



# Secure Cloud Infrastructure

AWS Migration Project Week 4 • European Logistics Company  
Cloud Engineering Bootcamp • February 2026

# Project Overview

## Project Goal

Design and deploy a secure, cost-efficient cloud infrastructure on AWS that demonstrates enterprise-grade security practices while maintaining strict budget constraints. This project showcases the ability to balance security requirements with operational costs in a real-world cloud migration scenario.

## Our Approach

- Three-part architecture design following AWS Well-Architected Framework principles
- Security-first design with zero-trust network access and IAM-based authentication
- Infrastructure as Code for repeatability and version control
- Budget ceiling of \$100/month for sustainable operations



# Three-Part Architecture

Our implementation follows a structured approach that separates concerns and demonstrates different deployment methodologies. Each part builds on the previous, creating a cohesive and secure cloud environment.



## Part A: Manual Configuration

IAM roles and access control policies established through AWS Console. Sets the foundation for secure access patterns.



## Part B: Terraform

VPC, networking, and compute resources provisioned using Infrastructure as Code for repeatability.



## Part C: Deployment

Apache web server configuration with S3 integration for static assets and Systems Manager access.

# Part A: IAM Access Control

Our security model eliminates traditional SSH key management by leveraging AWS Identity and Access Management. This approach provides granular control, full audit trails, and reduces attack surface by removing publicly exposed management ports.



## EC2-SSM-Role

Assigned to EC2 instances to register with AWS Systems Manager. Enables secure, auditable management without SSH.



## SSM-Admin

Full Systems Manager access for senior engineers. Complete control for infrastructure management and troubleshooting.



## SSM-Operator

Session Manager access only. Limited permissions for day-to-day operations and routine maintenance tasks.



## ReadOnly

Audit and monitoring permissions. Enables compliance teams to review infrastructure without modification rights.

 **Key Principle:** Zero SSH keys in production. All access is IAM-based, logged, and follows principle of least privilege.



# Part B: Terraform Infrastructure

## Resources Provisioned

Our Terraform configuration defines a complete, production-ready networking stack. All resources are tagged, versioned, and managed as code in our Git repository.

- VPC with CIDR 10.0.0.0/16 spanning multiple availability zones
- Public subnet (10.0.1.0/24) and private subnet (10.0.2.0/24)
- NAT Gateway for secure outbound internet access from private instances
- Security groups with granular inbound/outbound rules
- Two t2.micro EC2 instances (1 public, 1 private)
- Internet Gateway for public subnet connectivity

## Why Terraform

### Repeatable

Deploy identical infrastructure across dev, staging, and production with a single command

### Auditable

Every infrastructure change tracked in version control with full history

### Scalable

Modify variables to instantly scale resources up or down based on demand



# Part C: Website Deployment

The deployment architecture separates public-facing web services from backend infrastructure, implementing defense-in-depth security principles while maintaining efficient content delivery.



## Public Server

Apache HTTP Server installed on public EC2 instance in public subnet. Hosts main website with direct internet access through Internet Gateway. Configured with security headers and minimal attack surface.



