Indicio Mediator Training - Lab 2

In this lab, we will demonstrate mediation when using a "Mobile" device. For this lab, the Aries Toolbox will act as stand in mobile device since it operates the same way as Mobile Agents for message retrieval.

We will use the Toolbox, an ACA-Py Agent representing Alice, and the Indicio Public Mediator as our mediator.

Setup

Prerequisites

Before we get started, we will need to download and install the following:

- Docker
- Docker Compose
- Node JS - Current LTS Release

Obtain Code

Setup for this lab is very similar to our first lab. If you've already cloned the Toolbox and the lab repo before, you can skip these first couple of steps but it is a good idea to check for updates with a `git pull` .

Aries Toolbox

```
$ git clone https://github.com/hyperledger/aries-toolbox
```

Lab Agent Configuration

```
$ git clone https://github.com/indicio-tech/mediator-labs
```

Start up the Toolbox

We will start by installing the Aries Toolbox and its dependencies and starting up an instance.

First, enter the toolbox directory:
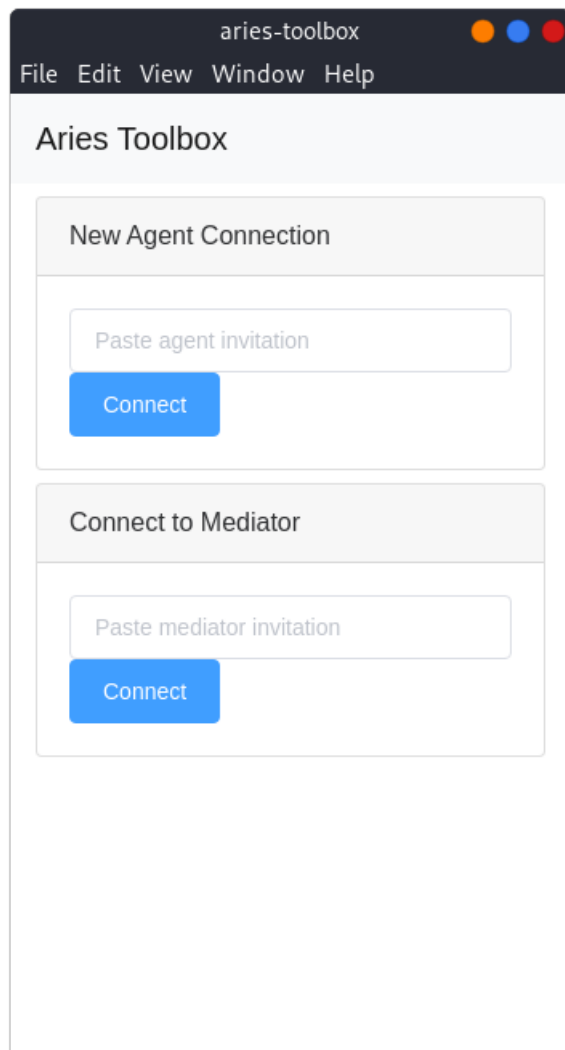
```
$ cd aries-toolbox
```

Now, install dependencies:

```
$ npm install
```

Finally, start the toolbox with:

```
$ npm run dev
```

A window like the following should pop up:

```
aries-toolbox                        ● ● ●
File  Edit  View  Window  Help

  Aries Toolbox

  ┌──────────────────────────────────────┐
  │  New Agent Connection                  │
  │                                        │
  │   ┌──────────────────────────────┐    │
  │   │ Paste agent invitation        │    │
  │   └──────────────────────────────┘    │
  │   ┌───────────┐                       │
  │   │  Connect   │                       │
  │   └───────────┘                       │
  │                                        │
  └──────────────────────────────────────┘
  ┌──────────────────────────────────────┐
  │  Connect to Mediator                   │
  │                                        │
  │   ┌──────────────────────────────┐    │
  │   │ Paste mediator invitation     │    │
  │   └──────────────────────────────┘    │
  │   ┌───────────┐                       │
  │   │  Connect   │                       │
  │   └───────────┘                       │
  │                                        │
  └──────────────────────────────────────┘
```

Start the Alice Agent

Next, we will start the Alice Agent that will serve as one actor in our lab.

Alice is configured as a Docker container with some additional infrastructure around it that will be started using Docker Compose.

First, enter the lab directory:

```
$ cd mediator-labs/lab2
```

Then run the compose script:

```
$ docker-compose up --build
```

Running on Linux

A separate compose script is included for Linux. To run it, use the following:

```
$ docker-compose -f docker-compose-linux.yml up --build
```

If you are a RHEL based system, you can also use `podman` directly with the included `Makefile.podman`. Feel free to explore this file and use it if you are comfortable but the details of its usage will not be shown here.

