0000000000000000000000000984513
Faber | Got DID: XEkfSVKaaahBVsrkzAPUk
Startup duration: 29.94s
Admin url is at: http://127.0.0.1:9021
Endpoint url is at: http://192.168.31.148:9020

#3/4 Create a new schema/cred def on the ledger
Schema ID: XEkfSVKaaahBVsrkzAPUk:2:temperature schema:76.10.34
Cred def ID: XEkfSVKaaahBVsrkzAPUk:3:CL:79:default
Publish schema/cred def duration: 168.20s

#5 Create a connection to alice and print out the invite details
Generate invitation duration: 0.17s
Invitation response:
{
  "connection_id": "91abc67d-7f02-4618-a11d-5aaef6faee29",
  "invitation": {
    "@type": "did:sov:BzCbsNYhMrjHiqZDTUASHg;spec/connections/1.0/invitation",
    "@id": "e2a48c3c-4b74-49c8-a2e8-eaef616359d8",
    "recipientKeys": [
```

```
1. python3 -m runners.alice -p 9030 (Python)
# zpro @ ZengdeMacBook-Pro in ~/Documents/Indy/intern/aries-cloudagent-python/demo on git:master x [9:49:41]
$ python3 -m runners.alice -p 9030

#7 Provision an agent and wallet, get back configuration details
Startup duration: 4.05s
Admin url is at: http://127.0.0.1:9031
Endpoint url is at: http://192.168.31.186:9030

#9 Input faber.py invitation details
Invite details: {"@type": "did:sov:BzCbsNYhMrjHiqZDTUASHg;spec/connections/1.0/invitation", "@id": "e2a48c3c-4b74-49c8-a2e8-eaef616359d8", "recipientKeys": ["6PrwnEfNdvvcphZw49ZEZc6ztAntAQkdEMJh7cCUJsUo"], "label": "Faber Agent", "serviceEndpoint": "http://192.168.31.148:9020"}
Invitation response:
{
  "connection_id": "b00817cd-00ff-4a55-b6e9-c1137ad070fd",
  "request_id": "72c8486d-53e1-4a5a-96d3-86a17f10da98",
  "state": "request",
  "their_label": "Faber Agent",
  "invitation_mode": "once",
  "my_did": "4Kn4SoQ9JiyjXrYxJTQdaY",
  "initiator": "external",
  "invitation_key": "6PrwnEfNdvvcphZw49ZEZc6ztAntAQkdEMJh7cCUJsUo",
```



Raspberry Pi Indy agent

```
1. zpro@ZengdeMacBook-Pro: ~/Documents/route (zsh)
{"@type": "did:sov:BzCbsNYhMrjHiqZDTUASHg;spec/connections/1.0/invitation", "@id": "e2a48c3c-4b74-49c8-a2e8-eaef616359d8", "recipientKeys": ["6PrwnEfNdvvcpH7cCUJsUo"], "label": "Faber Agent", "serviceEndpoint": "http://192.168.31.148:9020"}
*****
Waiting for connection...
Faber | Connected
(1) Issue Credential, (2) Send Proof Request, (3) Send Message (X) Exit? [1/2/3/X] 1

#13 Issue credential offer to X
{'temperature': '44'}
Faber | Credential: state = offer_sent , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c
Faber | Credential: state = request_received , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c

#17 Issue credential to X
Faber | Credential: state = issued , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c
Faber | Credential: state = stored , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c
(1) Issue Credential, (2) Send Proof Request, (3) Send Message (X) Exit? [1/2/3/X] 2
```

Issue temperature credential (Faber)

```
1. python3 -m runners.alice -p 9030 (Python)
Alice | Credential: state = offer_received , credential_exchange_id = a2149e59-67aa-49dd-9514-bea8aba50bec

#15 After receiving credential offer, send credential request
Alice | Credential: state = request_sent , credential_exchange_id = a2149e59-67aa-49dd-9514-bea8aba50bec
Alice | Credential: state = credential_received , credential_exchange_id = a2149e59-67aa-49dd-9514-bea8aba50bec
Alice | Credential: state = stored , credential_exchange_id = a2149e59-67aa-49dd-9514-bea8aba50bec
Alice | Stored credential in wallet
Credential details:
{
  "referent": "f22085fe-ec29-436e-b31c-7c26e915b8b4",
  "attrs": {
    "date": "2019-08-20",
    "name": "Raspberry Pi",
    "temperature": "44"
  },
  "schema_id": "XEKfSVKaaahBVHrKzAPUk:2:temperature schema:76.10.34",
  "cred_def_id": "XEKfSVKaaahBVHrKzAPUk:3:CL:79:default",
  "rev_reg_id": null,
  "cred_rev_id": null
}
```

Received and saved credential (Alice)

Raspberry Pi Indy agent

```
1. zpro@ZengdeMacBook-Pro: ~/Documents/route (zsh)
a999-4ce8-a03e-841b1e07847c
Faber | Credential: state = request_received , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c

#17 Issue credential to X
Faber | Credential: state = issued , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c
Faber | Credential: state = stored , credential_exchange_id = 641276e3-a999-4ce8-a03e-841b1e07847c
(1) Issue Credential, (2) Send Proof Request, (3) Send Message (X) Exit? [1/2/3/X] 2

#20 Request proof of temperature from alice
Faber | Presentation: state = request_sent , presentation_exchange_id = 502c38ce-5309-4eee-88a3-96671b491552
Faber | Presentation: state = presentation_received , presentation_exchange_id = 502c38ce-5309-4eee-88a3-96671b491552

#27 Process the proof provided by X

#28 Check if proof is valid
Faber | Proof = true
Faber | Presentation: state = verified , presentation_exchange_id = 502c38ce-5309-4eee-88a3-96671b491552
```

Request proofs from Alice and it is verified (Faber)

```
1. python3 -m runners.alice -p 9030 (Python)
720274341490919212827250092575980634036674434255693587507591706057470500582486420852776000285409038119657427537321915126035330201829346143923897562763240901011819168490292766781985203605694302219352887309985774522172996196223906929120126113419325483462357500",
  "vr_prime": null
},
"nonce": "712155342682095260521787",
"master_secret_name": "aliceagent662600"
}

Alice | credential_id f22085fe-ec29-436e-b31c-7c26e915b8b4
Alice | credential_definition_id XEkfSVKaaahBVHrKzAPUk:3:CL:79:default
Alice | schema_id XEkfSVKaaahBVHrKzAPUk:2:temperature schema:76.10.34
Presentation: state = request_received , presentation_exchange_id = 0c630b52-4242-42f2-bb2b-69fad2980091

#24 Query for credentials in the wallet that satisfy the proof request

#25 Generate the proof

#26 Send the proof to X
Presentation: state = presentation_sent , presentation_exchange_id = 0c630b52-4242-42f2-bb2b-69fad2980091
(3) Read Sensor (4) Display Message (5) Display Letter (6) Input New Invitation
```

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

