Creating a Python Deployment Package on AWS Lambda

Jane Gerard

February 2018

Overview

AWS Lambda is tricky to configure. The big problem is that packages must comply with the architecture of AWS servers, so compiled code will need to be compiled on an Amazon machine. So to make sure that everything works properly, the build order is:

1. Write Code on Local Machine
2. Transfer Code to AWS EC2.
3. Create a Virtual Environment on EC2 and Install Dependencies
4. Zip Code on EC2 and Transfer .zip File back to Local Machine
5. Upload Zip File to Lambda

1 Writing Code

Only thing to keep in mind here is that Lambda will by default only run the function `lambda_handler(event,context)` in the python file `lambda_function.py`. All of the code that you want to run should be placed in here or called from here. Here is the default format for `lambda_function.py`:

```python
# Put your imports here
def lambda_handler(event, context):
    # TODO implement
    return 'Hello from Lambda'
```

2 Transferring Code to AWS EC2

First, you will need an EC2 server. Set one up on AWS and then store the `.pem` file in your home directory (∼/ on Linux based systems). Then, in your home directory, run the command

```
$ chmod 400 <your_file>.pem
```

Then you are going to `scp` the files you want to transfer to the home of your EC2 directory. To do this run the command:

```
$ scp -i "<your_file>.pem" path/to/your/file ec2-user@<EC2_Server_Name>:~/
```

(you can also use the `-r` option to have it copy an entire directory). The EC2 Server Name is can be found on the AWS website.

3 Setting Up a Virtual Environment on EC2

Before doing anything you need to log on to EC2. To do this, at your terminal run the command:

```
$ ssh -i "<your_file>.pem"  ec2-user@<EC2_Server_Name>
```

Now you should be in EC2. We are going to create a virtual environment. This will involve the command virtualenv. Here are the commands you will want to run:

```
$ virtualenv <project_folder_name>
$ cd <project_folder_name>
$ source bin/activate
```

You are now in the virtual environment. Python has been freshly installed in this environment and it will only reference libraries you have installed in this environment. Pip will also install the libraries you need in the folder `lib/python2.7/site-packages/` or in `lib64/python2.7/site-packages/`. So now move all your code and files you need into the virtual environment. Use the command:

```
$ mv <source> <destination>}
```

(The `<source>` of your files will likely be ∼/`<filename>` and the `<destination>` if you are in the folder ∼/`<project_folder_name>` will be `.`). Now simply install the python libraries you need using the `pip` command. For example, if I need requests, I would type:

```
$ pip install requests
```

Do this for every package you need and then test using:

```
$ python -c "import lambda_function as lf; lf.lambda_hadler(<your_test_case>)"
```

When you are done using your virtual environment, simply exit it by typing the command `$ deactivate`.

4 Zipping

Once you have finalized your package and everything works as you intended when running the python command in the virtual environment, it's time to zip it up and send it back to your local machine. First we are going to zip the lib packages. Then we are going to zip the lib64 packages to the zip we've created. Then just zip the code and data files you need to the zip file. To do this correctly, run the following commands

```
$ cd ~/<project_folder_name>/lib/python2.7/site-packages/
$ zip -r9 ~/<project_folder_name>/deployment_package.zip *
$ cd ~/<project_folder_name>/lib64/python2.7/site-packages/
$ zip -gr ~/<project_folder_name>/deployment_package.zip *
$ cd ~/<project_folder_name>
$ zip -g deployment_package.zip <source_file>.py
$ zip -g deployment_package.zip <anything_else>
```

Now all that remains is to retrieve the file on your local machine. To do this exit the EC2 server and then from your local machine run the command:

```
$ scp -i "<your_file>.pem" ec2-user@<EC2_Server_Name>:~/<project_folder_name>/deployment_package.zip .
```

This will copy the file from the VM to your local machine.

5 Upload to Lambda

If your zip file is under 50MB then you can upload the file directly to AWS Lambda. Otherwise you will have to first upload it to S3 and then give lambda the url of the zip file in S3. After you do this, just save and then create a test case to make sure everything works smoothly.