Output

You should see output for the Alice Agent similar to the following:

```
                  Alacritty                                    ● ● ●
alice_1      | ::::::::::::::::::::::::::::::::::::::::::::::::
alice_1      | :: Alice                                    ::
alice_1      | ::                                          ::
alice_1      | ::                                          ::
alice_1      | :: Inbound Transports:                      ::
alice_1      | ::                                          ::
alice_1      | ::    - http+ws://0.0.0.0:3000              ::
alice_1      | ::                                          ::
alice_1      | :: Outbound Transports:                     ::
alice_1      | ::                                          ::
alice_1      | ::    - http                                ::
alice_1      | ::    - https                               ::
alice_1      | ::                                          ::
alice_1      | :: Administration API:                      ::
alice_1      | ::                                          ::
alice_1      | ::    - http://0.0.0.0:3001                 ::
alice_1      | ::                                          ::
alice_1      | ::                              ver: 0.6.0-pre ::
alice_1      | ::::::::::::::::::::::::::::::::::::::::::::::::
alice_1      |
alice_1      | Listening...
alice_1      |
alice_1      | Created new invitation
alice_1      |     connection: {'invitation_key': '4ARRXnyJytPTXVNtJcAcszsiuv16L8yxQPDbr2YrEsh1', 'inv
itation_mode': 'once', 'state': 'invitation', 'accept': 'auto', 'updated_at': '2021-02-09 02:33:35.50872
3Z', 'connection_id': 'c2ee914b-2715-4cab-8480-9ba0a8d22e25', 'routing_state': 'none', 'created_at': '20
21-02-09 02:33:35.508723Z', 'their_role': 'invitee', 'rfc23_state': 'invitation-sent'}
alice_1      |
alice_1      | Invitation URL (Connections protocol):
alice_1      | https://7443e46df232.ngrok.io?c_i=eyJAdHlwZSI6ICJkaWQ6c292OkJ6Q2JzTlloTXJqSGlxWkRUVUFTS
Gc7c3BlYy9jb25uZWN0aW9ucy8xLjAvaW52aXRhdGlvbiIsICJAaWQiOiAiNTFlMGNhOTEtMThhNC00MWYzLThmOWYtM2Y0MmY4OTIxZ
jlkIiwgInJlY2lwaWVudEtleXMiOiBbIjRBUlJYbnlKeXRQVFhWTnRKY0Fjc3pzaXV2MTZMOHl4UVBEYnIyWXJFc2gxIl0sICJsYWJlb
CI6ICJBbGljZSAoQWRtaW4pIiwgInNlcnZpY2VFbmRwb2ludCI6ICJodHRwczovLzc0NDNlNDZkZjIzMi5uZ3Jvay5pbyJ9
2  ❯  0 sudo ❯                                                              ❮ 21:34:02
```
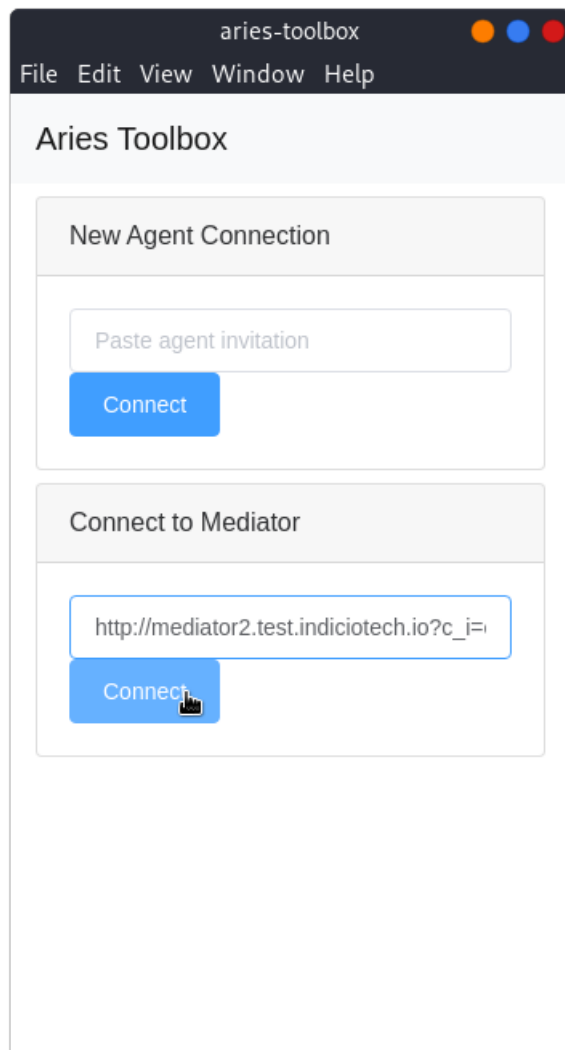
Connect Toolbox to Mediator

