

Inter-operable, reproducible and efficient deployment of software and software changes in the Jetstream open stack environment

BRANDON LWOWSKI, University of Texas at San Antonio

With technology on the rise in the education system, the use of applications and software for such activities is growing exponentially."Jetstream is the first production cloud funded by the NSF for conducting general-purpose science and engineering research as well as an easy-to-use platform for education activities. [Fischer et al. [n. d.]]" The ability to deploy and reproduce software to multiple cloud platforms with the capabilities of seamless version control is essential. With a vast range of cloud platforms spreading the population of users between them, there must be a uniform way to deploy software into a multi-cloud environment. In order to reach a bigger population, you need your software readily available across the major cloud platforms including Openstack, Oracle, Google, Amazon, and Microsoft. Not only do you need to be able to deploy a multi-cloud environment, you need to be able to push new versions onto these cloud platforms as well. In this paper i will discuss the research done during my research experience for undergraduates at Indiana University and working alongside the Jetstream team. I will discuss the implementation of the Spinnaker plug-in, and how i plan to use it in the Jetstream Openstack Cloud. More specifically i will explain some of the successes and failures during this process. At the end of the 8 weeks, I plan to give Jetstream the ability to deploy their software not only to their Openstack environment and be able to spread their applications and software to multiple clouds.

Additional Key Words and Phrases: Openstack, Jetstream, VM, REU, Spinnaker, Jenkins, Devstack

ACM Reference format:

Brandon Lwowski. 2017. Inter-operable, reproducible and efficient deployment of software and software changes in the Jetstream open stack environment. 1, 1, Article 1 (July 2017), 2 pages. https://doi.org/0000001.0000001_2

1 INTRODUCTION

First and foremost, coming into this project, I had zero cloud computing in my knowledge base. I had limited linux command line skills and just completed my second year of my bachelor's degree in computer science. When reading through the procedure and methods that follow this section, some steps may seem redundant or unnecessary to most, but starting from ground zero and building was necessary. I separated my 8 week block into 3 sections. I gave myself 3 weeks to learn as much as I could about openstack and Jetstream. I read wiki pages, watch hours and hours of openstack summit videos, built and launched VMs. The goal was to get as familiar with openstack, openstack command line, and the Jetstream AP as possible in a short amount of time. The next 3 weeks, the goal was to install and configure the Spinnaker plug-in and connect it to

the Jetstream openstack environment. The last 2 weeks of my REU I wanted to test Spinnaker, have a working demo to show the power of Spinnaker, and present it to the Jetstream team.

2 BENEFIT OF MULTI CLOUD SOFTWARE DEPLOYMENT

The major benefit of using the Spinnaker plug-in is the power to deploy software and software changes to multiple cloud platforms. This in itself promotes interoperability and reproducibility. Having this capability greatly improves deploying of software and version control for software developers. For example, being able to release a bioinformatic software created in the Jetstream Atmosphere, and reproduce that same software, using the Spinnaker pipeline, to the customers on the Google cloud just increased the potential audience and users of your software. Now imagine yourself adding another stage into the Spinnaker pipeline to deploy to the Amazon cloud as well. With a few extra steps you potentially tripled the audience of your software.

3 PROCEDURE

In the following sections, I will explain the steps I took in order to get Spinnaker connected to the endpoints of Jetstream. I tried 3 different methods in order to accomplish the installation and configuration of Spinnaker. I will explain in detail how far I got to completion in each method and the difficulties and headaches that I came across during these processes.

