AUTOMATING IT ALL AND KEEPING TRACK OF IT

Virtual Design Master - Season 5 - Challenge 3

Gareth Edwards | www.virtualisedfruit.co.uk | @GarethEdwards86

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

TABLE OF CONTENTS

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

DOCUMENT CONTROL SHEET

Change Control

Customer Name	Virtual Design Master Challenge, Season 5	
Document Title	Challenge 2	
Version	V1.0	
Document Reference	Challenge 2 - Attack of the Zombie crypto.docx	
Project Reference	VDM Challenge 2	
Date of Creation	27 June 2017	
Date of Last issue	18 July 2017	
Author	Gareth Edwards	@GarethEdwards86

Distribution

Name	Position
Creative Panel	
Eric Wright	Creative Team
Melissa Wright	Creative Team / Deliverer of Evil??
Angelo Luciani	Creative Team
Judges	
Rebecca Fitzhugh	VCDX 243
Byron Schaller	VCDX 231
Lior Kamrat	VCDX 230

Version Control

Version	Description of Change

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

0.1	Initial Creation
0.2	Final Revisions
1.0	Release to the world

Associated Documents

Title	Date	Source	Version
Season 4 – Challenge 1 – Back to Earth	01/07/2016	GitHub	1
Season 4 – Challenge 4 – GitHub			

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Abbreviations

Abbreviations	Description
VPN	Virtual Private Network
VDM	Virtual Design Master
AI	Artificial Intelligence
DNS	Domain Name Services
DHCP	Dynamic Host Configuration Protocol
SMB	Server Message Block
DFS-R	Distributed File System Replication
AWS	Amazon Web Services
P2P	Point to Point
VM	Virtual Machine
DB	Database

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

SYNOPSIS

The designs you have been working on through the first two challenges of this season have proven to solve many of the challenges faced by the Humanity Recovery Teams. We now have three sites running the HumanityLink software, and they have been secured. However, three sites will not be enough for long, as we begin to take more and more of the Earth back.

Some of the future sites slated for deployment have not yet been declared Zombie Free Zones, and are still considered dangerous. In addition, there are additional sites off-Earth being looked at for deployment. It is not always feasible to send a human team to do perform an infrastructure deployment for these reasons, and many others. We are also very short staffed, so the ability to perform centralized remote operations is also desired. Being able to deploy and manage the infrastructure and application in this manner is the next goal.

You must design and an environment to provision and manage your infrastructure. HINT: You may want to use open source products, but this is not a requirement or constraint. The design must include currently available products, it cannot utilize non-released features.

Choose and document how you will deploy and manage your infrastructure from the initial provisioning through to the continuous configuration management. Your submission must also include operations procedures for managing version control for your infrastructure and code. Every layer of your infrastructure from the cloud and virtualization layer up to the application must be included.

Practical walkthroughs and example code are valuable add-ons to your design document

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

INTRODUCTION

Overview

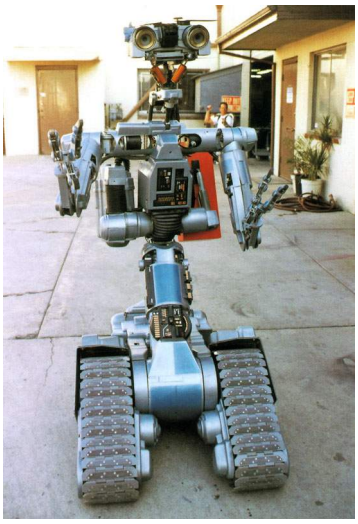
So far all we know the Zombies are back and we need to start pushing on with our effort. Not all zones are declared safe so we need to ensure we have some autonomy so we can scale quickly but we may not be able to send in humans to build the sites.

Intended Audience

This document is intended for the design board (our judges) to help make key decisions on implementing our new infrastructure.

Project Summary

In this document we are going to cover how we will deploy our systems remotely or prebuilt on a live site and then implemented by our robots. We need a way of making this as autonomous as possible. There is also a possibility that some of these sites may not be on earth. I am also starting to wonder what the robots look like or are they different types. This guy seemed pretty capable so maybe we can put him on datacentre builds



Project Requirements

The requirements of this project are to provide the design board with the following.

- Ability for centralised management
- Deployment of infrastructure with non human teams
- The design must include currently available products, it cannot utilize non-released features

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Project Assumptions

These are the assumptions we have made of this project is to provide the design board with the following.