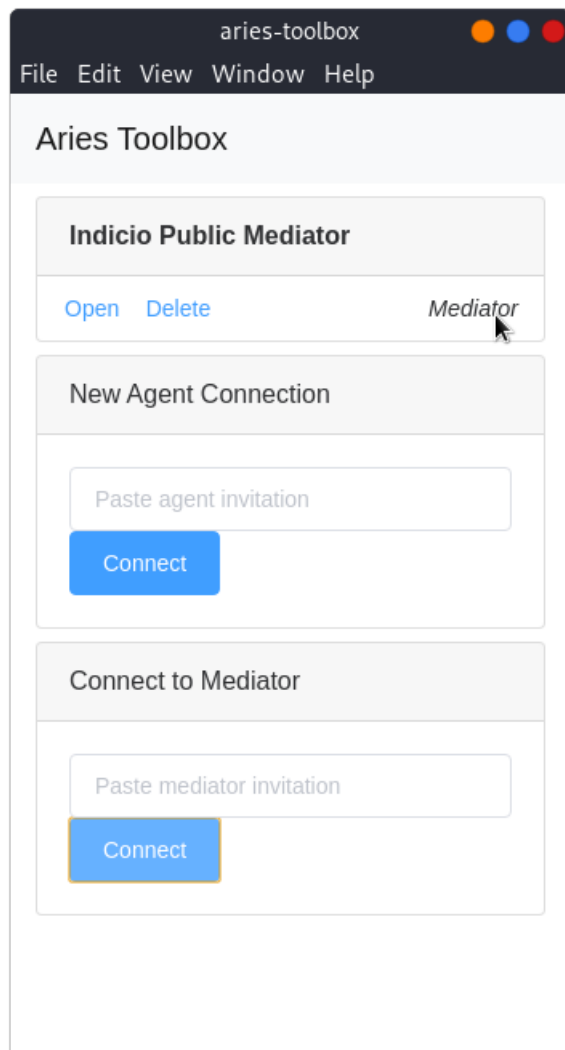Follow this link to the Indicio Public Mediator: https://indicio-tech.github.io/mediator/
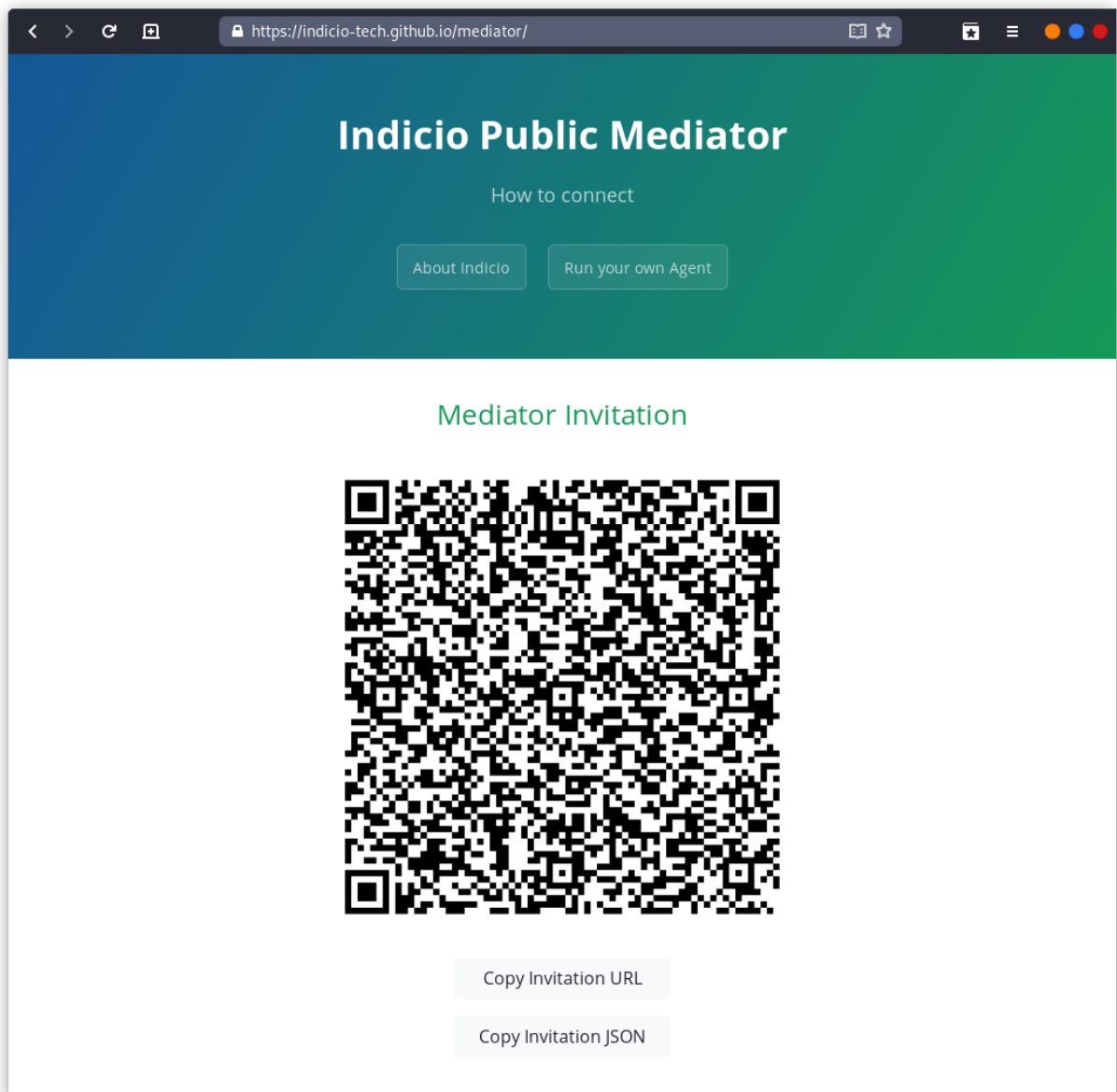
Click **Copy Invitation URL**. This will