PUG CHALLENGE AMERICAS

How to: Secure and Scale your Modern OpenEdge® application on aws

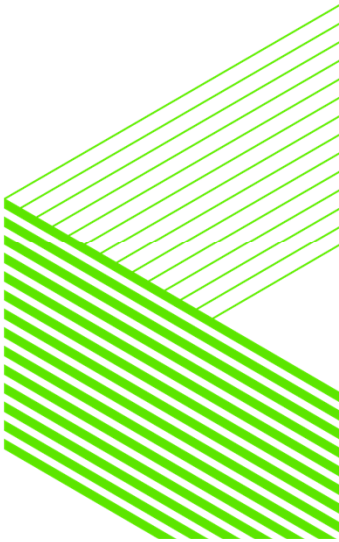
By Paul Guggenheim

About PGA

- OpenEdge Evangelist, Developer, Trainer, and Course Designer
- **Progress/OpenEdge** Partner
- Amazon Authorized Instructor
- AWS Solutions Architect
- **White Star** Software Strategic Partner
- **Consultingwerk** Partner
- **AppPro** Reseller

Overview

- Traditional On-Premise Application Infrastructure
  - Infrastructure Responsibilities and Tasks
  - Risks and Costs
- Why AWS?
  - Background
  - Relevant Cloud Services
- Cloud Application Infrastructure
  - Advantages
    - Availability and Resilience
    - Security
    - Performance
    - Monitoring
    - Cost
- Class Participation Demo
  - 3 - tier architecture
  - Enter Data and Email Notification

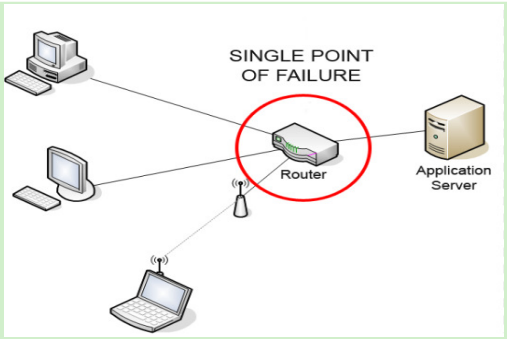
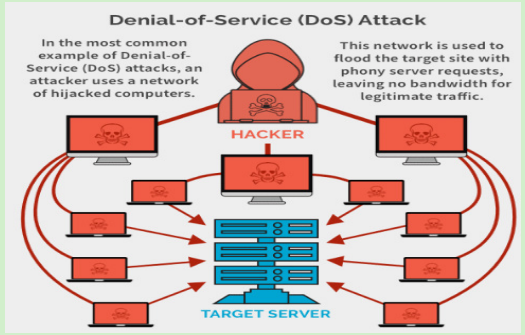


Traditional On-Premise Application Infrastructure

- In addition to the software application, your company is responsible for:

Rack and Stack	Power Backup	HVAC
		

Traditional On-Premise Application Infrastructure Risks

Natural Disasters	Hardware Failure	Network Failure
		
Denial of Service (DoS) Attack	Computer Identity Hackers	IT Growth Planning Risk
		

Traditional On-Premise Application Infrastructure Costs

- To achieve high availability or fault tolerance in the event of a natural disaster, hardware failure or network failure:
  - additional computer and networking hardware must be purchased
  - An additional remote physical location must be obtained (bought or leased) in the event of a natural disaster or terrorism.
- To prevent DoS and Identity hacking on-premise attacks, additional costs are incurred for employing IT security professionals or hiring 3rd party security audit firms.
- Maintaining tight **facility** security to reduce the likelihood of a breach, involves more personnel, additional biometric hardware, and more detailed security audits resulting in higher costs compared to the cloud.
- The consequence of not possessing ample hardware infrastructure for growth is lost revenue.
- If too much hardware is purchased ahead of the anticipated growth then those funds are misallocated and not needed and should been applied to other research projects.

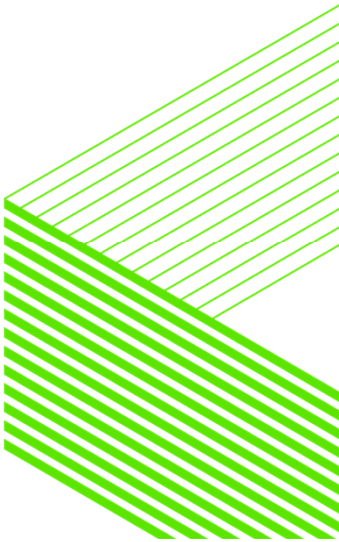
Why Amazon Web Services?

- First company to deliver worldwide cloud web services in 2006
  - S3 (Simple Storage Service) Cloud Storage
  - EC2 (Elastic Compute Cloud) Service – VMs
  - SQS (Simple Queue Service) Messaging

- AWS is the market leader
  - Revenue $80 Billion, Income $22.8 Billion 2022 (theregister.com 4/26/23)
  - As of Q4 2021 Market Share (according to Synergy Group)
    - AWS 33%
    - Microsoft Azure 21%
    - Google Cloud 10%

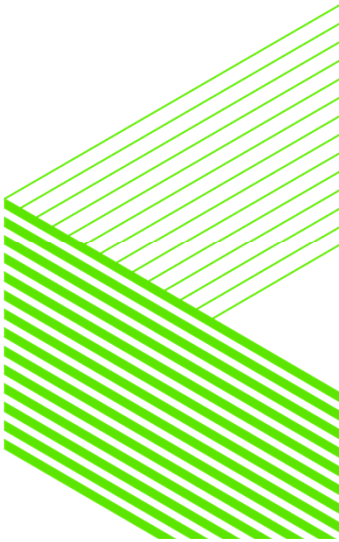
AWS Cloud Structure

- 32 Geographic Regions
- At least 3 Availability Zones (AZ) per Region
  - Isolated
  - Physically Separated
  - >= 60 miles apart from next AZ
- At least 1 Data Center per AZ
  - Highly Secure
  - No Physical Access to public
  - Many Server Racks



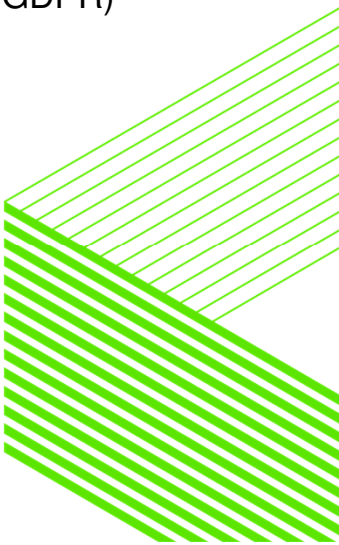
Cloud Application Infrastructure Availability Advantages