- We can use open source software
- Some of the infrastructure can be pre built on site prior to robot installation
- Mobile signal is available in all of the locations
- Satellite is available in some area
- Cloud datacentres are not always available in some of the locations desired
- MAC Addresses can be provided by supplier in a simple file

Project Constraints

The constraints of the project are outlined below.

- We still need to adhere to known data limits for offsite replication and latency

Project Risks

The risks of this project are outlined below.

- The robots may fail when installing some of the assets
- Weather may affect uplinks

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

LOGICAL DESIGN SUMMARY

Application Delivery

In our previous design we had certain failings that allowed for an SMB attack so that we could defend some of our design decisions. We have since relooked this this and decided to work with the engineers to come up with a better design. As suggested before the application and servers will still be deployed via (Terraform) but we will look at using NGNIX and also MongoDB for our web and database services. All code will be stored via GitHub to ensure change control as provide in an AWS Example by **clstokes** in his article (Deploy a Complex Infrastructure in AWS, 2016). As before have decided to get our engineers to build out this code for our deployment so it also performs actions to our VMware environment and issues VMs to AWS if the sites allow.

Infrastructure Delivery

We have decided that the infrastructure will be controlled and managed by a CMDB database file held within GitHub. There are several reasons for this the first being we can use this as a central store regardless of what platform the system is built on and only append the variables required. Secondly it allows any of the engineers to track any changes and who did what. This has mainly spawned from my build process for the physical datacentres following this article (ESXi scripted installation PXE PowerCLI) by Christophe Calvet. I knew I wanted to be able to connect my chassis to a build network for them to then auto boot and be built. An example of the CSV file can be found here https://github.com/neoof86/vdm-season5-ep3/blob/master/UMRKCMDB.csv. We also were lucky that GitHub still had some of our old code that automatically created the vCenter https://github.com/discoposse/vdmS401/blob/master/powershellconfigenviroment.txt

As many of the commands for ESXi and vCenter can be referenced within PowerShell we can use this to fill in any gaps that Terraform cannot function.

Networking

The networking will be delivered over several mediums. Our main data centres will still be connected by MPLS links with VPN as backups. Any new remote site will be delivered via satellite where possible with a backup of cellular data connections until such physical connections can be put in place. The only exception to this would be an orbiting datacentre on the moon or in the international space station. Where possible it would be nice to implement NSX for the management tiers but as the application and database tiers can easily be thrown away and started again this is not essential. This does mean there will be a high reliance on DNS.

As the 3 main sites are geographically spread out and as further ones come online the land communications will create a mesh to uplink to the ISS to allow for a small DC to be run there.

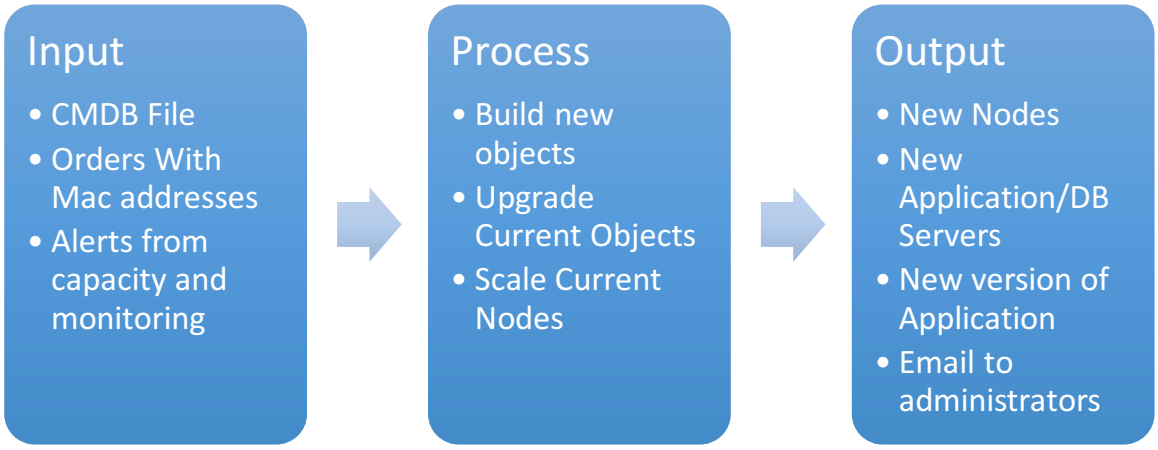
CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Code Flow

At a very high level we expect the following to occur



CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

PHYSICAL DESIGN SUMMARY

Sites

With this scenario we are assuming that the break out has occurred at our physical datacentres of which we control being the following

- Primary Site:- Mojave Desert (Back to Earth, 2016)
- Secondary Site:- NGD, Wales, United Kingdom
- Tertiary Site:- Sydney
- Quaternary Site:- International Space Station
- Quinary Site(s):- AWS Oregon, N.California, Ohio

Virtual Infrastructure

I have outlined what I would use for the infrastructure below, some site we may also opt for a C3000 instead if this wanted to be a small scale site.