copy a connection invitation from the mediator to our clipboard.
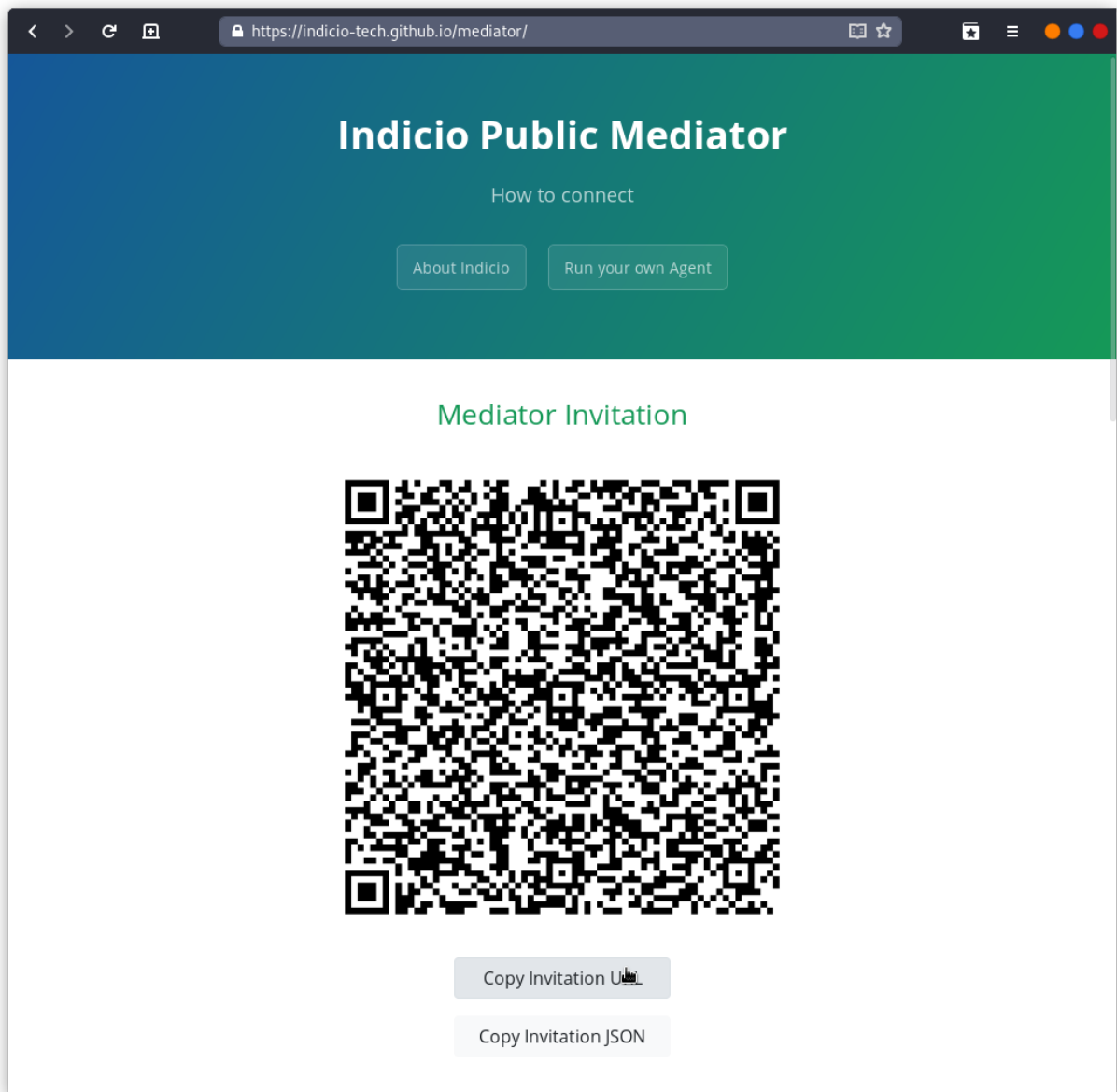
Next, paste the invitation into the **Connect to Mediator** box in the toolbox and press **Connect**.