- Worldwide location choices for infrastructure
- Availability zones
- Private Networking – Virtual Private Cloud (VPC)
- Resilient web services
  - Fully Managed
    - Internet Gateway
    - NAT Gateway
    - Messaging – SQS and SNS
    - Serverless – Lambda
  - Provisioned
    - Virtual Machines – EC2
    - Storage



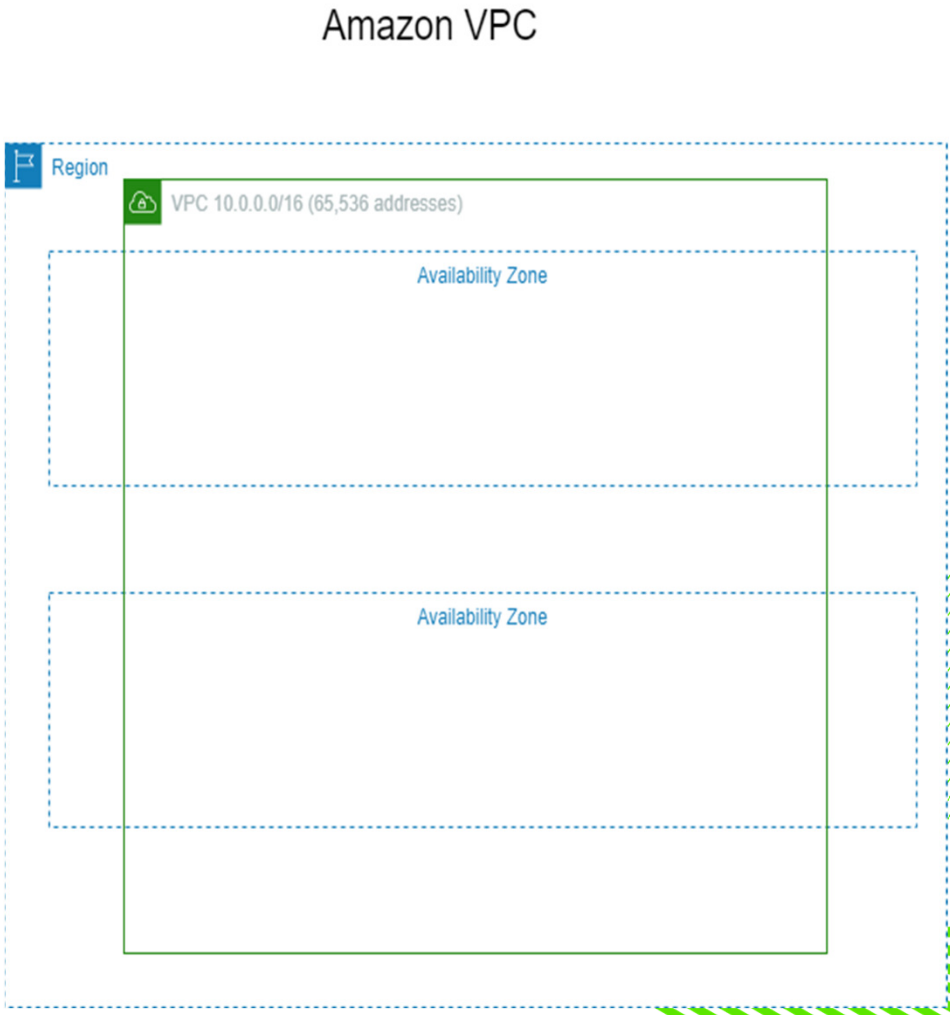
Locations

- Internet is the slowest network
  - Everyone is on it.
  - Data packets travel through many server hops to arrive at a destination, this is called latency.
  - When working in the cloud, the goal is to minimize time and distance spent on the Internet.
  - Typically, a region is selected based on:
    - Proximity of application customers or developers to minimize latency
    - Compliance with the region's governing laws and regulations concerning data and privacy (eg GDPR)
    - Service Availability
    - Cost
- Use multiple AZs to ensure high availability
  - Designed for Fault Isolation
  - Interconnected using high-speed private links



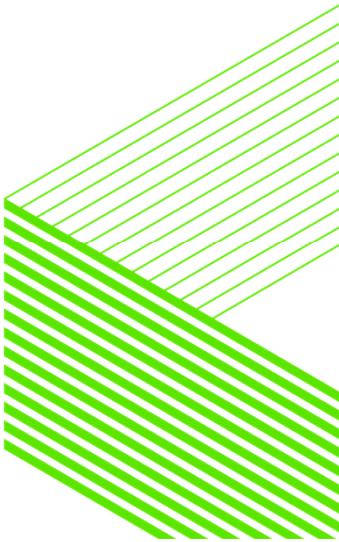
Virtual Private Cloud (VPC)

- Provides a logical isolation for application workloads
- Provides for custom access controls and security settings
- Bound to a single region, a region may have many VPCs
- A single VPC may span many AZs
- A single AZ may contain more than 1 VPC



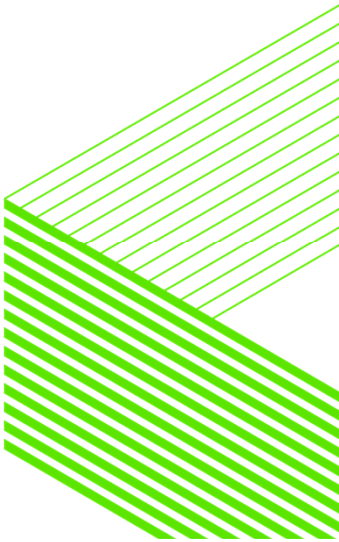
Fully Managed Web Services

- What does Fully Managed by AWS mean?
  - Horizontally Scaled
  - Redundant
  - Highly Available
  - No bandwidth constraints
  - Don’t need to worry about them failing on providing inadequate bandwidth



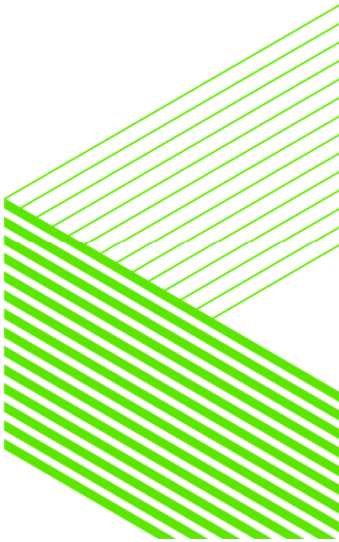
Provisioned Web Services

- User determines how many to use
- Requires management by User
- EC2 is resilient – may be placed in multiple AZs for failover
- Storage Types
  - S3 – Object Storage
    - Encrypted by default
    - Automatically redundant - provide 99.999999999% (11 9's) of data durability per year
    - Popular Backup and Restore Use Case – 6 million backups/day by 400,000 AWS Customers
  - EBS – Elastic Block Storage
    - Durable Block Level Storage
    - Detachable to various EC2 instances
    - Attach one or more EBS volumes to a single EC2 instance as needed