3.1 Devstack

When installing Spinnaker for the first time, I thought that if I used the Jetstream Openstack Cloud to implement Spinnaker, and I happened to mess up the configuration, then somehow I could mess up the entire cloud. This misconception led me to a few weeks of frustration. My original plan to install and implement Spinnaker was to create my own test cloud using devstack. With this test cloud, I then could run install and run Spinnaker in a test environment without the fear of crashing Jetstream. If could do so correctly then I could also apply it to the Jetstream Openstack Cloud. The pro's of me trying this first is I learned from scratch the different components and projects used to get a test cloud up and working. The difficult part of this method was Spinnaker is very specific on what release of openstack you are using and minimum requirements for versions needed for each project. After successfully producing a test cloud with the mitaka release, I still had to upgrade each component of the services needed. This ended up being a difficult task that I struggled with and ended up hitting a wall. I could not configure my openstack test cloud correctly in order to get Spinnaker installed and running correctly. After several about a week of trial and error and doing more research on Spinnaker, I reached out to PHD students, Professors, and the Spinnaker Open source community, searching

This work is supported by the National Science Foundation, Indiana University, and the Jetstream Team.

Authors' addresses: Brandon Lwowski, Computer Science Department, University of Texas at San Antonio, One UTSA Circle, San Antonio, TX 78249, US;

© 2017 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in , https://doi.org/0000001.0000001_2.

for advice on moving forward with this problem. After voicing my thought process, I learned I could just use the Jetstream Openstack endpoints given to me in the Jetstream API to connect Spinnaker to the openstack cloud. This discovery lead me to my second method.

3.2 Different VM's

This method may have worked for a person with background in networking and knowledge in json programming. A Lot of the configuration needed was manipulation of json files to connect Jenkins and Spinnaker together. Once they were connected I could then push new code onto my github library which would trigger Jenkins to poll for changes and start the Spinnaker pipeline. This method started off excellent. I had Spinnaker linked correctly to Jetstreams openstack. I correctly installed Jenkins on the separate VM. I was able to push code and have Jenkins poll for changes. The problem occurred when trying to link the Jenkins master to the Spinnaker pipeline. I could not figure out how to configure the Jenkins settings to broadcast to the Spinnaker host.

3.3 Same VM

Installing Jenkins and Spinnaker on the same VM is the method that got me the farthest in a working implementation of a Spinnaker pipeline. Having the 2 applications running on a the same VM made the configuration of connecting Jenkins and Spinnaker a smaller task. There were less steps needed to get Spinnaker and Jenkins talking to each other. Out of all the methods I tried, this was definitely the easiest to get started. Once I had all all the pieces installed and connected, it was time to test the pipeline. This is where things began to get a little difficult. From my first year in computer science and everything up to this point that I had learned about openstack, I have never used Jenkins and the Jenkins job builder before. I had to learn how to create a free style project that Jenkins could build and deploy to trigger Spinnaker. After reading tutorials about Jenkins, I came across a "hello world" type project for Jenkins on the Spinnaker

website. Now I had a Jenkins project that was able to build and poll for changes and also be used in my Spinnaker pipeline. It was time to spin up a instance and bake a VM with the new software from the Jenkins build. This final step is where I am currently stuck on. The Jenkins job builds correctly in the Spinnaker pipeline, but once it's time to bake the VM and deploy it to a cluster group, the pipeline fails. The error I keep coming across is, Spinnaker has the authentication to use Jetstreams openstack, but is not receiving the correct authentication to use the premade jetstream flavors and images. Spinnaker could not pull this information from the Jetstream openstack.

4 CONCLUSION

The Spinnaker plug-in is a powerful "multi-cloud continuous delivery platform for releasing software changes (SOURCE)". While Spinnaker is in its early stages and is used with multiple clouds including Google, Amazon, Microsoft, and Openstack, it is still very immature in the openstack environment. Companies like Target have successfully implemented it into their company's cloud and I think that the Jetstream cloud could benefit greatly from a successful implementation of this plug in. If Jetstream had the available resources to allow their successful educational software to be released to other cloud platforms, it would help spread their resources to a bigger audience and help spread the word of the Jetsteam's mission. While difficult to install and configure correctly for a amateur computer science student, once this plugin matures for the openstack environment, I believe it would be extremely useful in promoting interoperability and reproducibility of applications made in the Jetstream cloud.

REFERENCES

Jeremy Fischer, George Turner, David Y Hancock, Winona Snapp-childs, John Michael Lowe, and Craig A Stewart. [n. d.]. Jetstream : A cloud system enabling learning in higher education communities. ([n. d.]).