When is it appropriate to use containers in cloud deployments and what are the security benefits of doing so? In Project 1 I used containers with my jumpbox provisioner(public ip virtual machine) fanning into a network to create and run ansible scripts used to setup two additional private IP virtual machines and an ELK stack server in addition to an ELK container established on the ELK server. The use for the containers was appropriate because docker containers are less resource intensive compared to creating a new virtual machine to serve the same purpose. The security benefits of using containers is that they are small resources compared to virtual machines so there are less components to update and a smaller target for attackers. Containers also go in and out of use quicker making them more interchangeable components of a network. I configured the VMs by downloading elk-docker. With elk installed I was able to add the groups, ip addresses, and usernames needed to configure the elk stack container in the configuration and host files. I pulled the elk container from elastic.co using and ansible playbook to configure the container itself. To verify the container was running correctly I ran a ping command, it showed successful. You can achieve a similar network scenario using VMs in place of containers. In this example you would install elk directly to the OS. The advantages of doing it without containers would be VM management. As the number of containers a company deploys grows it can become difficult to manage and service the numerous containers connected to the network, with VMs software updates and management is more centralized. The disadvantages of using VMs are that VMs are resource intensive, especially if you are deploying a VM to use one program specifically. Containers allow you run a program without the additional bloat of a VM.