SECURED INFRASTRUCTURE & SECURITY CONTROLS DEMO

Presented to: Deloitte Cyber Center
Directors & Managers

Objective: Demonstrate a secure AWS environment deployed via Infrastructure as Code (IaC) in alignment with NIST CSF v1.1

Presented By Zach Almog

Goals

- **Develop and present:**
  - **slide deck demonstrating a secure AWS environment.**
- **Process Run Dows:**
  - **outline steps aligning with NIST CSF (protect function).**
- **Demo:**
  - **AWS-based solution (Lambda, S3, IAM), IaC deployment (CloudFormation), and security controls demonstration (logging, encryption, least privilege).**

Approach & Tools

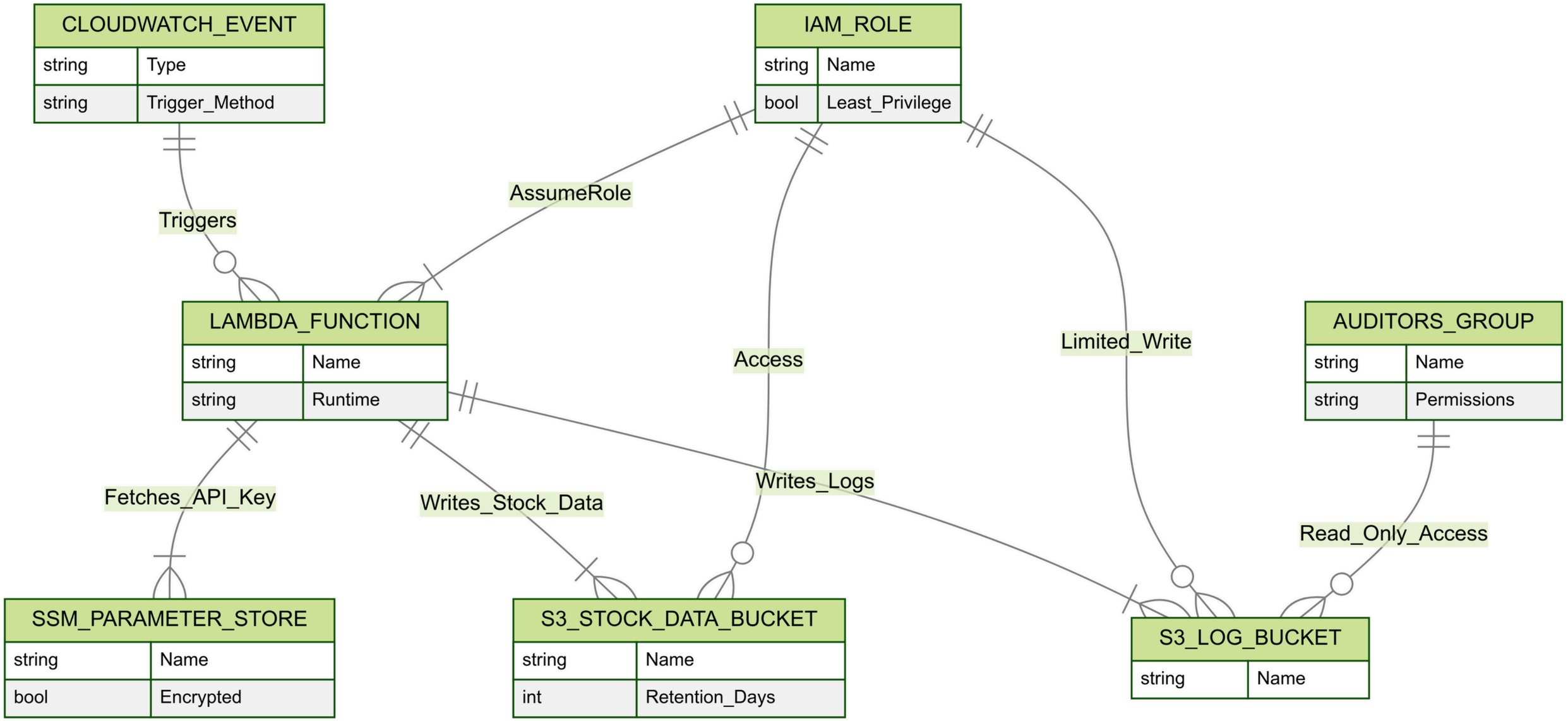
- **Framework: NIST Cybersecurity Framework v1.1**