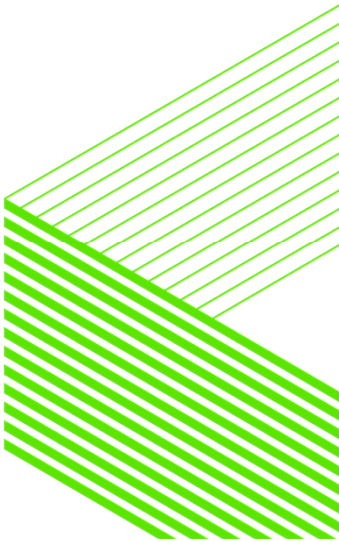
Provisioned Web Services (continued)

- Storage Types
  - EFS – Elastic File System
    - Encrypted by default
    - Allows multiple EC2 instances to simultaneously connect to same EFS



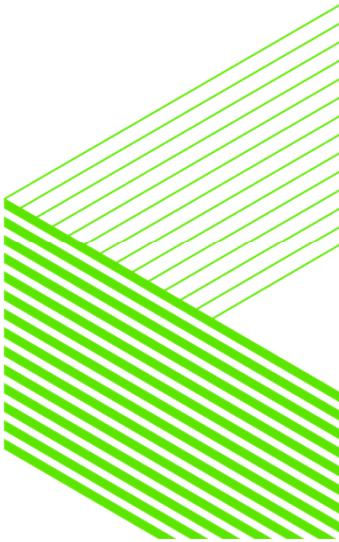
Cloud Application Infrastructure Security Advantages

- Principal of Least Privilege
- Identity and Resource Security
- Apply multiple security layers
- Temporary Access Roles
- Multiple Account Security
- Easy Access to 3rd party authentication systems
- Convenient setup of 2 factor authentication
- Rotating encryption keys
- Rotating IP addresses
- Encryption at rest
- Private Networks
- Fully Managed Services, Serverless



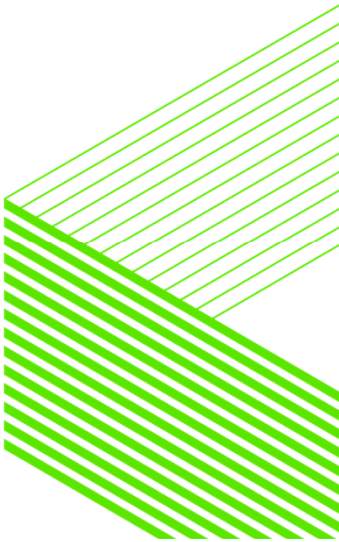
The Principle of Least Privilege

- Grant users only the level of access that is required.
- AWS Security is provided in a granular way to make this happen.
- Access to resources is implicitly denied (default).
- This requires an explicit allow policy to allow access.
- An explicit deny policy will override and explicit allow policy for the same type of access.



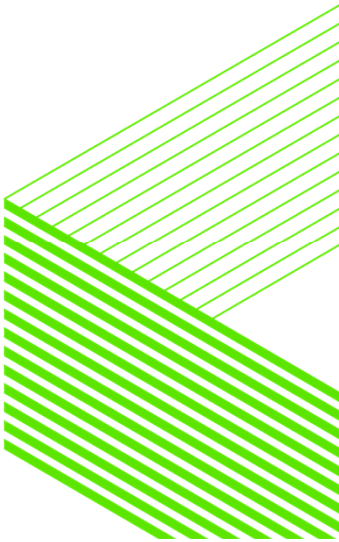
Identity Security

- Identity Security refers to a principal making a request for an action or operation on an AWS resource.
- What is a principal?
  - A defined IAM (Identity Access Management) User
  - IAM Role – Grants temporary permissions
  - AWS Service
  - Identity Provider (IdP)
    - Identities outside of AWS IAM
    - For example, login to Microsoft Active Directory, Facebook, Google
    - Provides Authentication and AWS provides Authorization

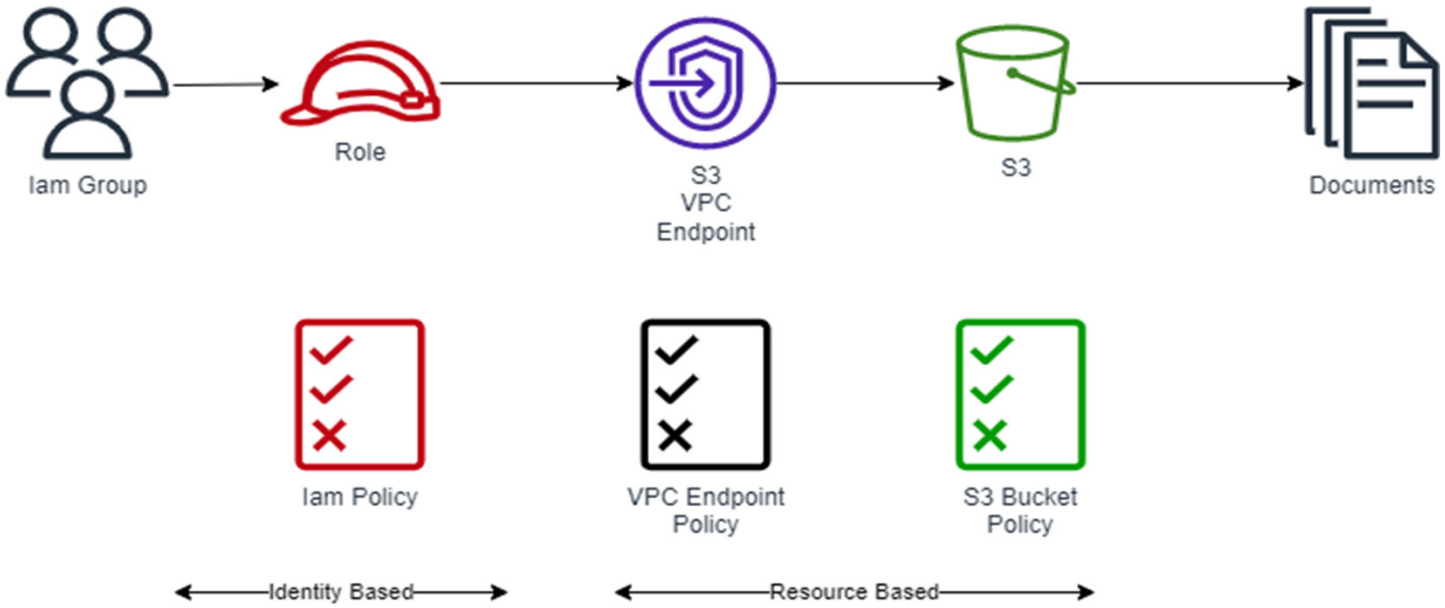


Resource Security

- Resource Security are policies that are attached to a single resource, and principals are specified as to what actions are allowed or denied.