## Static Assets

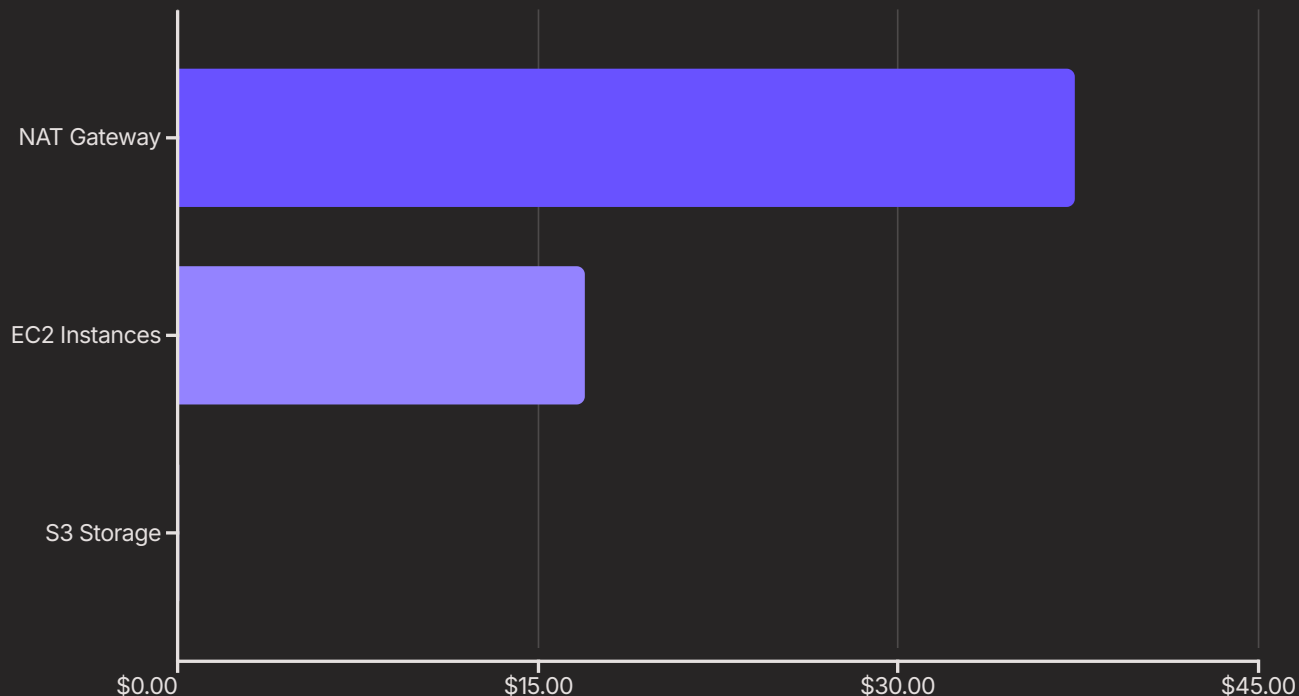
S3 bucket configured for static content delivery. Stores lightweight image assets totaling just 19.7 KB. CloudFront-ready architecture for future CDN integration to improve global performance.



## Private Server Access

Backend EC2 instance in private subnet with no public IP address. Accessed exclusively via AWS Systems Manager Session Manager. Zero exposure to internet threats while maintaining full management capabilities.

# Cost Breakdown



**\$54.33**

**Total Monthly Cost**

Well under the \$100 budget ceiling

**54%**

**Budget Utilized**

Room for scaling and additional services

The NAT Gateway represents the largest cost component, enabling secure outbound internet access for private subnet instances. Alternative solutions like NAT instances could reduce costs further if needed.

# Key Achievements

This project demonstrates the ability to design and implement production-grade cloud infrastructure that meets enterprise security standards while operating within startup-level budget constraints.

## Enterprise-Grade Security

Network isolation with public and private subnets. Private resources have zero direct internet exposure. All ingress and egress traffic controlled by security groups with strict rules following principle of least privilege.

## IAM-Based Access Control

Zero SSH keys or passwords in production environment. All access authenticated through AWS IAM with full CloudTrail audit logging. Role-based access ensures proper separation of duties across team members.

