Case Study

MIROCULUS

Miroculus delivers an affordable microfluidics platform, which enables seamless workflow integration from novel molecular protocols to data analytics.

Cloud Challenge

Miroculus were looking to develop a new version of their service to cope with an increase in workload.

Their existing AWS setup **was not optimized to scale**, so nubeGo were enlisted to help design and implement a multi-account structure. This would provide the basis for Miroculus to deploy an ECS cluster to run their new workload on and then eventually to use a combination of EMR, Sagemaker and Neptune for data processing.

During a period of rapid growth for both companies, **nubeGo helped mitigate issues** by keeping in regular contact to ensure that all parties had a clear understanding.

> “Knowing Fernando and his expertise, we decided to work with nubeGo. They reacted in a professional way, hearing us first and then implementing the solutions”
>
> Matias De Carli, Software Manager & Technical Leader

nubeGo Engagement

Miroculus had a plan of how they wanted to rebuild their application architecture in AWS for scale and nubeGo were able to help assist in reshaping and redesigning their target architecture to adopt the right compute tools for the job.

Working closely with AWS Solution Architects, **nubeGo reviewed and advised on target network and compute architecture.** In creating a new account with AWS, a limit was reached which prevented the creation of any further accounts. This required the nubeGO team to raise and escalate a request with AWS to increase the limit while ensuring the team at Miroculus up to date.

A new protocol emerged as a result; to raise a request with AWS at the earliest opportunity to reduce the impact of the restriction.

Realised Benefits

nubeGo presented a **new account structure with AWS Organizations** that included a master billing account as the main account and then accounts for each environment (Development, Staging and Production). It also included accounts for backups, logging and finally, a shared services account that would be used for hosting services including CI/CD tools.

For access control, IAM roles were defined for each account which would then link into a SAML integration with G Suite so that access to AWS resources could be managed centrally.

Finally, a 1 VPC to 1 Account structure was proposed with multiple subnets provisioned in each VPC.

Tools Used

Terraform was used to code the deployment of all the resources within AWS from accounts down to the subnets in each VPC. Using Terraform provides the ability to scale out infrastructure in a repeatable manner.

The client was most impressed by the introduction of Terraform templates and told the team that they worked very well from the first day. He said “The whole process has been great” and it had been of benefit to them by installing “a solid base to build or business”.

> “Professionalism, technically really strong, and the ease to interact with nubeGO members”
>
> Matias De Carli, Software Manager & Technical Leader

nubeGo

www.nubego.io