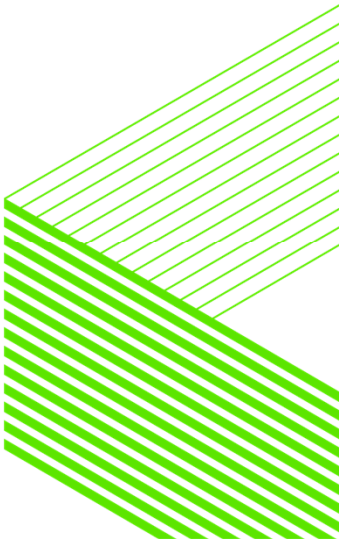


Apply multiple security layers



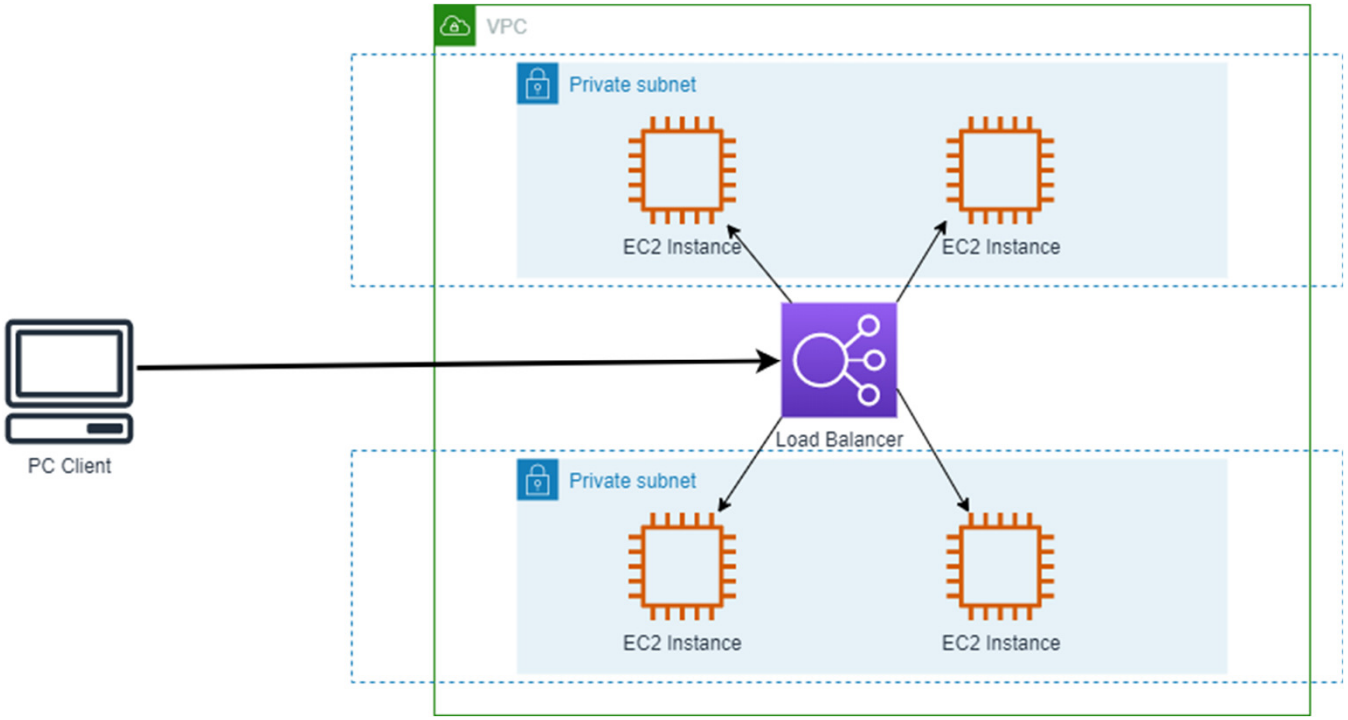
Cloud Application Infrastructure Performance Advantages

- Load Balancing
- Auto Scaling
- Edge Servers
- Serverless – Lambda
- Containerization
- Private High Speed Internet Backbone
- Scale to any size server
- Scale to any disk array configuration
- High speed networking options
- Tenancy Control – Dedicated Instances/Hosts
- Clustering and Caching Options



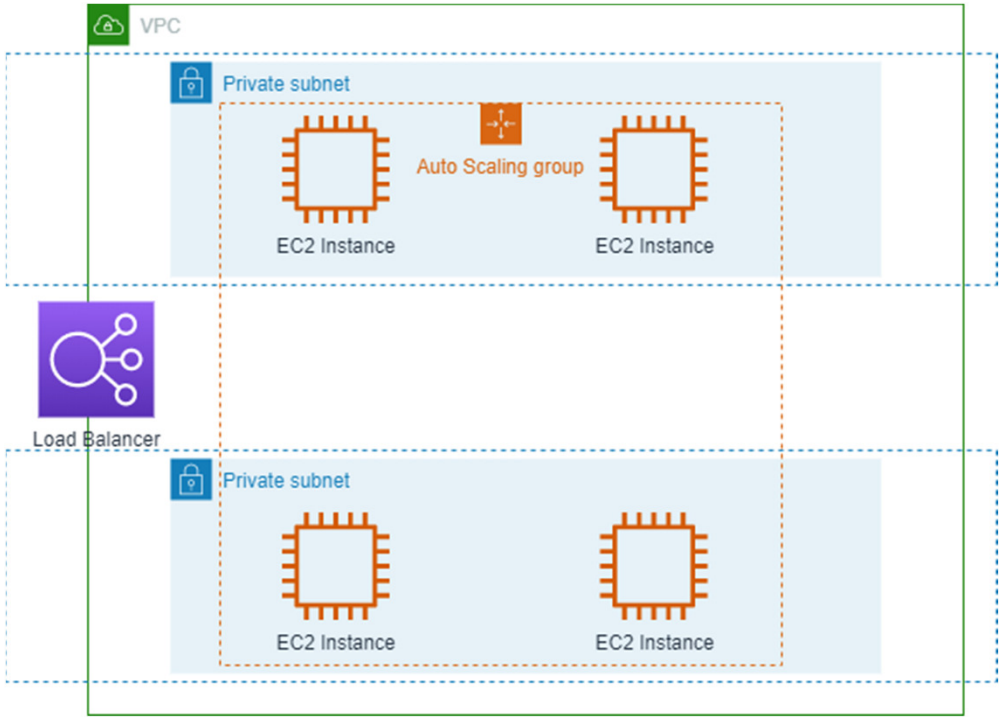
Load Balancing

- A load balancer distributes incoming application traffic across multiple targets such as EC2 instances, in multiple Availability Zones to increase the availability and performance of an application.



