

Join the Stack Overflow Community

Stack Overflow is a community of 6.3 million programmers, just like you, helping each other.
Join them; it only takes a minute:

[Sign up](#)

Installing node modules in Openshift Python gear

I have a Python application in Openshift, with Python 3.3 and PostgreSQL cartridges. The Python cartridge is running Django 1.8, based off the template on the Website.

Recently, I started using Gulp to automate my build, and while it's worked great on my local machine, I can't figure out what to do to use it in Openshift. I have [django-gulp](#) installed so it just runs whenever I use `runserver`, but the Openshift server obviously doesn't have gulp or any plugins installed, so that won't do anything. I don't know how to install them on the server, though.

Including a `package.json` does nothing. I've tested it and it works fine if I go with a node cartridge, but I've got a Python one.

Since npm is on the server, I tried SSHing and running `npm install` manually, but it threw up a permission denied error.

[node.js](#) [npm](#) [openshift](#)

asked Jul 4 at 22:05



Alexskc
31 8

1 Answer

So, after trying to work this out for a while, I've finally figured out a solution. This is a bit convoluted, but it works.

1. Install your own Node and npm. The version included in Openshift is hopelessly outdated and doesn't work. So, you need to install a local version in your dependencies folder. `cd $OPENSHIFT_HOMEDIR/app_root; wget https://nodejs.org/dist/v6.3.0/node-v6.3.0-linux-x64.tar.xz; tar xf node-v6.3.0-linux-x64.tar.xz` Install the dev version. DO NOT install the stable version as that'll break your application. `node-v6.3.0-linux-x64.tar.xz`
2. Change your PATH to include the new node version. `export PATH="$OPENSHIFT_HOMEDIR/app_root/dependencies/node_modules/.bin:$OPENSHIFT_HOMEDIR/app_root/bin:$PATH"` Environment variables get reset whenever you disconnect from the server. You can change them permanently with `rhc env set`, but as this is only used before deployment, I recommend sticking this in `/.openshift/action_hooks/pre_build`.
3. You also need to change the `NPM_CONFIG_USERCONFIG` variable. `export NPM_CONFIG_USERCONFIG=$OPENSHIFT_HOMEDIR/app_root/build-dependencies/.npmrc`. Once again, I recommend doing this in the `pre_build` hook.
4. At this point, you can change the cache without getting a permission denied error. `npm config set cache "$OPENSHIFT_HOMEDIR/app_root/dependencies/.npm"`
5. You can now finally use `npm install`! Install the packages you need. Use the `--prefix` flag to install into the `node_modules` folder.
6. However, to use gulp, you need to have a gulp module installed in the place you're calling from, so `npm install gulp` WITHOUT the prefix flag.
7. You can now call `gulp`! However, your gulpfile will not find the modules unless you edit your gulpfile to use the correct path. So instead of `require('gulp-cssnano')`, you'd have `require([OPENSHIFT_HOMEDIR]/app_root/dependencies/node_modules/gulp-cssnano)`. I keep a separate gulpfile in my Openshift to maintain my sanity.
8. Call gulp with your new gulpfile in `pre_build`.

answered Jul 14 at 18:50



Alexskc
31 8