The **Indicio Public Mediator** connection will appear in our agent list and after a moment *Mediator* will appear in the box as well.

This indicates that the Toolbox has connected to the **Indicio Public Mediator** agent, requested mediation, and received the notification of granted mediation. The Toolbox is now ready to receive connections through the Mediator.

Connect Alice and the Toolbox through the Mediator

Now, we need to copy the invitation URL from the **Alice** terminal.

```
                                    Alacritty                                       ● ● ●
alice_1          | :::::::::::::::::::::::::::::::::::::::::::::::::             [0/69]
alice_1          | :: Alice                                       ::
alice_1          | ::                                             ::
alice_1          | ::                                             ::
alice_1          | :: Inbound Transports:                         ::
alice_1          | ::                                             ::
alice_1          | ::   - http+ws://0.0.0.0:3000                  ::
alice_1          | ::                                             ::
alice_1          | :: Outbound Transports:                        ::
alice_1          | ::                                             ::
alice_1          | ::   - http                                    ::
alice_1          | ::   - https                                   ::
alice_1          | ::                                             ::
alice_1          | :: Administration API:                         ::
alice_1          | ::                                             ::
alice_1          | ::   - http://0.0.0.0:3001                     ::
alice_1          | ::                                             ::
alice_1          | ::                             ver: 0.6.0-pre ::
alice_1          | :::::::::::::::::::::::::::::::::::::::::::::::::
alice_1          |
alice_1          | Listening...
alice_1          |
alice_1          | Created new invitation
alice_1          |     connection: {'invitation_key': '4ARRXnyJytPTXVNtJcAcszsiuv16L8yxQPDbr2YrEsh1', 'inv
itation_mode': 'once', 'state': 'invitation', 'accept': 'auto', 'updated_at': '2021-02-09 02:33:35.50872