## Cost-Optimized Infrastructure

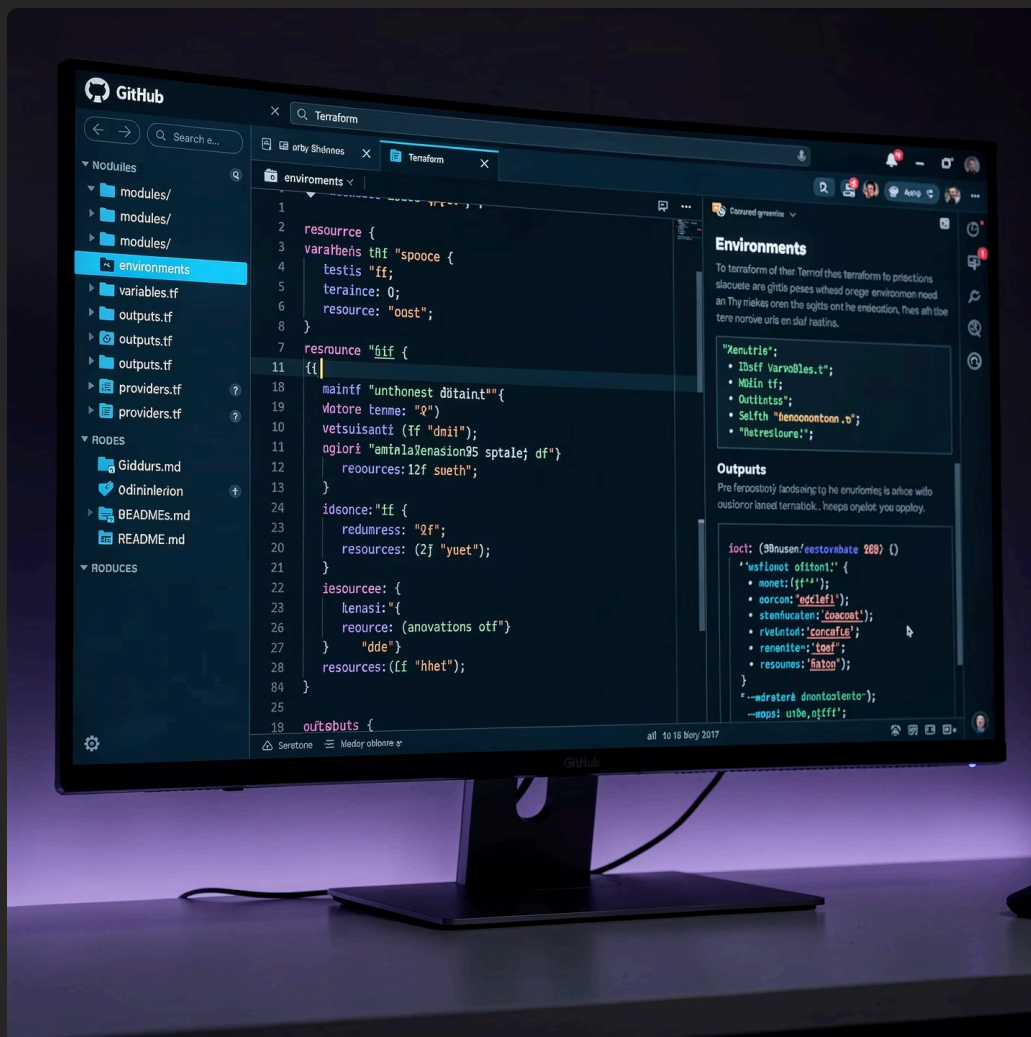
Strategic use of t2.micro instances and efficient resource allocation. S3 for static content rather than expensive EBS volumes. Total monthly cost of \$54.33 leaves budget headroom for growth and additional services.

## Production-Ready Design

Infrastructure follows AWS Well-Architected Framework across all five pillars. Terraform-managed resources enable rapid disaster recovery. Architecture can scale horizontally with minimal code changes to support increased traffic.



# Project Repository



## GitHub Repository

### week4-secure-web

The complete project source code is available in our GitHub repository, providing full transparency and enabling peer review. All infrastructure is defined in version-controlled Terraform files with detailed documentation.

## Repository Contents

- Complete Terraform configuration files with modular structure
- Technical documentation including setup instructions and troubleshooting guides
- Architecture diagrams showing network topology and data flow
- IAM policy definitions and security group rules
- Cost estimation scripts and budget tracking

[View Repository](#)



# Secure Cloud Infrastructure Project

Week 4 – Cloud Engineering Bootcamp

---

## Questions?