Deployment 2

Darren Freeman

“Chron Job”

In this deployment we will be deploying Jenkins and running a chron job.

Now, you may have noticed that the word chron resembles “chronicle” or Kronos the Greek god of time and the king of titans.

The purpose of a chron job is to set up different processes to run at certain times. You can even set it up to automatically terminate your deployment.

The first step here is to install java and Jenkins onto our Linux EC2.

SSH into the EC2 and run these commands:

sudo yum install java-1.8.0-openjdk-devel

*sudo wget -O /etc/yum.repos.d/jenkins.repo \*

[*https://pkg.jenkins.io/redhat-stable/jenkins.repo*](https://pkg.jenkins.io/redhat-stable/jenkins.repo)

*sudo rpm --import* [*https://pkg.jenkins.io/redhat-stable/jenkins.io.key*](https://pkg.jenkins.io/redhat-stable/jenkins.io.key)

*sudo yum upgrade*

*sudo yum install jenkins java-1.8.0-openjdk-devel -y*

*sudo amazon-linux-extras install epel -y*

*sudo yum update -y*

*systemctl start Jenkins*

Then you will want to port into your Jenkins using the public IP and port 8080

Now we are ready to create a new item in Jenkins

1. Configure the job accordingly by adding following details like job details, time to execute the job, build steps etc.;

1. Add the job details from the General tab.
2. Schedule your build from Build Triggers tab by writing following CRON syntax and select ‘Build Periodically’ option.

This uses 24hr time and the det times follow the following format

[**min**] [**hour**][**day/month**] [**month**] [**day/week**]

Therefore *0 18 \* \* \* would result in the process running everyday at 6am*

Build your job to test if everything is working as you’ve expected