- **Selected Subcategories:**
  - **PR.PT-1 (Audit/log records).**
  - **PR.DS-1 (Data-at-rest protection).**
  - **PR.AC-3 (Remote access management).**

- **AWS Services:**
  - **Lambda for stock data retrieval.**
  - **S3 for data & log storage.**
  - **IAM for least-privilege roles/groups.**
  - **SSM Parameter Store for secret management.**
  - **IaC: AWS CloudFormation (GitHub: almogzach/deloitte).**

Architecture & Demo Flow

- **Lambda Function:**
  - **Fetches stock prices using an API key in SSM Parameter Store.**
  - **Writes data to StockDataBucket.**
- **S3 Buckets:**
  - **StockDataBucket for fetched data.**
  - **LogBucket for Lambda logs.**
- **IAM & Auditors:**
  - **Least-privilege role for Lambda (S3 + SSM access only).**
  - **AuditorsGroup read-only access to LogBucket.**
- **Encryption:**
  - **Server-side encryption (SSE-S3) for data-at-rest.**
  - **No plaintext secrets (API key in SSM Parameter Store).**


CLOUDWATCH_EVENT
string
Type
string
Trigger_Method
IAM_ROLE
string
Name
bool
Least_Privilege
AssumeRole
Triggers
LAMBDA_FUNCTION
string
Name
string
Runtime
Access
Limited_Write
AUDITORS_GROUP
string
Name
string
Permissions
Writes_Logs
Fetches_API_Key
Writes_Stock_Data
Read_Only_Access
SSM_PARAMETER_STORE
string
Name
bool
Encrypted
S3_STOCK_DATA_BUCKET
string
Name
int
Retention_Days
S3_LOG_BUCKET
string
Name

NIST CSF Controls Implementation

- **PR.PT-1 (Audit/Log Records)**
  - **CloudWatch triggers Lambda.**
  - **Logs stored in LogBucket.**
  - **AuditorsGroup can review logs regularly (segregation of duties).**
- **PR.DS-1 (Data-at-Rest Protection):**
  - **S3 server-side encryption.**
  - **API key secured in SSM Parameter Store.**
- **PR.AC-3 (Remote Access Management):**
  - **Least-privilege IAM role for Lambda.**
  - **Read-only policy for auditors.**

Results & Next Steps

- **Results:**
  - **Secure AWS environment deployed via CloudFormation.**
  - **Logging, encryption, and access control aligned with NIST CSF.**

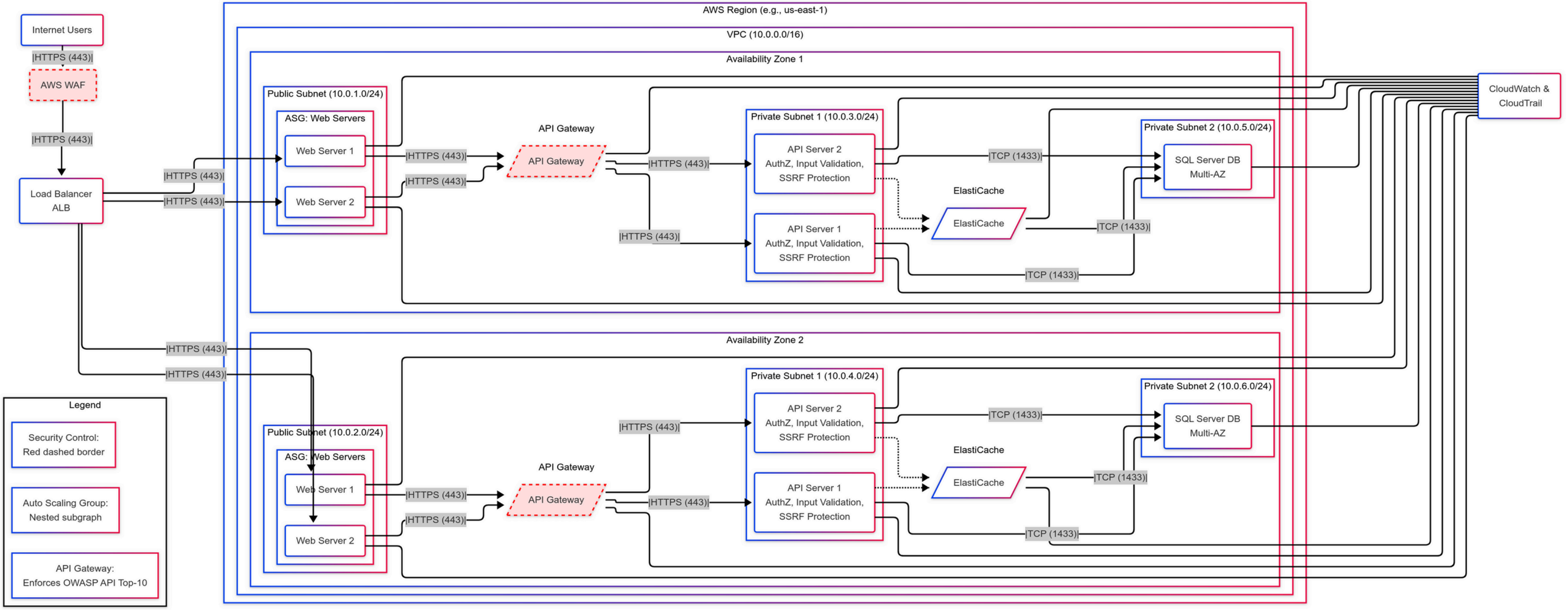
- **Next Steps:**
  - **Integrate with more services (e.g., EC2, VPC).**
  - **Add CI/CD pipelines with automated security checks.**
  - **Expand monitoring with AWS Security Hub, GuardDuty, etc.**

