Azure CLI

About me

- Justin VanWinkle
- Very early hire of Switchboard Live
  - https://switchboard.live
  - Theoretical Title: Doer of What Needs to be Done
  - Actual Titles:
    - Software Developer
    - Interim Infrastructure Lead
- Lover of containerization

Switchboard & Azure

- Joined Switchboard when infrastructure was very young
  - All apps on Azure App Services
    - Great for early stages!
    - Our specific needs required something more

- Set up our automations
  - Builds
  - Deployments <- Azure CLI simplifies
  - Tasks <- these steps

- Revamped critical parts of infrastructure
  - Dynamic Scalability <-
  - Self healing <- Made possible by Azure
  - Regional redundancies & failovers <-

Overview

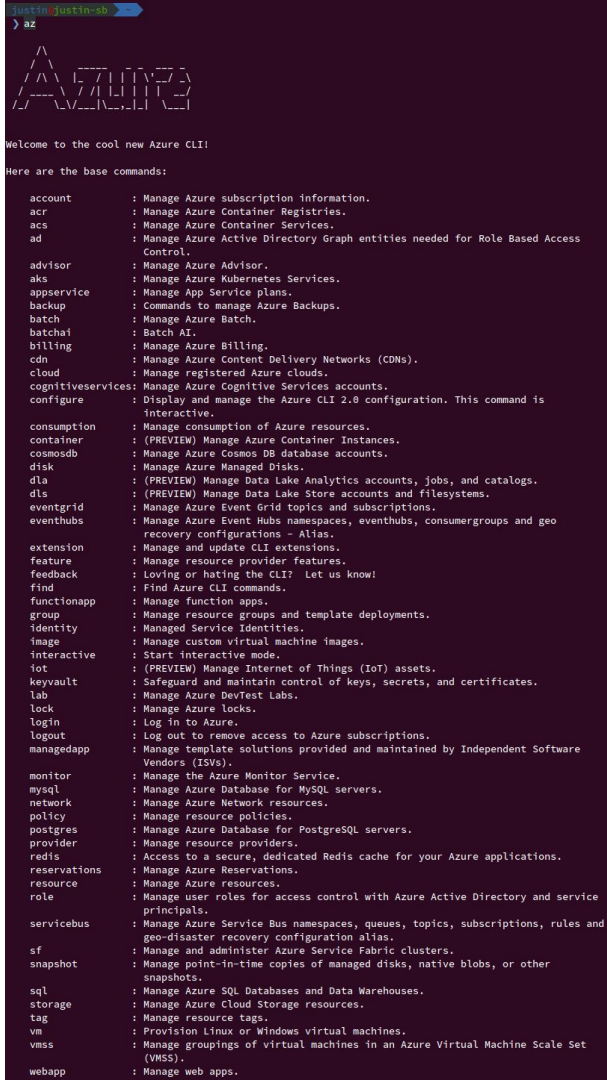
- **Azure CLI**
  - **How to navigate**
  - **Deploying singletons**
  - **Deploying entire infrastructure**
    - **Azure Resource Manager Templates**
    - **Template Deployments**
    - **Post-deployment configuration with Cloud-Init**
- **Using Make for sanity**
- **Demo - Docker Swarm on Azure**

Azure CLI

- Great tool for managing resources on Azure.
- Performs actions via Azure APIs.
- Contains functionality Azure Portal lacks.

Availability:

- Windows, MacOS, Linux
- Docker
- Azure Cloud Shell



Azure CLI - Jam Packed!

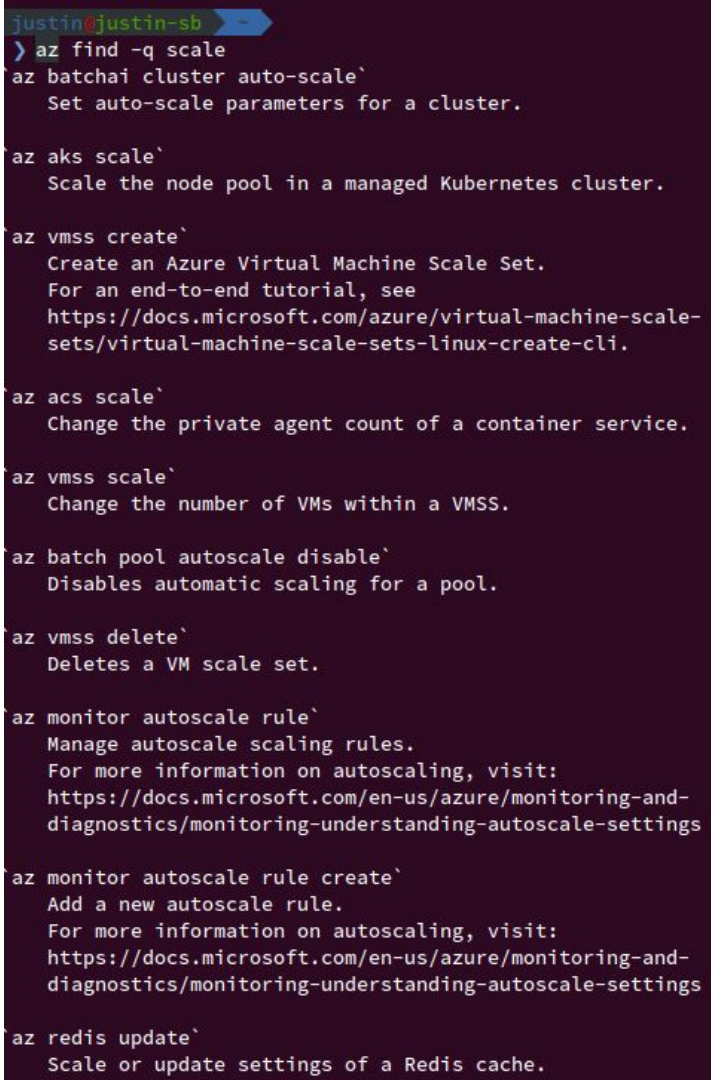
```
feedback            : Loving or hating the CLI?  Let us know!
find                : Find Azure CLI commands.
```

How often do you see these in a CLI tool?

