VMWare ESXi in a home lab environment

Breydon Brennan

Eastern Illinois University

CIT-3883 Virtualization

# Introduction

With a constantly evolving information technology industry, the prevenance of virtualization in both testing and production environments has led to decreased data center size and cost. One of the technologies that has helped with the reduction in size of data centers is VMWare ESXi.

To apply this project to my daily life, my dad approached me regarding his current home lab architecture. One of the issues is using multiple services per node, leading to an increased risk of outage and a lack of allocation of resources per service. In addition to allocating resources, keeping the logs for each service is crucial to identifying any problems with a given service and developing a disaster recovery plan in the event of a failure. Lastly, we need to implement an instance of Home Bridge to connect all non-matter compatible devices to Google Home. To circumvent this, we can use VMWare ESXi to create a demonstration of how to divide up each home lab service.

The importance of developing a secure, high-functioning infrastructure is immeasurable. Many smart home devices control pumps to saltwater aquariums, leading to a loss of life if these services were to go down. Planning for a high availability system ultimately allows for the integration of more smart home equipment, like locks and cameras. Ultimately, having a uniform system in place where smart home compatibility should not be a concern when purchasing new equipment.

# Current Infrastructure

Currently, the overall infrastructure consists of a Raspberry Pi with multiple things installed. The services ran on this node include:

* A DNS server
* Home Assistant
* A DHCP server

To meet one of the requirements of combining multiple IoT devices into the Google Home application, Home Bridge is required as a middleware service to allow interaction and automation of various smart devices.

Some of the various hardware upgrades include running on an old Dell OptiPlex to allow for virtualization of the production environment. I went and used my desktop to create a test environment.

# VMWare ESXi Project Structure

The highest priority service is going to be the Home Bridge, which runs on Ubuntu. In total, four virtual machines are required to ensure a smooth transition during the upgrade. Scalability, while nice, is not a primary concern in working within this environment.

