UCAR JSON-LD back-end validator and GUI installation instructions 2019-04-30

The UCAR JSON-LD generator has three parts

1) The splash screen. Static html and css
2) The React UI. This works off of schema files to generate a UI.
3) The back-end Validator. This uses Node to run.
   All of these run off of a Node web server. Assuming you want to run this securely, you'll need to install keys and certificates so you can run https instead of http.

The installation instructions below assume a CentOS 7 or Red Hat Enterprise Linux 7 system. They've been tested with versions 7.1 and 7.5

NODE.js

- Instructions and Binaries
- For example for CentOS7 run the following as root:
  - `curl -sL https://rpm.nodesource.com/setup_10.x | bash -`

Centos Prep

- You shouldn't need SELinux; so you can disable as root:
  - `setenforce 0`
  - Edit /etc/selinux/config
    - `SELINUX=disabled`
- Make sure you're using the latest https as root
  - `yum update -y nss curl libcurl`
- Install Epel repository as root
  - `yum -y install epel-release`
    - If you have them, solve "cannot retrieve metalink" issue editing `/etc/yum.repos.d/epel.repo` and `/etc/yum.repos.d/epel-testing.repo` files, commenting all entries starting with `mirrorlist=...` and uncommenting all the entries starting with `baseurl=...`but this should be avoided by the yum update command above
- `yum groupinstall 'Development Tools'`
- `curl -sL https://rpm.nodesource.com/setup_10.x | bash -`
- `sudo yum install -y nodejs`

Test node

- `node`
- Should show '>'
- Enter `1+1`
- Should show `2`
- Enter `process.version`
- Should show `v10.*.*`

- Enter `process.exit()` to quit

Initialize Backend (run as root)

- `cd /var`
- `git clone https://github.com/earthcubearchitecture-project418gui/server/jsonld-ucar-isti`
- `cd /var/jsonld-ucar-isti`
- `npm install`
- Modify node.jsonld.service and put here the name of your server SSL certificates. You can also change other parameters here, e.g. if you want to use somewhere besides /var/www for your document root. (ExecStart, Environment=NODE_CERT, Environment=NODE_KEY and WorkingDirectory)
    - If you don't have a certificate and want to run with http right now, append " --unsecure" to the end of the ExecStart line, e.g.:
        - ExecStart=/usr/bin/node /var/jsonld-ucar-isti/src/back/web.js --unsecure

      This will ignore CERT and KEY and PORT lines and will start up at port 8080

- `cp node.jsonld.service /etc/systemd/system`
- `systemctl start node.jsonld.service`
- `systemctl enable node.jsonld.service`
- `systemctl status node.jsonld.service`
    - If this doesn't show "active (running)" after a few minutes, troubleshoot your `/etc/systemd/system/node.jsonld.service`
- If you're running at port 8081 (--unsecure) or via SSL at 443, you may need to open a port to access externally. As root for example:
    - `firewall-cmd --permanent --zone=public --add-port=8081/tcp`
    - `firewall-cmd --reload`

To inspect Backend systemd log

- `journalctl -u node.jsonld`

In case of failure:

- # Remove service
    - `systemctl stop node.jsonld.service`
    - `systemctl disable node.jsonld.service`
    - `rm /etc/systemd/system/node.jsonld.service`
- To enable verbose logging (log input JSON to file, on parse fail)

    - Pass '--debug' to program. This is the default of node.jsonld.service (End of line 5)

To Build React frontend (recommend building on local machine)

- As root:
    - npm install parcel -g
- As a regular user:
    - git clone https://github.com/earthcubearchitecture-project418gui/client
    - cd client
    - npm install
    - npx patch-package
    - npm run build
- As root:
    - mkdir /var/www
    - Copy contents of dist/ to a folder on server /var/www/ to deploy the interactive UI
        - cp dist/* /var/www
    - Copy the splash directory to get the splash screen
        - cp -r splash /var/www

You should now be able to go to
http://yourserver.com/splash or
https://yourserver.com/splash
You'll see the splash screen, and then clicking the links will get you to the React UI, which will run the back end validator as needed.

CUSTOMIZATION

- In the client repository, you can find and change example URLs here: sets/schema/schemas.js
- If you change the splash screen or location of the back-end validator, you can edit app.jsx
- If you make changes, you need to run "npm run build" and then copy the contents of the dist directory to the docroot again.
- You can edit the splash screen index.html and corresponding css with a text editor, you don't need to do npm for them.