- Primary Build:-
    - 1 x C7000 Blade Enclosures (10U)
    - 8 x HPE ProLiant BL460c Gen9
    - 1 x Primary All Flash Nimble Array
    - 4 Node Rubrik Cluster
    - 1 x Automated Tape Library
    - 2 x VCenter Installation in Linked Mode
    - 2 x Meraki MX 600
    - 2 x HP 5406R
- Secondary Builds:-
    - 1 x C7000 Blade Enclosures
    - 8 x HPE ProLiant BL460c Gen9
    - 1 x Primary All Flash Nimble Array
    - 4 Node Cohesity Cluster
    - 1 x Automated Tape Library
    - 2 x VCenter Installation in Linked Mode
    - 2 x Sophos XG 750
    - 2 x Cisco Nexus Switches

CHALLENGE 3

Virtual Design Master

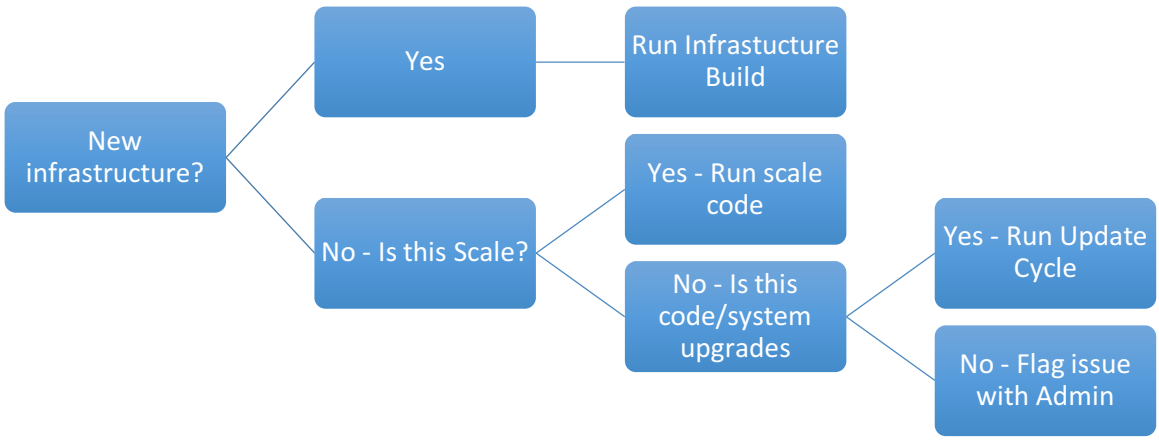
AUTOMATING IT ALL AND KEEPING TRACK OF IT

Physical Transportation

In order to allow for the sites to be prebuilt and then potentially be installed by robots with have decided to use some of the following components. The main chassis will be (Amazon Racks) and these are stackable shown here (CPCases, Interstacker and Wheelboard). We are hoping the robots have enough dexterity to connect some simple cables and move cellular links of satellites. These cases are durable and IP rates to resist any attack from the elements as they are approved product for government and MoD / DoD requirements. We have also been able to tackling any cooling issues if any server rooms need to be built by adding the (Cool Collar) of which also provides further space in the cases for equipment.