Auto Scaling

- AWS Auto Scaling monitors applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost.



Edge Services

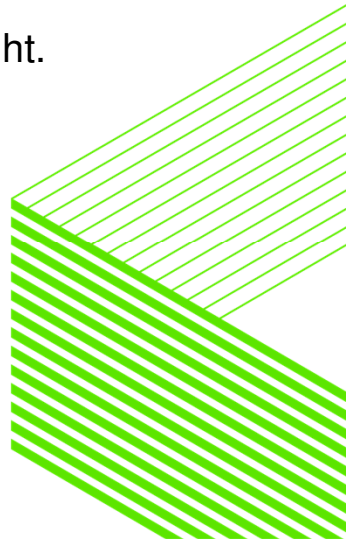
- A CloudFront edge server provides the fast delivery to the end user by reducing the number of hops across the Internet by using the AWS network instead.
- Cloudfront has the ability to cache web *objects or files* in multiple edge locations around the world.

Serverless Computing - Lambda

- What is Serverless computing?
  - Build and run applications and services without:
    - Provisioning
    - Scaling
    - Managing
  - Servers
- Lambda 
  - Run code without provisioning, scaling or managing servers
  - Performs all administration of the compute resources
  - Lambda supports: Node.js, Java, C#, Python, Go, Ruby and Powershell
  - What about OpenEdge? – It's possible through a custom runtime

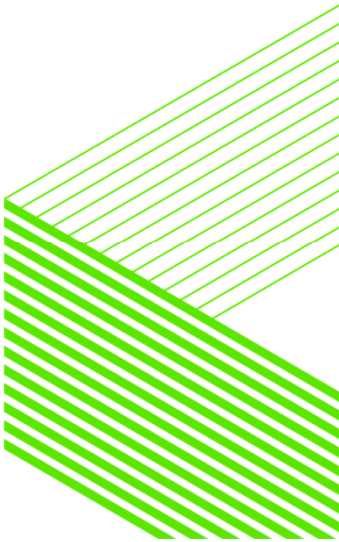
Containerization

- What is a container?
  - Containers are packages of software that contain all of the necessary elements to run in any environment.
  - Notice it doesn't mention the operating system.
- Virtual Machines vs Containers for running applications
  - VMs take much longer to invoke because they include the operating system.
    - This requires more memory and disk
  - Containers use a container engine to startup quickly, hence much smaller and lightweight.
  - Many container instances may be run on one VM.
- AWS offers 2 container orchestration services to manage the container execution:
  - Elastic Container Service
  - Elastic Kubernetes Service



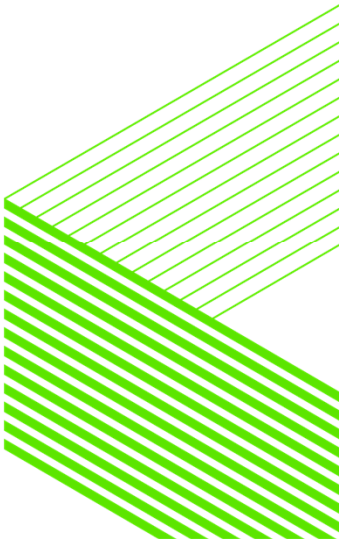
Cloud Application Infrastructure Monitoring Advantages

- Built in monitoring tools
  - Operational Health – Perform Health Checks
  - Application Performance – Metrics feed Load Balancing and Auto Scaling
  - Resource Utilization
  - Security Auditing
- What can be monitored?
  - Application Log Data
  - User activity
  - IP Traffic
  - Custom Logs
- Setup alarms
  - Notification
  - Take Action



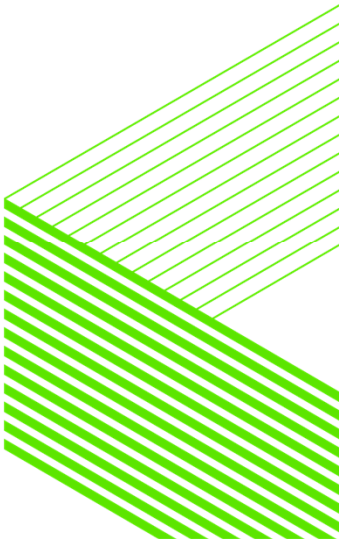
CloudWatch

- What is CloudWatch? – AWS service that provides a near real-time stream of system events.
  - Able to evaluate:
    - Resource use
    - Application performance
    - Operational health
  - CloudWatch Features
    - Automatic dashboards
    - Data with one-second granularity
    - Up to 15 months metrics storage and retention
    - Possesses AWS built-in metrics
    - Create your own custom metrics
    - Alarms
      - Send notifications
      - Automatically make changes to resources based on user-defined rules



Cloud Application Infrastructure Cost Advantages

- Minimize Fixed costs – only pay for what is used
  - Almost all AWS charges are based upon usage and variable costs
- Cost Tracking / Optimizing Tools
- Server Purchase Options
- Lower cost storage options
- Serverless Computing
- Containers – Loose Coupling

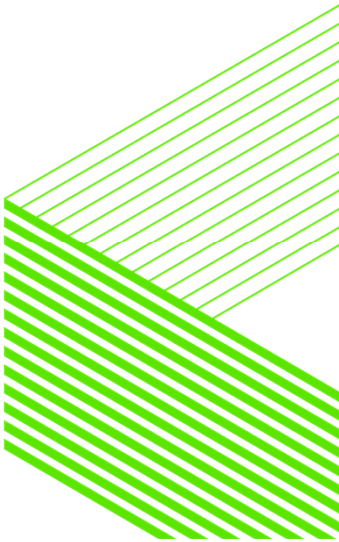


Cost Tracking / Optimizing Tools

- Pricing Calculator – Creates estimates
- Cost Explorer – Analyzes cost and usage
  - Tracks data for the last 12 months
  - Creates forecast for the next 12 months
- Compute Optimizer - Analyzes the configuration and utilization metrics
  - Generate recommendations to reduce cost and improve performance
- AWS Budgets – Set budgets and apply actions and notifications based on thresholds
  - Email notifications
  - For example, could prevent provisioning more EC2 instances if costs approach 80% of budget
- Cost Anomaly Detection - leverages advanced Machine Learning technologies to identify:
  - anomalous spend and root causes
  - Create alerts and take action

