Condo Daily Report 9-7-21

Wednesday

* Okta
  + Tried to connect okta to snowflake
    - Completed all steps on okta side
      * Set subdomain to <organizationName>-<accountName>
        + maa23857-main
      * set sign-on to saml2
      * set the application username format to email prefix : [scondo@bearcognition.com](mailto:scondo@bearcognition.com)
        + would be scondo
    - Completed all steps on the Snowflake side
      * Performed the saml\_identity\_provider query
      * Performed the create security integration oktaintegration query
      * This came with no success and a 400 error
      * Tried changing the SSO URL, and it resulted in a 403 error
      * Looked on several forums and found that certain aspects of the query needed to be capital even though the documentation said the opposite
        + Some components weren’t put in quotes either
      * This came with getting to the sign in page with the error of
        + SAML response is invalid or matching user is not found. Contact your local system administrator.
  + Went on call with Kayce
    - Discussed possible solutions
      * Talked about switching from okta
      * Discussed switching okta provisioning
      * Mike said the highest priority is Paylocity
        + Thinking about using Paylocity to provision accounts
      * Decided we can’t drop okta until 2023
      * Contacted okta support to fix problem
* Perceptivity API change
  + The perceptivity API changed taking down all the connectors
    - Went on call with Kayce to discuss action
      * Kayce got another API key and sent it to Teknion where they resolved the situation
    - I asked Kayce to learn how to create connectors and solve such a problem if it comes back again
      * He walked me through how to create connectors in postman
      * Made notes and created a practice connector card
* CloudFormation
  + Worked on the bearAppStack template file
    - Found a possible solution that uses the storage units as plug and play to take the old EBS from the other servers and just pop it into the servers in the new stack
    - Made some parameters to allow cross-stack referencing
    - Added ec2 as a resource
      * Started configuring all inner components that were necessary to remake the bear cognition environment
      * Stripped some components
      * Ran into trouble with some referencing
      * Did some debugging, and found that cloud formation debugging was tough because it doesn’t tell you where the break happens
        + They should make a cloud formation ide
      * Found that the reference isn’t the actual name of the output but the reference name within the output
      * Made the output names consistent with the reference names and got another bug
      * This bug was for VolumeId
      * The first bug was the fact that it was a case sensitive bug
      * The second bug was that the volume that was being targeted wasn’t in the same availability zone as the ec2 that was being made in the stack
      * Made a snapshot and a copy volume of the original targeted volume
      * The stack launched and the ec2 was made with the volume appended to it
      * Problem was there was another EBS volume being treated as root
        + Learned that there is a specific param I need to set to make sure no other volumes are made
        + also set the EBS path as /dev/sda1 which enforces root
      * accidentally appended portal security groups instead of tableau security groups
      * switched them to open port 22 for SSH to see if ruby was installed on the new instance
        + if it is then my hypothesis will be correct in thinking we can just clone the EBS’s from the bear cognition environment and spin up identical instances
  + bearAppStack template file : [CLICK ME](https://github.com/condosr/CloudFormation-YAML-Scripts/blob/main/perceptivity/bear_apps.yaml)