Resonate Community Credentials

This is Resonate's reference implementation of the Verifiable Credentials Data Model ("VC"). It is intended to explain the architecture ("VC Architecture") and provide a working example.

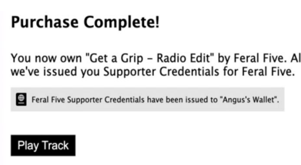
There are four parts to the implementation:

1. Application (vc-app). This is a reference implementation for how Resonate Applications, particularly our player, will interact with the VC Architecture. It is a Node.js (Express) app.
2. VC Layer (vc-layer-api). This is a dummy VC layer API provided by Verifiable Credentials Ltd, containing dummy endpoints performing the following functions:
   - VC Issuer
   - VC Holder
   - VC Verifier
3. User API (vc-user-api). This is a reference implementation for how the Resonate User API will interact with the VC Architecture. It is a Node.js app.
4. Service (discourse-verifiable-credentials). This is a reference implementation for how the Resonate VC Architecture will be used by services, both inside and outside the Resonate ecosystem. It is an open source Discourse Plugin that generically implements the VC Verification, with a specific implementation of Resonate's VC implementation.

Use Case

The reference implementation contains a complete and working example of the reference use case:

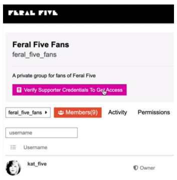
1. Verifying whether a person has paid for content produced by, or performances conducted by, an artist.
2. Granting access to restricted content and engagement based on the verification.



Purchase Complete!

You now own "Get a Grip – Radio Edit" by Feral Five. Also, we've issued you Supporter Credentials for Feral Five.

Feral Five Supporter Credentials have been issued to "Angus's Wallet".



Feral Five Fans
feral_five_fans

A private group for fans of Feral Five

A few things to note about this use case:

1. The granting of the credential can occur at various places in the Resonate ecosystem, and / or via Resonate partner organisations.
2. Any service, whether controlled by Resonate or not, can use credentials produced by the Resonate VC Architecture.
3. Resonate Verifiable Credentials contain metadata that can be of further use for more granular or varied access depending on the relevant resource policy of the Service.

Running the implementation

To see the reference implementation in action you need to run all four parts. Open a new terminal for each and leave it running:

Application

```
## From this repositories' base directory
cd vc-app
npm install
npm start
```

VC Layer API

The instructions on how to set this up are in the document "Installing_the_Dummy_Verifiable_Credentials_Layer.docx", see "Installation Instructions". Once you have your dummy wallet in your home directory, the `node-red` server running, and the "DummyVCLayer.json" installed leave this server running.

Make sure you run `node-red` in the same directory where your `VCredentials.json` file is located.

VC User API

```
## From this repositories' base directory
cd vc-user-api
npm install
node app.js
```

Service

Set up Discourse locally following the relevant guide in this list, then move the discourse-verifiable-credentials folder in your local Discourse `plugins` folder.

Before you start the server, create yourself an admin user, if you haven't already

```
rake admin:create
```

Restart the server, then create a session with the admin user by going to `http://localhost:3000/session/YOUR_USERNAME/become` (change YOUR_USERNAME to the username of the admin user you created).

You should now be logged into your local discourse as an admin user.

Using the implementation

Once you have the implementation running locally (see above; you should have four separate apps running), you can use the implementation like so:

1. In the Application you should see a Player UI with a "Purchase" action. Purchasing an Album is one way a credential can be obtained in the Resonate ecosystem. Click the "Purchase button". Once the album is purchased the Application interacts with the VC Layer to issue the relevant credential.
2. Then go over to your local Discourse instance running the discourse-verifiable-credentials plugin. As an admin, create a group for the artist who made the Album and set the "Membership > Access" permissions to "Allow users to send membership requests to group owners"
3. Open an incognito window and create a new (non-admin) user account by signing up normally, then navigate to the page of the group you created in 2. You should see a "Verify Credential" button beneath the group description.

When you click the "Verify Credential" button the Discourse Plugin will use the VC Layer (both the Holder's wallet and the Verifier) to verify that the user has the relevant credential, and add them to the group if they do.

https://github.com/resonatecoop/verifiable-credentials