Q&A

- **Git Repository: GitHub: almogzach/deloitte**
- **AWS Console Demo (S3, Lambda, IAM, CloudWatch)**
- **Contact: [Zach Almog / zalmog@gmail.com]**

Let get to part 2...

Secure Three-tier Web Application



High Availability & Scalable Cloud Architecture

- **Objective:**
  - **Design a highly available, scalable, and secure web application infrastructure.**
  - **Ensure protection against API security threats (OWASP API Top-10).**
- **Key Components:**
  - **1 AWS Region, 2 Availability Zones.**
  - **Load Balancer (ALB) for traffic distribution.**
  - **Auto Scaling Groups for web & API servers.**
  - **VPC with Public & Private Subnets.**
  - **ElastiCache & Multi-AZ SQL DB for performance & redundancy.**
  - **Security controls at every layer (WAF, API Gateway, IAM, Security Groups, Encryption).**

Addressing Scalability & Traffic Growth

- **Auto Scaling Groups (ASG):**
  - **Web servers and API servers scale horizontally based on demand.**
  - **ALB distributes traffic evenly across multiple instances.**
  - **ElastiCache (Caching Layer):**
  - **Reduces load on SQL database by caching frequent queries.**
  - **Ensures faster response times & cost efficiency.**
- **Multi-AZ Database (SQL Server):**
  - **Primary & Standby DB for high availability.**
  - **Failover enabled for automatic recovery.**
- **API Gateway + AWS WAF:**
  - **Filters malicious requests & scales automatically to handle spikes.**

Security & OWASP API Top-10 Mitigation

- **Web Application Security Controls:**
  - **AWS WAF: Protects against common web attacks (SQL Injection, XSS).**
  - **API Gateway: Implements authentication, rate-limiting, and OWASP API protections.**
  - **IAM Roles & Policies: Enforces least privilege for access control.**
- **API Security Based on OWASP API Top-10:**
  - **Authentication & Authorization (API Gateway & IAM) → Prevents Broken Auth (API1).**
  - **Input Validation & SSRF Protection (API Servers) → Prevents SSRF Attacks (API10).**
  - **Logging & Monitoring (CloudWatch & CloudTrail) → Detects Security Misconfigurations (API7).**

Secure Data Flow & Compliance

- **Secure Data Flow Across Tiers:**
  - **HTTPS (443) Everywhere: Encrypts traffic from users → ALB → API → Database.**
  - **Security Groups & ACLs: Restrict access between subnets.**
  - **Encrypted S3, Multi-AZ DB, IAM MFA: Data-at-rest protection.**

- **Final Takeaway:**
  - **The architecture scales efficiently, maintains high availability, and meets security best practices.**
  - **API security measures ensure protection from OWASP API Top-10 threats.**
  - **Optimized performance with caching & multi-tier design.**

💡 **Ready for enterprise deployment!**
Thank you for your time!