Azure CLI - Find

A powerful tool

- Returns relevant commands
- Great descriptions



Azure CLI - Deploying a Resource

Deploying a resource is simple:

```
justin@justin-sb ~
❯ az vm create -n MyLinuxVM -g MyResourceGroup --image UbuntuLTS
```

Great for quick one-off deployments.

Azure CLI - Deploying Multiple Resources

Problem: “az <product> create ...”

- **Creates 1 item at a time**
- **Configuration commonly required after deploy**

Solution: “az group deployment create ..."

- **Deploy entire infrastructure in single command**
- **Post-deploy configuration not necessary**

```
az group deployment create \
    --template-file ../templates/cluster.json \
    --parameters @parameters/cluster.json \
    --resource-group $(RESOURCE_GROUP) \
    --name cli-$(LOCATION) \
    --output table \
```

Azure CLI - Templates

- Version Controllable
- Highly Configurable
- Produces consistent duplications of infrastructure

```
{
    "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "",
    "parameters": {  },
    "variables": {  },
    "resources": [  ],
    "outputs": {  }
}
```

Azure CLI - Parameters

- Data that is variable between deployments
- Sensitive data
- Anything you wouldn't want to version
- Dynamically generated
  - I use a python script to generate this
  - There are OSS alternatives

What is "managerCustomData"?

- Base-64 encoded Cloud-Init Config
- Post-Deployment Configuration Script
- Cloud-Init (http://cloud-init.org)

Azure CLI + Cloud Init

Cloud-Init performs post-deploy configuration actions

- You specify configuration
- Azure passes Cloud-Init configs to new VM
- When you scale, each new VM runs Cloud-Init config
- VM is up and ready to serve in minutes - identical to the last

http://cloud-init.org

Azure CLI + Make

AZ CLI is Often verbose

Makefiles to the rescue!

- Reduce ceremony
- Use in CI/CD pipeline
- Version Controllable

There are alternatives if you prefer

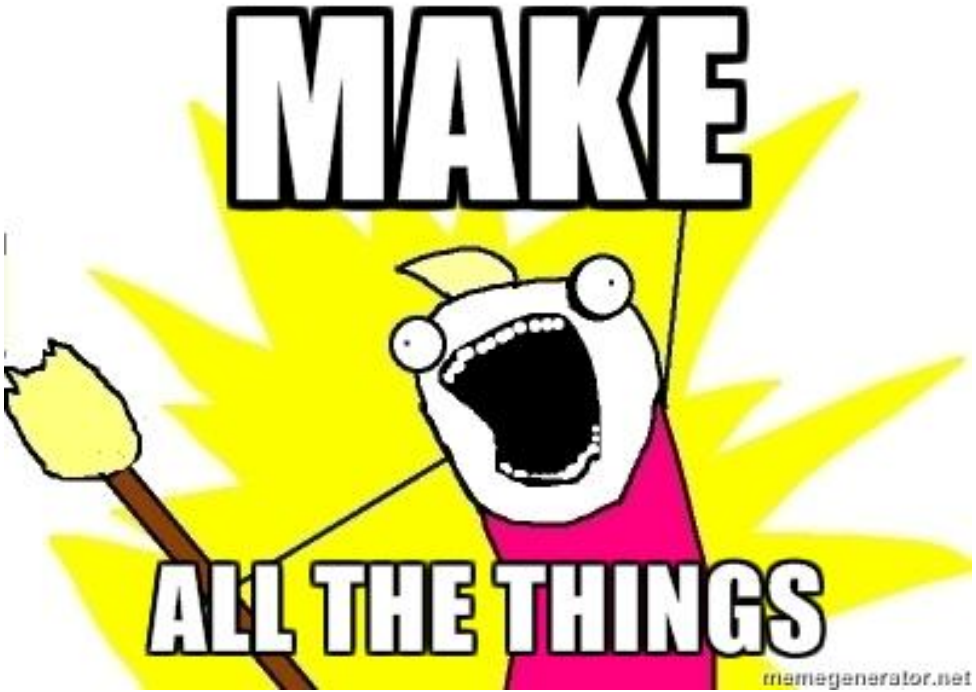
```
chaos-monkey:
    az vmss list-instances \
        --resource-group $(RESOURCE_GROUP) \
        --name $(VMSS_NAME) \
        --query [].instanceId \
        --output tsv \
    | shuf \
    | xargs -I{} az vmss restart \
        --resource-group $(RESOURCE_GROUP) \
        --name $(VMSS_NAME) \
        --instance-id {} \
        --output table

list-endpoints:
    az network public-ip list \
        --resource-group $(RESOURCE_GROUP) \
        --query '[].{dnsSettings:dnsSettings.fqdn}' \
        --output table
```

Azure CLI + Make

+1 for Make - Tab autocompletion!

Recommendation:



Azure CLI - Demo

- Deploy Application in Docker Swarm on Azure under 10 minutes.
  - Deploy infrastructure
  - Set up Swarm visualizer (A management tool)
  - Deploy replicated application

Based on https://github.com/rcarmo/azure-docker-swarm-cluster

Demo

Questions?

Twitter: @jclayvan

Email: justin@switchboard.live

Switchboard Live: https://switchboard.live