3Z', 'connection_id': 'c2ee914b-2715-4cab-8480-9ba0a8d22e25', 'routing_state': 'none', 'created_at': '20
21-02-09 02:33:35.508723Z', 'their_role': 'invitee', 'rfc23_state': 'invitation-sent'}
alice_1          |
alice_1          | Invitation URL (Connections protocol):
alice_1          | https://7443e46df232.ngrok.io?c_i=eyJAdHlwZSI6ICJkaWQ6c292OkJ6Q2JzTlloTXJqSGlxWkRUVUFTS
Gc7c3BlYy9jb25uZWN0aW9ucy8xLjAvaW52aXRhdGlvbiIsICJAaWQiOiAiNTFlMGNhOTEtMThhNC00MWYzLThmOWYtM2Y0MmY4OTIxZ
jlkIiwgInJlY2lwaWVudEtleXMiOiBbIjRBUlJYbnlKeXRQVFhWTnRKY0Fjc3pzaXV2MTZMOHl4UVBEYnIyWXJFc2gxIl0sICJsYWJlb
CI6ICJBbGljZSAoQWRtaW4pIiwgInNlcnZpY2VFbmRwb2ludCI6ICJodHRwczovLzc0NDNlNDZkZjIzMi5uZ3Jvay5pbyJ9
2  ❯ Copy  ❯ 0 [tmux] ❯                                                                 ❮ 21:45:33
```
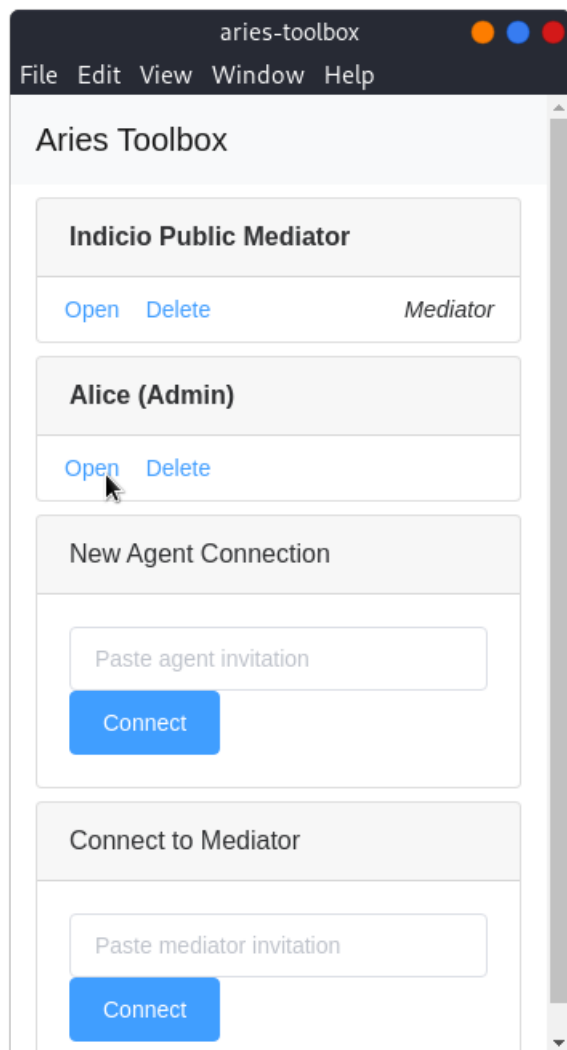
Paste the invitation in the **New Agent Connection** box in the Toolbox agent list window. Press **Connect**.
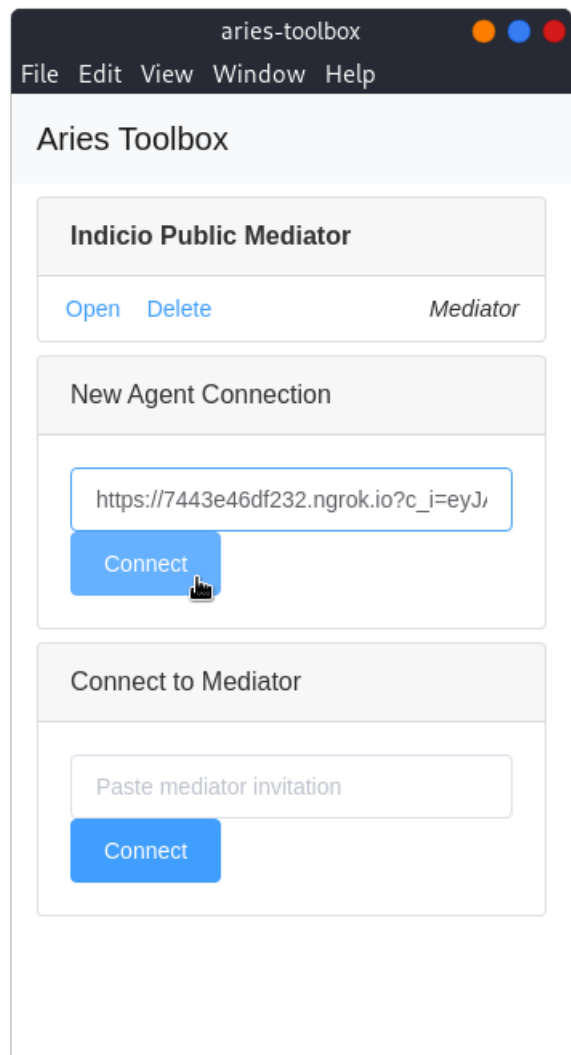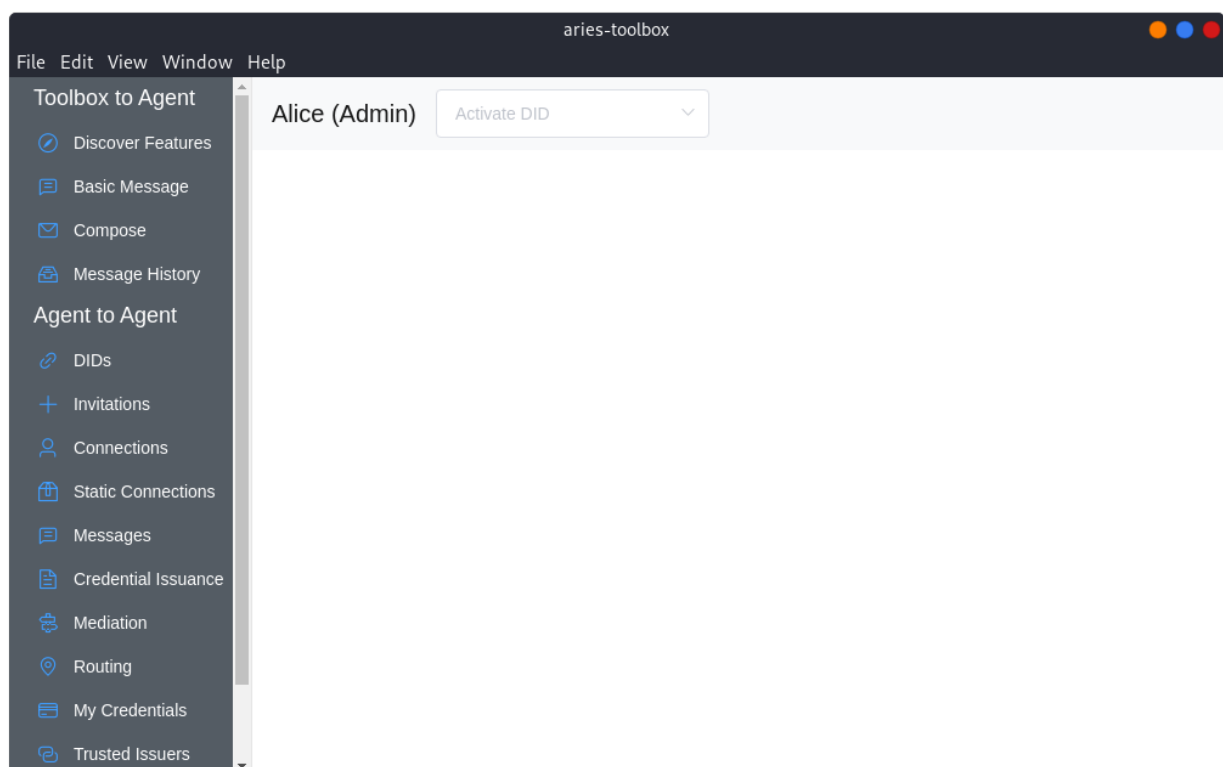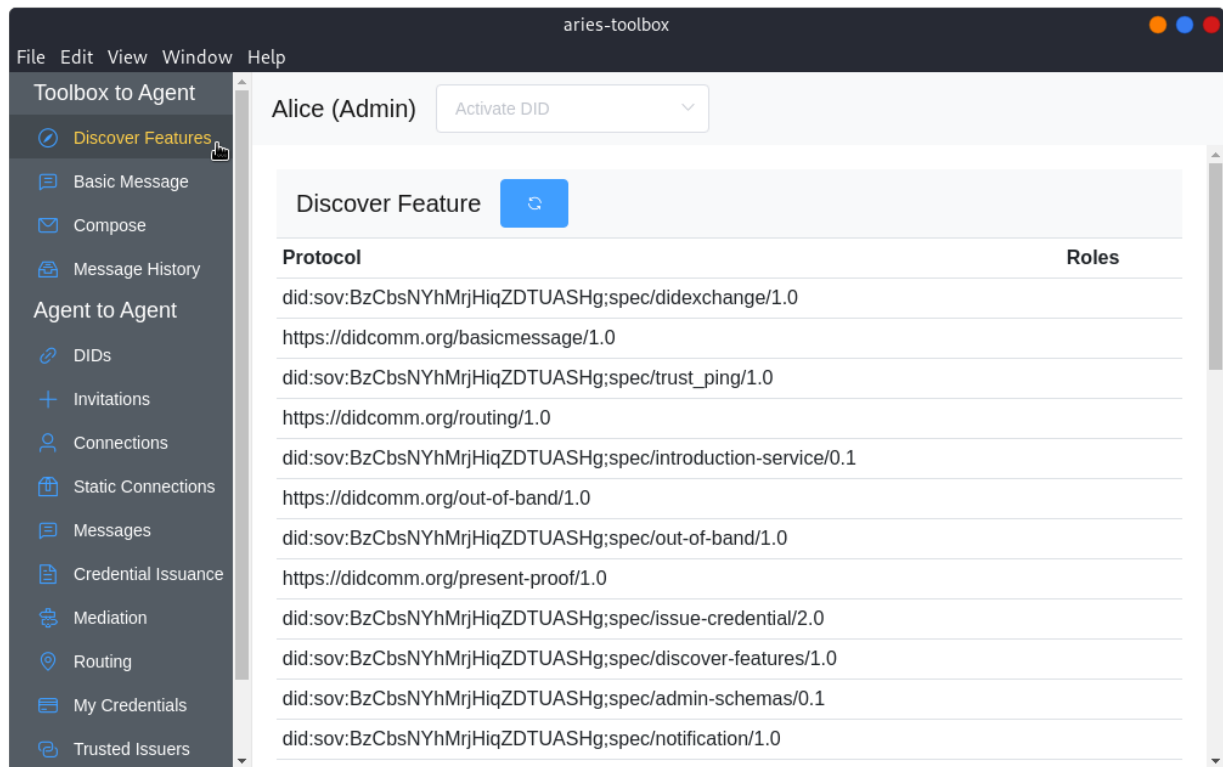
The **Alice (Admin)** agent will appear in our agent list. Click **Open** on the **Alice (Admin)** agent.

