CHALLENGE 1

REHABITING EARTH

By Kellan Damm (@kellandamm)

6-27-16

Table of Contents

Summary

Earth has gotten its second chance after the Zombie Apocalypse. During the initial Zombie take over a group of leaders made a conscientious decision to cryogenically freeze the entire Space X Design and Engineering team. The intentions were to someday have them deploy a fully autonomous machine that can be used on land, air or water. All design decisions were properly vetted to ensure that this machine could sustain all types of environments and run the HumanityLink Software Suite. Plans will be to deploy in Joe, Montana due to name affiliations with the all-time great, Joe Montana, who killed a Zombie with a perfectly thrown spiral. The moon will be second location and like earth will be an active site. Uptime and performance is paramount.

After the Zombie wave had subsided The Space X team was brought back from their deep sleep and quickly began to deploy the new infrastructure. The Space X team ensured that the environment was self-healing and enacted by implementing a fully mesh design to ensure that performance and uptime was achieved. Space X design team took architectural blueprints from a secret project Elon Musk had them developing. This hardware design had next-gen CPU's and the newly released 3D Xpoint Memory technology. But during deployment phase, Donald Trump who always sends the best strikes, used a planned attack to destroy what he thought was a large wave of Zombies. But it missed its target and destroyed the Space X's teams hard work. The team decided to begin scavenging for parts and members of the Space X team found a Fry's Electronics store that had a stockpile of Intel NUC's that would have to suffice for this deployment. The group decided that they would deploy a Docker Swarm Cluster and attempt to keep the design intact the best it could.

Requirements

R01	Able to fully support the HumanityLink Software with ability to handle growth from 3 web servers, 2 application servers, 1 database to 25 web servers, 10 application servers, 5 databases
R02	Application must be utilized locally
R03	Uptime and Performance of HumanityLink Software

Risks

RI01	Location that provides proper power to run components.
RI02	Performance of Available Equipment
R03	Single NIC Port on Intel NUC

Constraints

C01	Unknown Bandwidth capabilities and connectivity between Earth and Moon
C02	Availability of Electricity to run equipment
C03	Donald Trump blowing up primary components
C04	Only equipment available at Fry's Electronics
C05	HumanityLink Software dependencies
C06	Completely new environment
C07	Application growth
C08	Administrative Knowledge of Application

Assumptions

A01	Performance is of utmost importance for application to run properly
A02	Administrators are not familiar with application due to new environment
A03	Hardware is limited due to zombie apocalypse
A04	Satellite connection will only be available between Earth and Moon

Overview

Since connectivity between the Earth and the Moon is unknown the environment for both locations will be deployed first on Earth and then the secondary environment will be transported to the Moon once connectivity is determined and established. A total of 12 Intel NUC's will be utilized in the environment with 50% going to Earth and other 50% going to the moon. As a design decision both sites will be in an Active/Active state to ensure that everyone will have proper performance and access depending on location since latency will be a major concern between Earth and the Moon.

All NUC's will have the latest version of CentOS Minimal installed and Docker will be deployed to run all application and orchestration layers for environment. The docker machines will be also be utilizing CentOS and a separate dockerfile will be implemented for each web, application and database environment. The orchestration layer will be deployed within Docker to ensure High Availability. This layer will be a mixture of Ansible, Ansible

Tower, Universal Control Plane to provision machines and applications and also ensure optimal uptime of all required.

Power

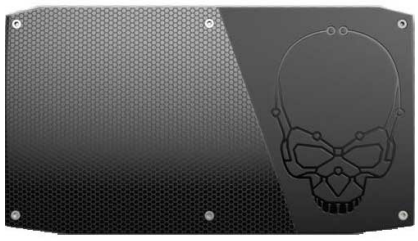
The environment will need to have the ability to run anywhere and be self-contained so best option available is to utilize solar power to run the equipment.

Server Hardware

Intel NUC's that can take up to 32gb of ram will be deployed per location. Item's described were found in Fry's Electronics store. Earth and Moon locations will each location will have 6 Intel NUC's with growth as needed. List below is items needed for one complete Intel NUC.

Quantity	Description
1	Intel Mini PC Intel® NUC Kit NUC6i7KYK (Skylake) i7-6770HQ processor
1	Patriot Memory Signature Line 32GB (2 x 16GB)
2	Samsung 950 Pro M.2 256GB NVME (OS)
1	Patriot LX Series 256GB SDXC Card (Logs)

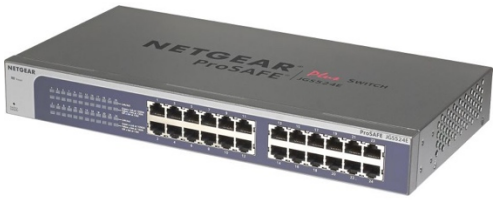
Top View Intel NUC



Front and Rear View Intel NUC

Networking Equipment

- Netgear JGS524E-200NAS 24Port Gigabit Ethernet Switch with VLAN support if needed.
- Two switches will be deployed per location with one switch being a cold standby due to NUC's one networking port.
- A usb nic could be utilized in environment but compatibility with Linux OS could cause stability issues.