Code Cycle

The cycles show what we would have done with the code permitting we had the time to fully write this



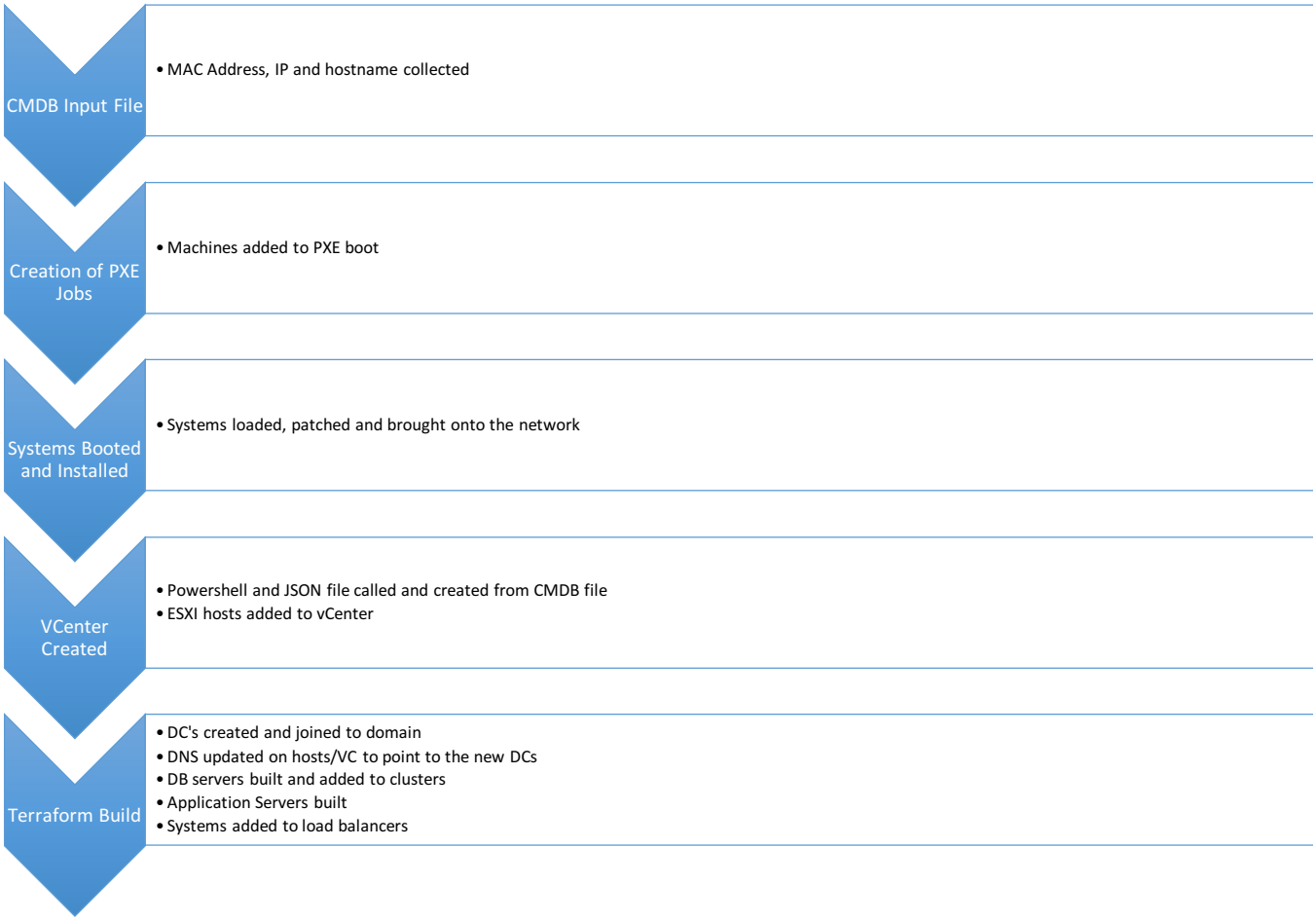
CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Infrastructure Builds

The below shows how new infrastructure would be built from scratch when placed into the build labs. Some elements are very high level as networking and routing need to be added where possible via further APIs or scripts.