## Project Architecture Flowchart

A diagram of a network

Description automatically generated

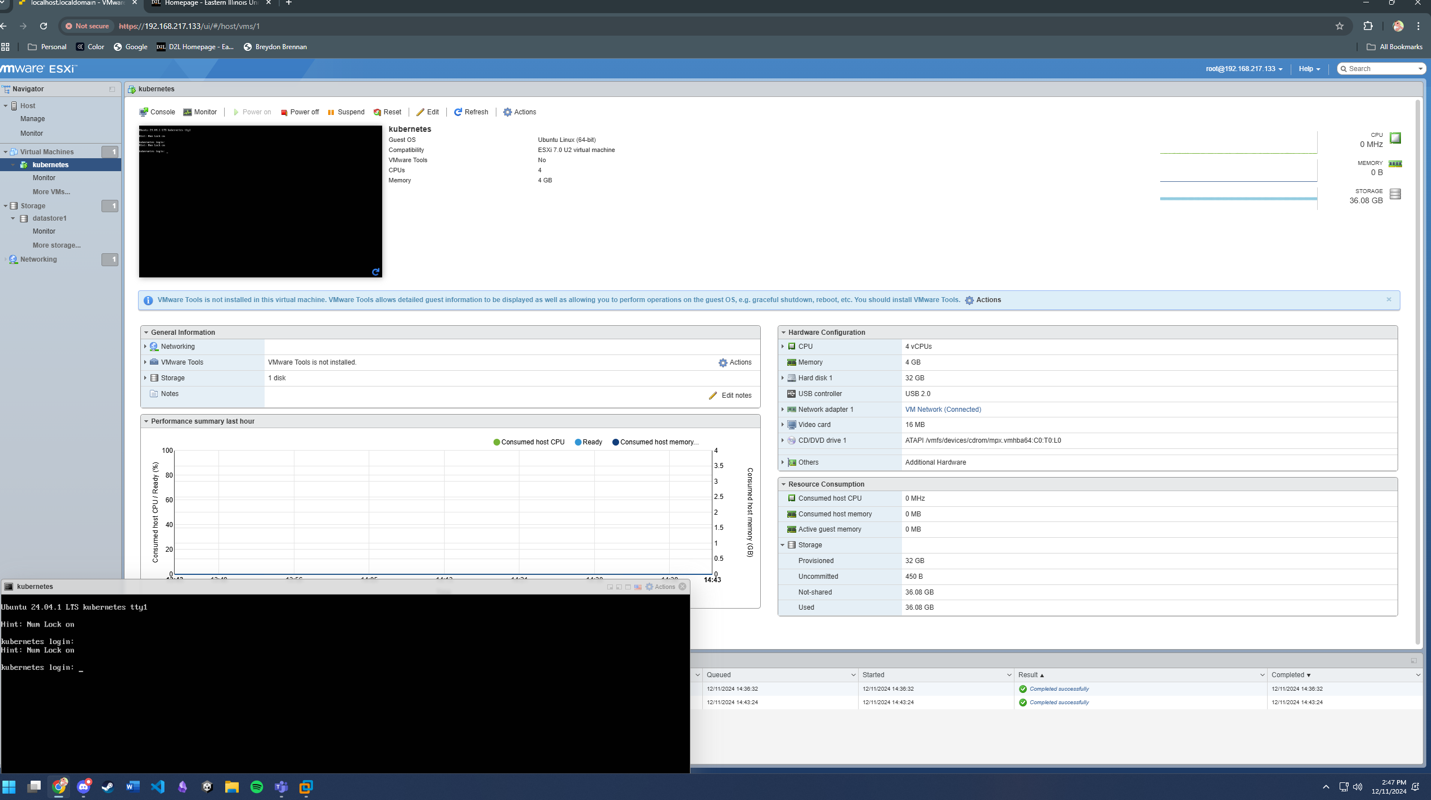
## Networking

The virtual machine running on my local machine uses a NAT connection to provide network access for updating the child virtual machines’ operating system. Static networking isn’t a primary concern since this is a home lab instance.

# Current Instance of Infrastructure

Currently, I have a virtual machine set up to host a Kubernetes cluster to host Home Bridge and Home Assistant. I am working on implementing DuckDNS and a DHCP server.

## Screenshot of Instance



## Problems Encountered

Some of the biggest problems I encountered included resource allocation in ESXi virtual machine and the child virtual machine. To circumvent this, I went ahead and allocated all the resources on my computer to run the operating system through VMWare Workstation Pro. Another problem that I encountered was the fact that the ESXi boot partition currently takes up 127 GB of storage. Because I am limited on storage, I had to shrink the boot partition. I was able to do this by changing the boot partition when installing ESXi. Lastly, I did encounter an error with creating a local datastore. This was because of the combination of the large boot partition and the allocation of resources.

# Successes and Failures

Some of the successes of this project include being able to easily divide resources in child virtual machines while still being able to host each service on a single computer. This solution is also scalable, being able to add more virtual machines depending on whether I plan on making hardware upgrades. ESXi is also not operating system dependent, meaning I could potentially run some Windows services, like Active Directory and Microsoft SQL Server, in the future. Initially, it serves as a good demo for me to test out Kubernetes and Docker on the smart home utilities.

Some of the failures include the trouble of allocating resources to each virtual machine based on performance requirements. I also played around with the idea of testing out a Ceph cluster for object storage, but that was ultimately replaced by ESXi’s datastore feature. Lastly, the trial period of ESXi is 60 days. So, switching to an open-source software, like Proxmox, will be a necessity. With the switch to Proxmox, I can get more control in creating reliable storage with Ceph in the future.

# Summary

In conclusion, the ability to utilize ESXi to solve my problems through the Kubernetes node has been easy. The only problem that I struggled with was changing the boot partition, which required a bit of research to figure out. ESXi only has a 60 day evaluation period, which does not help with this being a long term solution, but this does teach me about how to deploy the infrastructure required for our home lab.