Condo Daily Report 9-21-21

Tuesday

* AWS Module

* IAC
  + Starting
    - Set up build managing and version control
      * This is important because it ensures that there will be no version conflicts, and the code will have to build correctly before being pushed
  + Idempotency
    - The act of running IAC multiple times and resulting with the same build
      * EX
        + Two different templates are created
        + They both create ec2 instances
        + One of them generates a random string for a name and the other has a consistent name
        + The one with the consistent name practices idempotency

Idempotency is important because when the ec2 instance with the same name builds multiple times, the ec2 state  will be saved, but the random generated ec2 instance will be torn down and rebuilt meaning the state will not be the same upon recreation meaning the code is not idempotent

* + Modularity
    - Reusability
      * The ability to reuse code in a plug an play sense
        + This can be seen in nested stacks
    - Scalability
      * Being able to go back and add onto code easily
        + Loose Coupling
    - An example of this is the process that i am currently doing with our bearStack
  + MultiStage
    - Your software can be observed and tested throughout each stage in its life cycle
  + Automated
  + Hybrid model
    - The IAC can be used to reach public cloud, private cloud, on premises… etc.
  + Playbooks
    - Playbooks are sequences of tasks
      * Uses some module that will then use that module to execute a task in a target system
    - Ansible
      * Scale is a problem
      * If rollback occurs all created resources remain which is trouble
      * Use ansible when you want to automate or configure a multitude of systems at once with little initial effort
      * Uses ssh connections going out
    - Chef
      * Uses client server arch
      * Install agent for servers u want to connect
      * Uses cook books and recipes for each node
      * Used for configuration management
        + High level languages
        + Ruby
      * Custom DSL
        + Looks a lot like ruby
    - Puppet
      * Similar to chef
      * More declarative rather than chef being imperative
      * Puppets syntax is super easy to read and understand
      * Arch is client server like chef
      * Community
        + Puppet forge
    - Terraform
      * Very similar to aws cloudformation
      * Declarative
      * HCL language
    - Pulumi
      * See state
      * Uses high level languages
        + Python
        + Go
        + Etc.
  + IAC Challenges
    - Facing the fact that you have to work around the existing infrastructure that is in place
      * Cant just make new and better IAC without taking in consideration resources that are already in place
    - Module based programming
      * It can be dangerous if modules are not inspected and understood before use
        + This can be a huge security risk
        + Dont take the shortcuts, instead inspect the code
* GitOps
  + 4 key points
    - Desired state declared in code
    - Software agents continuously reconcile
    - Desired state is versioned and immutable
    - Only operate using these principle
* CloudFormation
  + Bearstack
    - Ran into issue with all stacks loading at the same time
    - Had to find a way to defeat this issue
      * Looked up stack dependencies
        + If stacks are referenced eachother, the referenced stack deploys before the other stacks
        + This would be helpful if the stacks were referenced
      * After research I found a clause that forces resources to create in a specified order
        + The clause name is DependsOn:

In the clause you state the resource that the current resource depends on

* + - This defeated the issue, but then got reference problems passing values to the stacks
* Keeper Meeting
  + SSO for keeper from azure AD
    - Kayce and I are admins so we had to take turns moving each other over
    - We did so and enabled sso for keeper
    - We have a small user pool of around 4-6 people designated for the tech team