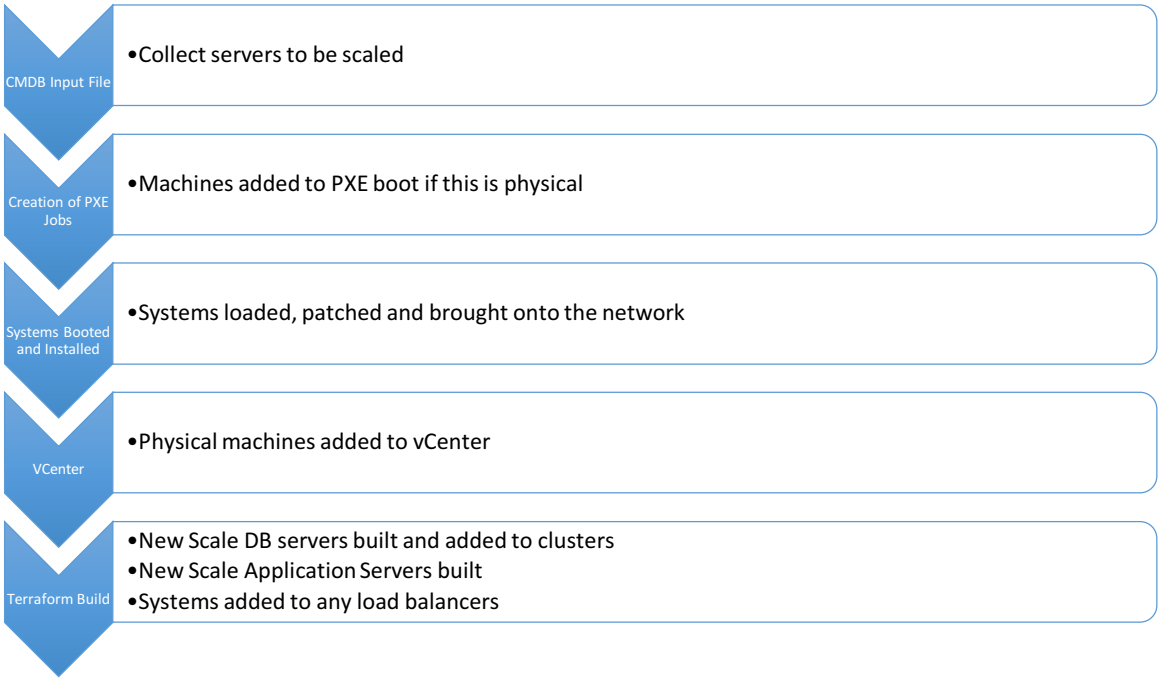
CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Scale Builds

The below codes build upon the infrastructure code and just adds additional parts to the required infrastructure.



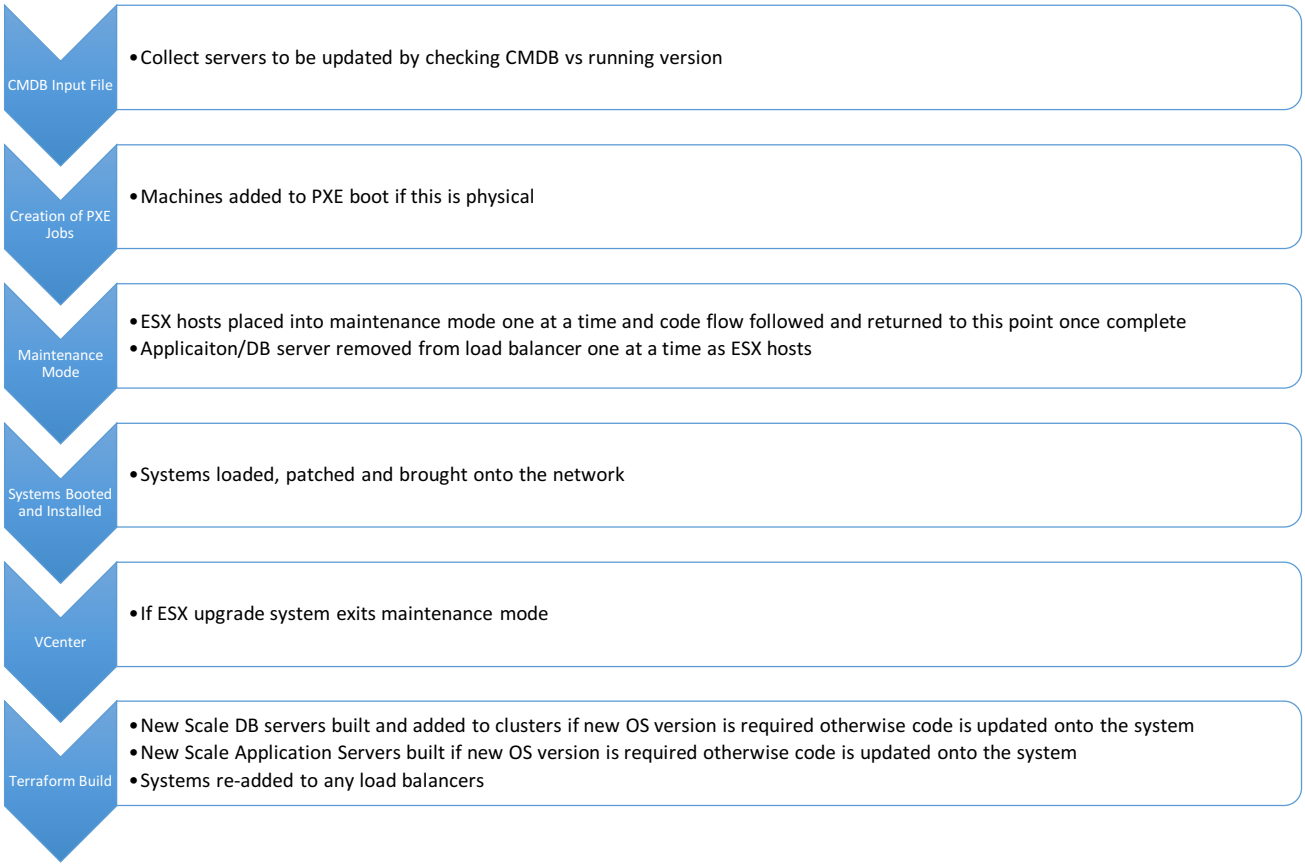
CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

