Declan Morris

declanmorris.com

Personal Statement

I am an IT Cloud Infrastructure Engineer with a fervent dedication to continuous learning, both within my professional role and during my personal time. As a self-taught developer, I have immersed myself in the ever-evolving landscape of the industry, recognizing the paramount importance of self-education in a field characterized by rapid technological advancements and evolving work methodologies.

My enthusiasm is particularly directed towards infrastructure as code, where I am committed to championing automation and repeatability in all my projects. I firmly believe that harnessing the power of automation not only enhances efficiency but also ensures the consistency and reliability of every solution I create.

Work Experience

Drax Group **Selby, England**

IT Cloud Infrastructure Engineer *2022 – Present*

- Standardized Linux builds to CIS Lvl 2 Hardened levels using Packer and Ansible, enhancing security and compliance.
- Developed Terraform code to automate and streamline AWS infrastructure upgrades, transforming manual tasks into a one-click pipeline.
- Designed Terraform Modules to simplify machine provisioning, empowering other teams to self-service while adhering to guardrails.
- Implemented a CI process for static code analysis to ensure standardized and secure infrastructure code practices.
- Authored Terraform and Ansible scripts for the configuration and hardening of ESXi hosts, including the creation of a vCenter, compute cluster, vSAN, switch, port groups, and VNICS from scratch.
- Engineered code to deploy an OpenShift container platform on vSphere, leveraging Afterburner and Ignition to achieve rapid platform creation.

Work Experience

Drax Group **Selby, England**

Software Engineer *2019 – 2022*

- Developed, deployed, configured, and maintained numerous mission-critical microservices, ensuring adherence to industry best practices.
- Created and deployed an Azure Function API and associated cloud resources using Terraform Infrastructure as Code (IAC), including PAAS MSSQL, VNET, and Linux VMs configured with Ansible.
- Led the upgrade of the plant metering system, along with all related services, databases, and queries, establishing a resilient system capable of real-time data retrieval, 24/7/365.
- Designed and implemented secure APIs to seamlessly integrate external services with HR data, prioritizing data security and confidentiality.
- Spearheaded the adoption of standardized CI/CD pipelines for all projects and guided the team in transitioning to YAML-based CD pipelines.

Work Experience

StarCompliance **York, England**
Business Analyst *2016 – 2018*

- Developed a new workflow that involved close collaboration with Chief Business Officers to comprehensively understand their needs. Designed a Perl script to automatically evaluate over 800 product backlog enhancements based on predefined criteria, assigning scores for efficient review by Chief Business Officers.
- Designed, documented, and delivered software specifications, including detailed requirements and state diagrams, facilitating effective communication between development and business teams.
- Collaborated closely with developers, quality assurance professionals, and client teams to gather research and functionality requirements, fostering a collaborative and cross-functional work environment.
- Assisted the technical project manager in transitioning from a Waterfall development methodology to a more agile approach, enabling more adaptive and responsive project management.

Personal Projects

Personal Website: I have my own personal site for putting out articles on things I've found interesting. The whole deployment is IAC, and the site gets updated on pushes to main.

- Deployment of the infrastructure using terraform for the VM deployment and DNS.
- Configuration of webserver using ansible.
- Website coded using Hugo and Markdown for creating a personal blog.

Proxmox Home Server: I have a custom-built home server running Proxmox as a hypervisor. I follow IAC for this, and the LXC containers and any VMs running on the server are created via Terraform. All software and configuration are deployed and managed via Ansible. I open source all of this code in my personal infrastructure repo, which can be found here.

Education

Department of Chemistry, The University of York **York, England**
Master of Chemistry with a Year in Industry, 2:1 Honours Degree *2010 – 2014*

Cardinal Newman College **Preston, England**
2008 – 2010

A Level	
Chemistry	A
Maths	A
Biology	B

As Level	
Computing	C
General Studies	C

Languages

English: Fluent *Mother tongue*

Terraform: Intermediate

Ansible: Intermediate

Python: Basic

Packer: Basic

References

References are available on request