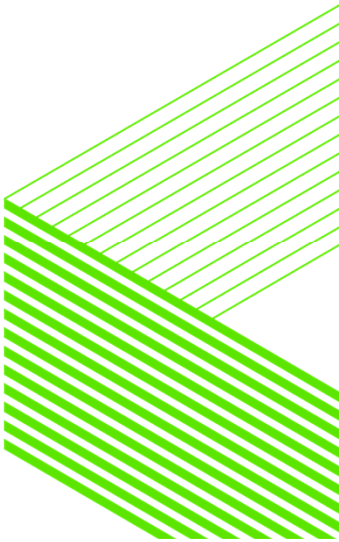
Server Purchase Options

- On-Demand
  - Pay for compute capacity with no long-term commitments
- Savings Plans
  - 1 or 3 year commitment
  - EC2 Instance Saving Plans
    - Flexible across AZ, Size, OS and Tenancy
  - Compute Savings Plans
    - Flexible across EC2 attributes plus Instance Family and Region
- Spot Instances
  - Take advantage of spare EC2 capacity for temporary use
  - For flexible workloads – select price willing to pay and capacity needed
  - Interruption can occur if no capacity at your maximum price



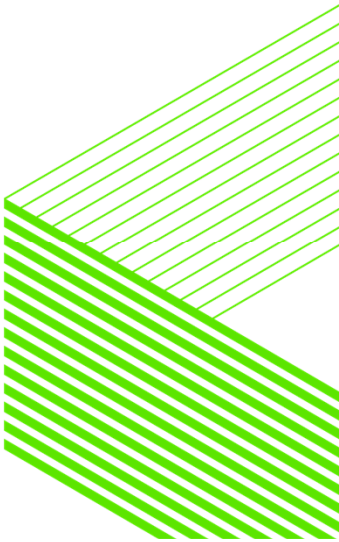
Lower cost storage options

- EBS Volume Options
  - HDD vs SSD
  - General Purpose Drives vs Provisioned io drives
  - Lowest cost HDD for less frequently accessed workloads
- S3 Options
  - 6 different storage classes from higher cost frequent access to lower cost infrequent access
  - Intelligent Tiering – Automates movement of objects between storage classes
  - User-Defined Lifecycle Policies
    - Delete or Move Objects to different storage class based upon age or activity
- EFS Options
  - 2 Storage Classes: Standard and Infrequently Access
  - 2 Throughput Modes: Bursting and Provisioned



Class Participation Demo

- This demo shows a ABL Client calling an ABL Server using REST Services.
- Employee CRUD operations from the sports2020 database is demonstrated.
  - GET, POST, PUT, DELETE verbs
- All Server resources are located in the AWS cloud.
- AWS Services Illustrated in this demo include:
  - Route 53 Hosted Zone DNS records
  - AWS Certificate Manager
  - Elastic Compute Code (EC2) Instances
  - Application Load Balancer
  - Virtual Private Cloud (VPC)
  - Availability Zones (AZ) for High Availability
  - Security Groups
  - Simple Notification Service (SNS) – Email – Command Line Interface (CLI)



Class Participation Demo

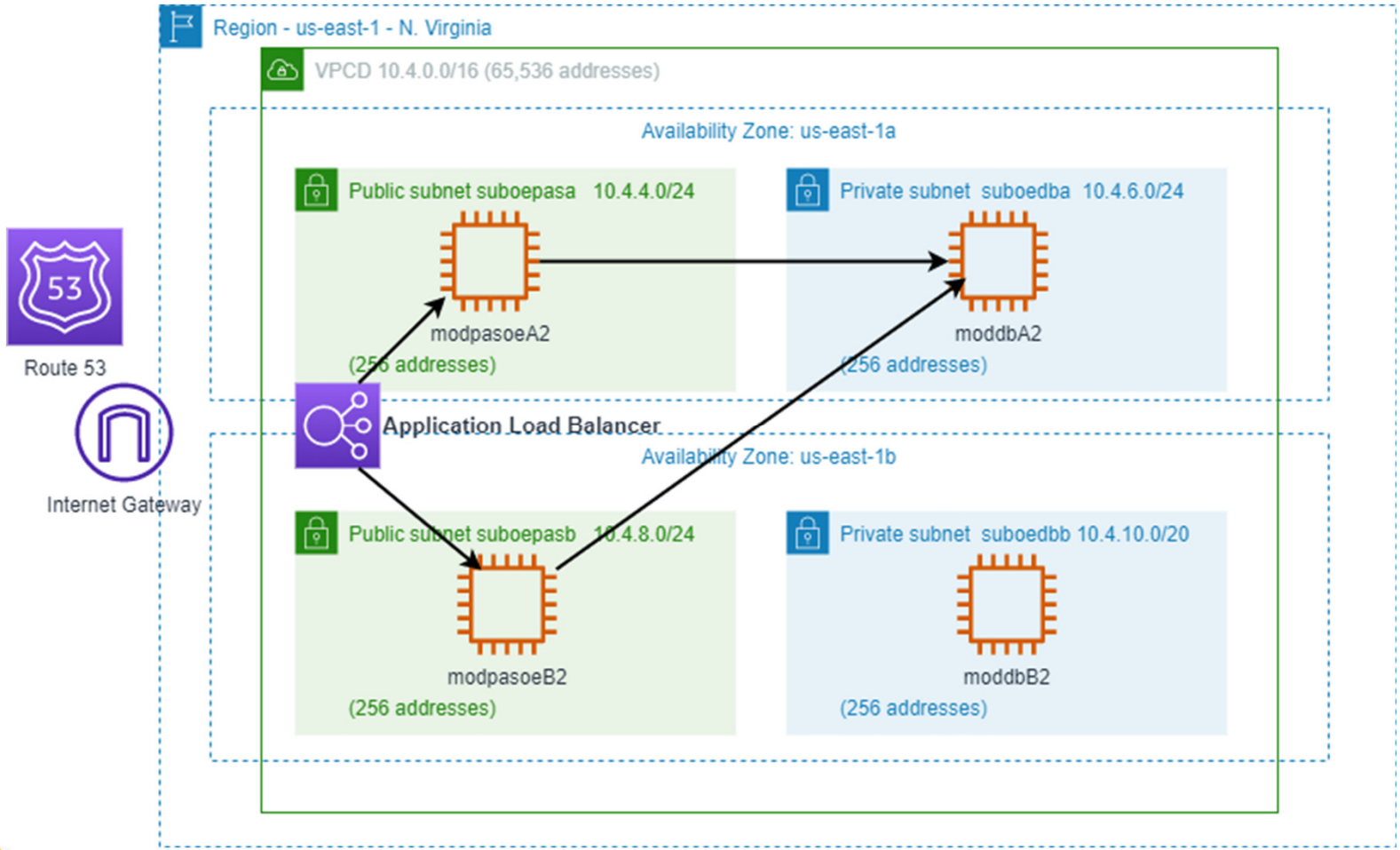
- To participate in the demo please subscribe to the Email SNS by typing the following URL into your browser:

https://pgapugchallenge.com:9852/ModernWeb/web/subscribe/<*email-address*>/yes

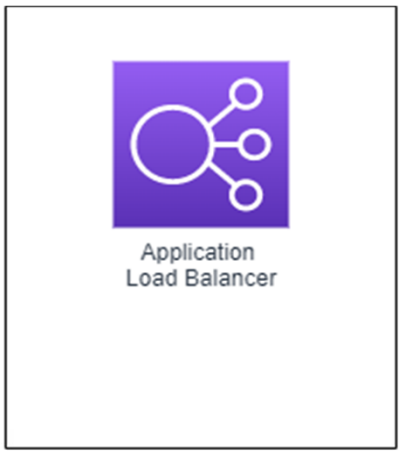
Or

- You will receive an email message from a user called Employee Change with a subject of:
  - AWS Notification – Subscription Confirmation
  - To confirm this subscription click the Confirm subscription link below.
  - This subscription will publish json text of changes to the employee table.

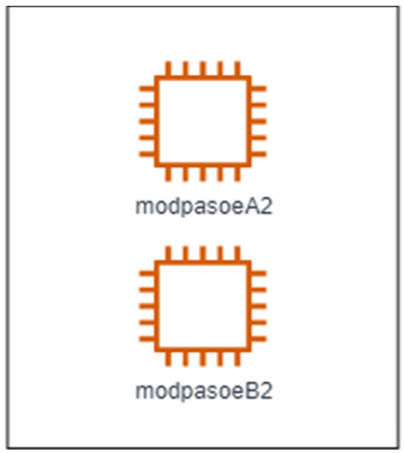
AWS Architecture



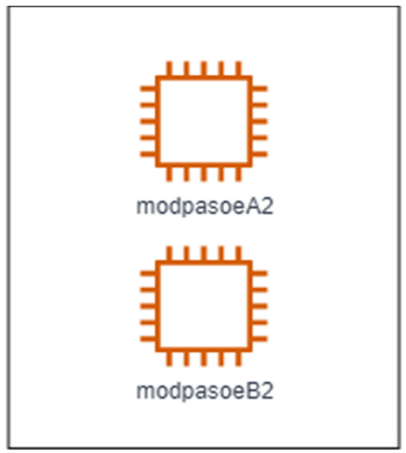
Security Groups



Security Group:
modalb



Security Group:
modpasoe



Security Group:
moddb

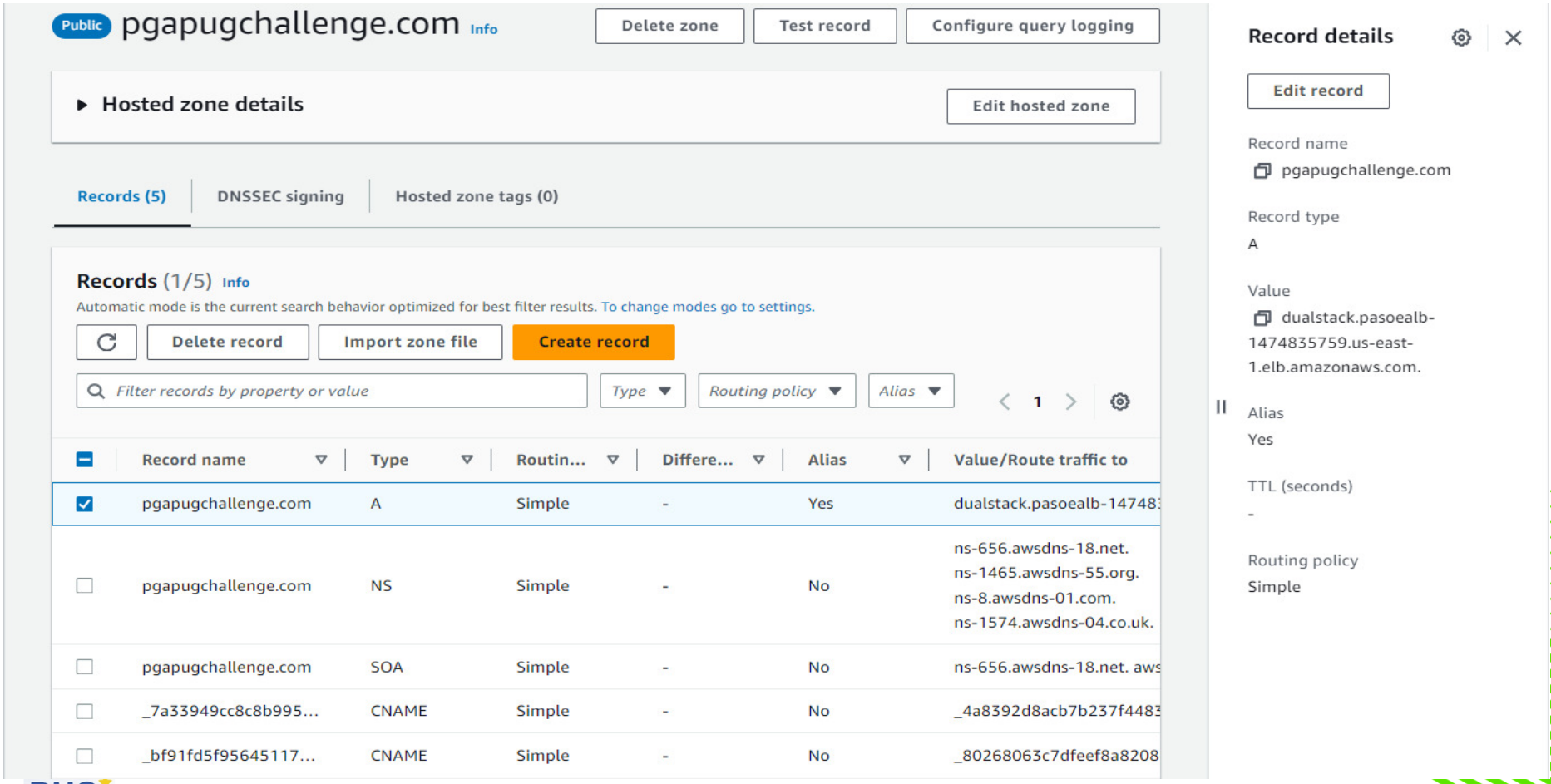
Inbound Rules

Protocol	Port	Source
ICMP	ALL	0.0.0.0/0
TCP	9851-9853	0.0.0.0/0

Protocol	Port	Source
TCP	9851-9853	**modalb**

Protocol	Port	Source
TCP	6666	**modpasoe**
TCP	1025-2000	**modpasoe**

Hosted Zone – pgapugchallenge.com



Simple Notification Service (SNS) – Email – Command Line Interface (CLI)

