GASHLEY DENEAU

Cloud explorer

Thebestprogrammer12@gmail.com (617-729-6334 95 Morton village Dr

Objective

A highly motivated and detail-oriented cloud explorer with strong foundational knowledge in cloud technologies, introduction to Python programming for cloud , Linux, and Bash scripting. Seeking an entry-level Cloud Engineer position where I can apply my skills in Microsoft Azure, cloud computing, and networking to drive operational efficiencies and contribute to the success of a growing team.

Professional Experience

Cloud Intern at DATACOM(virtual internship) 2025 – Present

Assisted in the deployment of Azure virtual machines, configuring Azure networking (V Nets, Subnets), and creating storage accounts.
Wrote Python scripts for automating Azure resource deployments and configuration management.
Assisted the senior team in troubleshooting and optimizing Azure environment performance.
Created Bash scripts for automating system tasks on Linux servers such as log file management, user provisioning, and system health checks.
Participated in weekly meetings to review cloud security practices and improve system availability using Azure Monitor.

Education

Bachelor of Science in CYBER SECURITY [QUINCY COLLEGE Present

Relevant Coursework: Cloud Computing (Microsoft azure , AWS,Networking Fundamentals, SUFFOLK UNIVERSITY (Cloud Systems Administration program)
Operating Systems, LINUX, BASH , Powershell

Technical Skills

Cloud Platforms
AWS ,Microsoft Azure (Azure Virtual Machines, Azure Storage, Azure Networking, Azure Active Directory, etc.)

Operating Systems
Linux (Ubuntu, CentOS, Debian), Windows

Tools & Technologies
Git, Docker (basic), Kubernetes (basic), Terraform (basic), Ansible (basic)

Monitoring & Security
Azure Monitor, Azure Security Center, basic knowledge of firewalls & VPN configurations

Programming/Scripting
Python, Bash scripting, PowerShell

Networking
TCP/IP, Subnetting, Routing & Switching, DNS, DHCP, VPN, Load Balancers

Databases
MySQL, PostgreSQL (basic knowledge)

Version Control
Git, GitHub, GitLab

Additional Information

Languages
Spanish (fluent),
French,English,German,creole

Availability
Immediate

Projects

Project 1: Automated Virtual Machine Deployment on Azure
Technologies Used: Microsoft Azure, Python, Azure SDK, Azure CLI, Bash
Created an automated Python script using Azure SDK to deploy and configure multiple virtual machines on Azure.
Integrated Bash scripts to configure the VM after deployment (e.g., installing necessary software, configuring system settings).
Used Azure CLI for checking the status and resource management.
AWS
✅ Core AWS Services
EC2 – Launching, configuring, and managing virtual servers
S3 – Object storage for backups, static websites, and data archiving
VPC – Networking setup including subnets, route tables, and security groups
IAM – Identity and access management for secure resource access
Infrastructure as Code / Automation
CloudFormation – Infrastructure provisioning using templates
Terraform (with AWS provider) – Popular third-party IAC tool
AWS CLI & SDKs (Python/Boto3) – Command-line and programmatic automation
AWS Systems Manager (SSM) – For patching, remote command execution
🔄 Deployment & CI/CD
CodePipeline – Automating release pipelines
CodeBuild / CodeDeploy – Build and deploy automation
Elastic Beanstalk – Simplified app deployment
CloudWatch – Monitoring logs, metrics, and setting alarms
☁️ Compute & Containers
ECS / Fargate – Container orchestration and serverless containers
Lambda – Serverless compute
Auto Scaling Groups – Scaling EC2 instances based on demand
Elastic Load Balancer (ELB) – Distributing traffic
🗃️ Databases & Messaging
RDS (e.g., MySQL, PostgreSQL) – Managed relational databases
DynamoDB – Serverless NoSQL database
SQS / SNS – Messaging and notification services
🛡️ Security & Management
AWS Config – Resource compliance tracking
CloudTrail – Governance, compliance, and auditing
AWS KMS – Key management for encryptionCloud Platforms: AWS (EC2, S3, IAM, VPC, Lambda, CloudFormation, CloudWatch, RDS, ECS, CodePipeline, SQS, SNS) Automation & IAC: Terraform, AWS CLI, Boto3, Python (for scripting), PowerShell

Project 2: Network Traffic Monitoring System using Python & Linux
Technologies Used: Python, Linux (Ubuntu), Bash, Network Protocols (TCP/IP)
Developed a Python-based system to monitor and log network traffic on a Linux server.