Storage

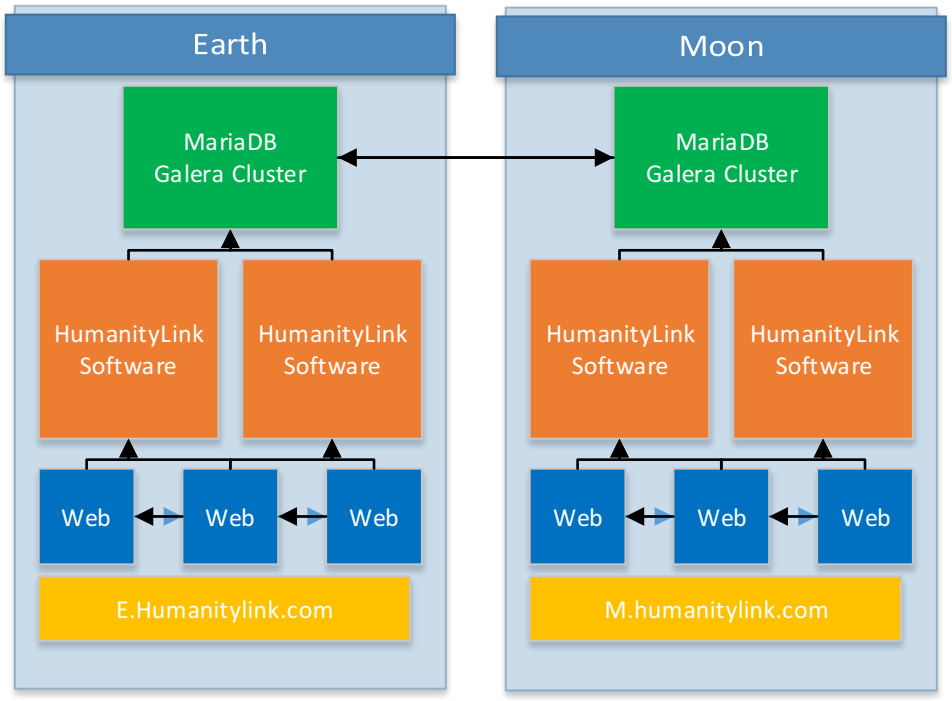
- Each NUC has 2 M.2 Slots that will be filled with a Samsung 950 Pro NVMe SSD.
- SSD's will be configured in a RAID 0 configuration as performance not redundancy is required for Application.
- Redundancy will be distributed through Application.
- Each NUC will also be supplied with a 256GB SDXC Card and Operating System will have all logs offloaded to this card.

User Access

Users will access the environment through web tier by going to E.Humanitylink.com for Earth and M.HumanityLink.com for the moon. Administrators will access the Management layer by accessing the servers in each location.

Application Layout

The entire environment will be configured on Earth and ran there until connectivity between the Earth and the moon is determined. As such the environments will still be configured as separate entities and replication between Databases enabled.



Operating System

The operating system that will be deployed to the environment will be Cent OS 7 Minimal. Only required applications will be installed to limit footprint and optimize performance.

Applications installed in OS (Primary Layer)

wget, python, python-pip, docker-engine, docker-compose, docker-swarm, docker-ucp

DockerFiles per Application

To ensure that all applications have all requirements for applications to run properly. We will separate each tier into a separate dockerfile.

Management Tier

Ansible

UCP

Ansible Tower

Web Tier

Nginx (Memcache)

Nginx as Load balancer

Application Tier

HumanityLink Software Suite

DataBase Tier

MariaDB

Configuration of Tiers

Management

The management layer will consist of Docker Universal Control Plane and Ansible to deploy, monitor and update applications. UCP primarily will handle the monitoring and deployment of Docker files. Ansible's initial purpose will only be for the Primary Layer to ensure that applications are running properly and everything is up to date. It will also be used to provision new hosts.

Web

The web tier will use NGINX as its primary web application. It will be utilized for the HumanityLink web layer but will also be utilized for Docker Control Plane as a load balancer. To balance the load for the Humanity Link software memcache will be also deployed in each docker container to minimize sessions being lost. Memcache will be clustered across the entire Web Tier and remove nginx as the single point of failure.

Application

The application layer will consist of the HumanityLink Software and will be deployed with all dependencies since this is a major constraint it will be required to test and ensure that the applications requirements are at level needed for version of software given. Assumption is the installation guide will provide requirements for dependencies.

Database

The database layer will consist of MariaDB and all its dependencies to run properly. For replication and clustering MariaDB will be configured in each container as a Galera Cluster. These are known features and benefits of this configuration. The benefits of this setup is that the Database can have multiple masters to ensure optimal uptime and application performance.

- Synchronous replication
- Active-active multi-master topology
- Read and write to any cluster node
- Automatic membership control, failed nodes drop from the cluster
- Automatic node joining
- True parallel replication, on row level
- Direct client connections, native MariaDB look & feel

Security

To ensure less attack surface only required applications will be installed and particular settings will be disabled. The benchmark utilized with be OpenSCAP and the requirements for this benchmark will be added to Dockerfiles and any Ansible .yaml files. Within the containers a cgroup will also be utilized to further lessen the attack surface of the application.

References

MariaDB Cluster Configuration

https://mariadb.com/kb/en/mariadb/what-is-mariadb-galera-cluster/

Docker UCP Installation and Load Balancing

http://www.docker.com/sites/default/files/RA_UCP%20Load%20Balancing-Feb%202016_0.pdf

OpenSCAP

https://www.open-scap.org