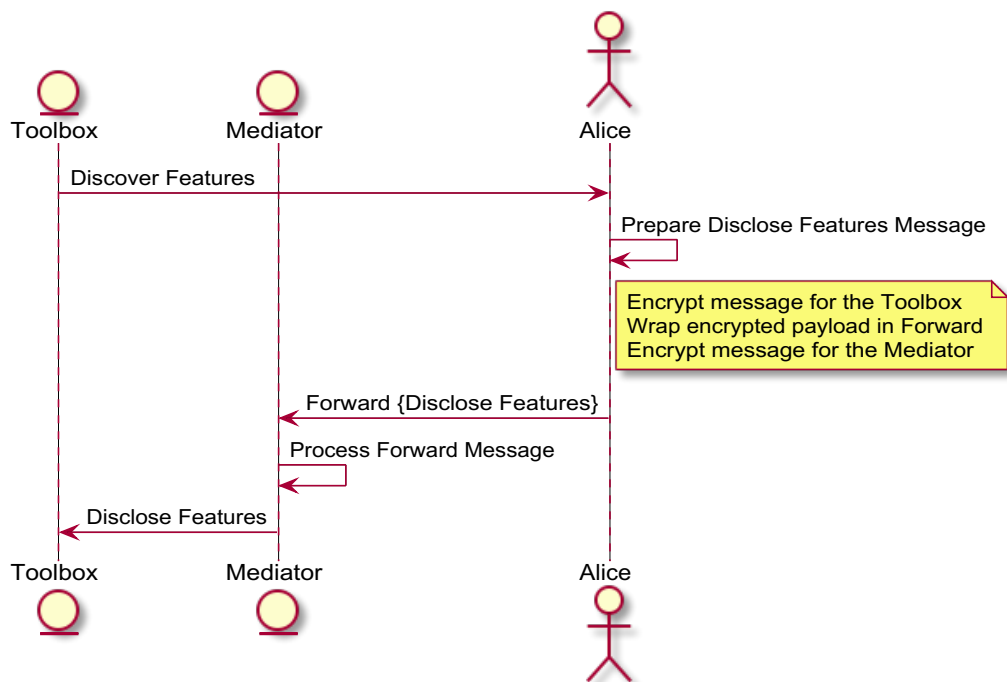A new agent window will appear representing the **Alice (Admin)** agent.

Select **Discover Features** from the navigation menu.

We have now validated that our mediated connection is functioning as expected. Congratulations! You have now successfully completed a connection from an Agent that has no endpoint without a mediator.

In Review

1. The Toolbox sent a `Discover Features` message directly to Alice using the endpoint Alice reported during the connection protocol.
2. Alice prepares a `Features Disclose` message for the toolbox, encrypts the message for the Toolbox, wraps the encrypted envelope in a `Forward` message, then encrypts the message for the Mediator.
3. The Mediator decrypts the message, inspects the Forward message, looks up the `to` field in its routing tables to find that the message should be sent to the Toolbox, then sends the payload to the Toolbox.
4. The Toolbox receives the message, decrypts it, and processes the message as normal.