Code/Patch

This code is essentially loops until finished where it will build a list of new components and place them into maintenance modes until the patching or updates are completed. This may be called from GitHub for example upon a code push to trigger terraform to commence with the builds. The code push may in itself need to update the CSV files to trigger version numbers.



CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

WHAT WOULD I HAVE DONE DIFFERENT

I had some good ideas for this week but I was unable to put any into practise due to limited knowledge and lab ability. It has provided me with a new passion for infrastructure as code with the view of rebuilding my home lab this way so I can get a true understanding of this. It would have also been nice to deploy the simple web app code and DB to fully understand their elements as I started to assume way too much.I also want to look further into long range communications and get down to the nitty gritty with TTLs and jitter. I also think I need to get a better appreciation for other tools like Ansible, Puppet, GitHub and Terraform to find a complete way to lifecycle the infrastructure and applications so each can cover all areas. I also want to look into monitoring and capacity tools such as SexiLog to see if we can then get them to trigger actions like scale VMs.

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

WORKS CITED

Calvet, C. (n.d.). *ESXi scripted installation PXE PowerCLI*. Retrieved from ESXi scripted installation PXE PowerCLI: http://thecrazyconsultant.com/esxi-scripted-installation-pxe-powercli/

clstokes. (2016, Aug 24). *Deploy a Complex Infrastructure in AWS*. Retrieved from Deploy a Complex Infrastructure in AWS: https://github.com/hashicorp/atlas-examples/tree/master/infrastructures/terraform/projects/01

CPCases. (2017). *Cool Collar*. Retrieved from Cool Collar: https://cpcases.com/products/climate-control/cool-collar/

CPCases. (2017). *Interstacker and Wheelboard*. Retrieved from Interstacker and Wheelboard: https://cpcases.com/products/19-racks/amazon-racks/interstacker-and-wheelboard/

CPCases. (n.d.). *Amazon Racks*. Retrieved from Amazon Racks: https://cpcases.com/products/19-racks/amazon-racks/

Edwards, G. (2016, July 01). *Back to Earth.* Retrieved from GitHub: Mojave Desert

GitHub. (n.d.). *GitHub HA*. Retrieved from GitHub Enterprise: https://help.github.com/enterprise/2.10/admin/guides/installation/high-availability-configuration/

Terraform. (n.d.). *Terraform*. Retrieved from https://www.terraform.io/#create: https://www.terraform.io/#create

CHALLENGE 3

Virtual Design Master

AUTOMATING IT ALL AND KEEPING TRACK OF IT

DISCLAIMER

The view expressed in this document are my own and do not necessarily reflect the views of my current, previous or future employer(s). This is a fictional design and some elements may not work correctly within your infrastructure. All data and information provided on this document is for informational purposes only. I make no representations as to accuracy, completeness, currentness, suitability, or validity of any information throughout the document & will not be liable for any errors, omissions, or delays in this information or any losses, injuries, or damages arising from its display or use. All information is provided on an as-is basis.

Covers were taken from the following places:-

https://pixabay.com/en/network-server-system-2402637/

https://pixabay.com/en/cloud-computing-cloud-device-data-1989339/

CMDB, Tracking & Management is key

Virtual Design Master Season 5 - Challenge 3

Gareth Edwards | @GarethEdwards86

www.virtualisedfruit.co.uk