- To publish a message from a file using the CLI:

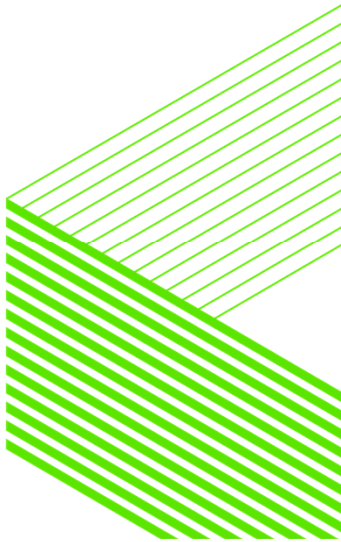
```
aws sns publish --topic-arn <arn> --subject "Employee Write" --region us-east-1 --message <filename>
```

- To subscribe to a topic:

```
aws sns subscribe --topic-arn <arn> --protocol email --notification-endpoint <emailaddress>
```

- AWS CLI Reference Link:

https://docs.aws.amazon.com/cli/latest/reference/



Employee Webhandlers

- The following are the resource URIs for the employeeWH webhandler:

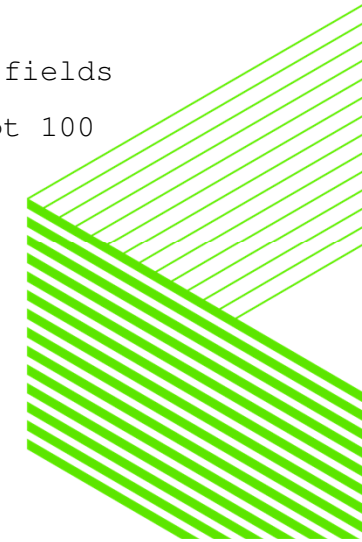
```
/employee/{EmpNum}/{nophone}
/employee/{EmpNum}
/employee
/employeedepartment/{DeptCode}/{nophone}
/employeedepartment/{DeptCode}
```

- As an exercise, try a few of the following URLs:

https://pgapugchallenge:9852/ModernWeb/web/employee/1 (shows employee with EmpNum=1)

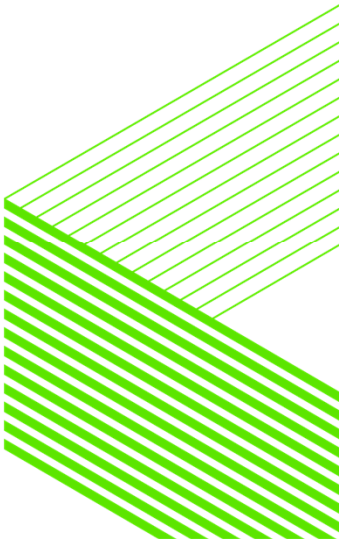
https://pgapugchallenge:9852/ModernWeb/web/employee/1/yes (shows EmpNum=1 with no phone fields

https://pgapugchallenge:9852/ModernWeb/web/employeedepartment/100 (shows employees of dept 100



How to parse the employee webhandler on the handleget method

```
 CASE poRequest:UriTemplate:
        WHEN "/employee" THEN ASSIGN filtervar = "".
        WHEN "/employee/~{EmpNum}"
     OR WHEN "/employee/~{EmpNum}/~{nophone}"
          THEN DO:
            ASSIGN cEmpnum = poRequest:GetPathParameter('EmpNum').
            IF cEmpNum = "all" THEN filtervar = "".
            ELSE filtervar = SUBSTITUTE("WHERE employee.EmpNum = &1", cEmpNum).
          END.
.
.
.
oemployeeBE:ReademployeeBE (filtervar, OUTPUT DATASET dsEmployee).
```

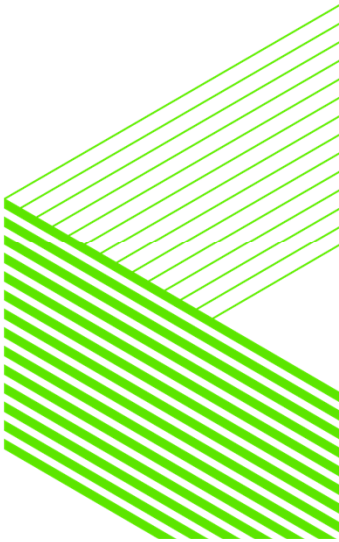


How to send http client requests using the ABL (POST example)

```
DEFINE INPUT        PARAMETER ipURI    AS CHARACTER NO-UNDO.
DEFINE INPUT        PARAMETER ipMethod AS CHARACTER NO-UNDO.
DEFINE INPUT-OUTPUT PARAMETER DATASET FOR dsemployeeclient.

DATASET dsemployeeclient:WRITE-JSON ("JsonObject", msgbody).
ASSIGN oRequest = RequestBuilder:Post(ipURI, msgBody):AcceptJson():REQUEST.
ASSIGN oResponse = Clientbuilder:Build():Client:Execute(oRequest)
       oEntity   = oResponse:Entity
       cLogMsg   = oResponse:ContentType.

IF TYPE-OF(oEntity, JsonObject) THEN
  DATASET dsemployeeclient:READ-JSON ("JsonObject",CAST(oEntity, JsonObject),"EMPTY").
```



Summary

- Use AWS Infrastructure to:
  - Improve Availability and Resilience
  - Tighten Security
  - Increase performance
  - Enhance application functionality
  - Reduce Cost
- To learn more about AWS:

https://docs.aws.amazon.com/

- To setup an AWS account use the Free Tier offering:

https://docs.aws.amazon.com/whitepapers/latest/how-aws-pricing-works/get-started-with-the-aws-free-tier.html

Questions