Wrote custom Bash scripts to automate the collection and reporting of network statistics, including IP address tracking and port activity.
Implemented basic network security measures by monitoring for unusual traffic patterns and sending alerts.

Project 3: DNS and DHCP Automation on Linux Server
Technologies Used: Linux (Ubuntu), Bash Scripting, Networking
Configured and automated DNS and DHCP services on a Linux server for a small office network.
Wrote Bash scripts to automate the assignment of IP addresses and the configuration of DNS records.
Successfully implemented a basic network security solution using firewall rules and regular updates to services.

Certifications

Microsoft Certified: Azure Fundamentals (AZ-104)
Microsoft Azure (AZ-10) – Hands-on Experience
Completed multiple projects aligned with AZ-10: Microsoft Azure Administrator certification objectives
Gained practical experience in Azure services such as Virtual Machines, Networking, Resource Management, and Monitoring
Actively preparing for the official AZ-10 certification exam.

Personal Projects:

Hands-on Cloud Engineering Projects | Personal Lab Environment (Stackroute.com) Suffolk university

1. AWS Infrastructure as Code with Terraform
Designed and deployed a highly available 3-tier web application on AWS using Terraform
Provisioned resources including VPC, EC2 instances, RDS, ALB, Auto Scaling Groups
Implemented infrastructure version control with Git, and monitored performance using CloudWatch

2- Azure Virtual Network and VM Automation
Built a secure network architecture using Azure Virtual Network, subnets, NSGs, and route tables
Automated provisioning of Windows/Linux VMs using Azure Resource Manager (ARM) templates and Azure CLI
Used Azure Bastion and Just-in-Time VM Access for secure remote management

3. Serverless Image Resizer with AWS Lambda
Developed a serverless image processing pipeline using S3, Lambda, API Gateway, and DynamoDB
Triggered automatic resizing of images on upload using S3 event notifications
Implemented usage tracking with CloudWatch Logs and alerts via SNS

4. Azure Function App with Blob Trigger
Created a serverless Azure Function triggered by file uploads to Blob Storage
Parsed incoming JSON/CSV data and stored results in Azure Table Storage
Enabled real-time monitoring with Application Insights

5. CI/CD Pipeline in Azure DevOps (or AWS CodePipeline)
Set up a CI/CD pipeline to deploy a sample web app to Azure App Service (or AWS Elastic Beanstalk)
Integrated GitHub with Azure DevOps Pipelines or AWS CodePipeline + CodeDeploy
Used Infrastructure as Code for repeatable deployments

6. Hybrid Cloud Lab (Azure & AWS)
Created a hybrid cloud lab by connecting Azure VNet to AWS VPC via site-to-site VPN
Simulated enterprise network scenarios including routing, private DNS, and secure VM communication
Validated setup using test VMs, SSH/RDP, and basic latency testing.

MASTER PROJECTS

1- Micro services Deployment on ECS with Fargate
Deployed a multi-container micro services app using Amazon ECS with Fargate launch type
Configured Task Definitions, Service Discovery, and Application Load Balancer for routing
Managed infrastructure using Terraform, and stored container images in Amazon ECR
Implemented auto-scaling policies based on CPU usage and monitored via CloudWatch

2. CI/CD Pipeline for ECS with GitHub Actions and CodePipeline
Built a full CI/CD pipeline for an ECS-based app using GitHub Actions, CodeBuild, and CodePipeline
Containerized a Node.js/Flask app with Docker, pushed to Amazon ECR, and deployed to ECS Fargate
Implemented blue/green deployment strategy and rollback mechanism for zero downtime

3. ECS Cluster with EC2 Launch Type
Set up an ECS cluster with EC2 launch type to host Docker containers on self-managed EC2 instances
Created a custom AMI with Docker and ECS agent preinstalled for optimized scaling
Used CloudFormation to provision the entire stack, including Auto Scaling Groups and ALB

4. Secure ECS Deployment with IAM and VPC Networking
Deployed containerized back-end services using ECS Fargate in a private subnet with NAT Gateway
Configured IAM Task Roles for secure access to S3, Secrets Manager, and CloudWatch Logs
Implemented security groups, VPC endpoints, and CloudTrail for compliance tracking

5. Monitoring and Logging for ECS Workloads
Integrated CloudWatch Logs and AWS X-Ray for distributed tracing and log aggregation of ECS containers
Set up metric filters and alarms to monitor application errors and container health
Used AWS CloudWatch Dashboard to visualize ECS service performance in real time

Soft Skills

- Strong problem-solving and analytical skills
- Ability to quickly learn new technologies and methodologies
- Effective communication and collaboration in team environments
- Strong attention to detail and organizational skills