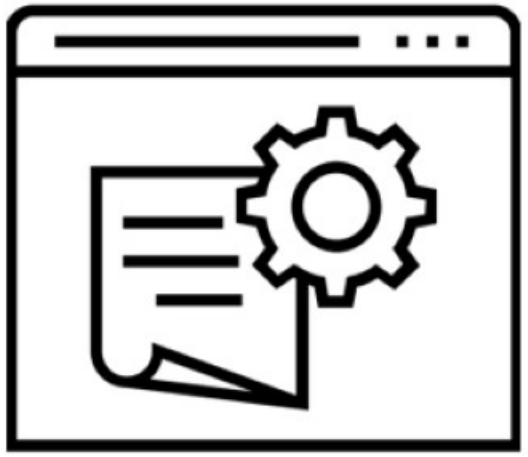
Amazon Web Services Architect Associate Certification

AWS Managed Services

What We will Cover

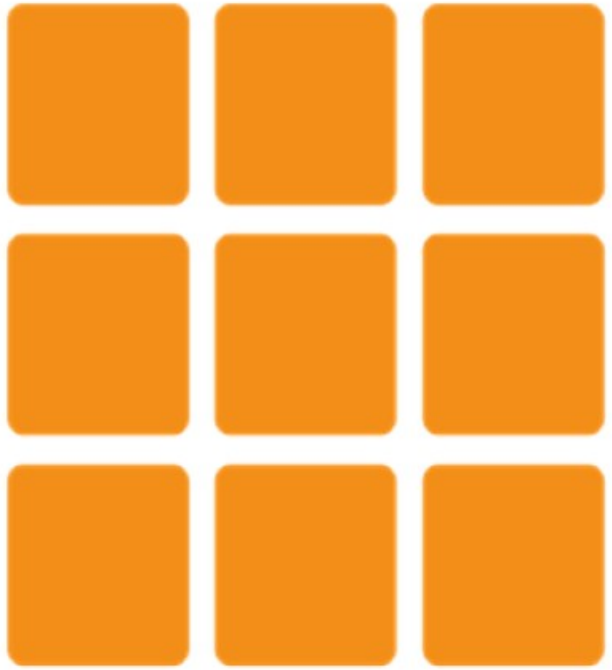
- Identity and Access Management Administration
- Deployment and Management of Relational Database Services, DynamoDB, and Aurora
- AWS Management Tools and what they are designed to do
- ELB load-balancing and Auto Scaling
- CoudWatch, CloudTrail, AWS Config
- Trusted Advisor
- DevOp Tools



Managed Services Concepts

AWS (Amazon Web Services)

- EC2 – Elastic Cloud Compute (Virtual servers)
- VPC – Virtual Private Cloud (Virtual network)
- EBS – Elastic Block Storage (Disk volumes)
- S3 – Scalable Cloud Storage (Buckets)
- S3 Glacier – Archive backup storage (Storage Vaults)
- RDS – Relational Database Service
- DynamoDB – Managed NoSQL DB
- Scale – Load-Balancing, Auto Scaling
- Automation – CloudFormation, Lambda
- Management - CloudWatch, CloudTrail, Trusted Advisor



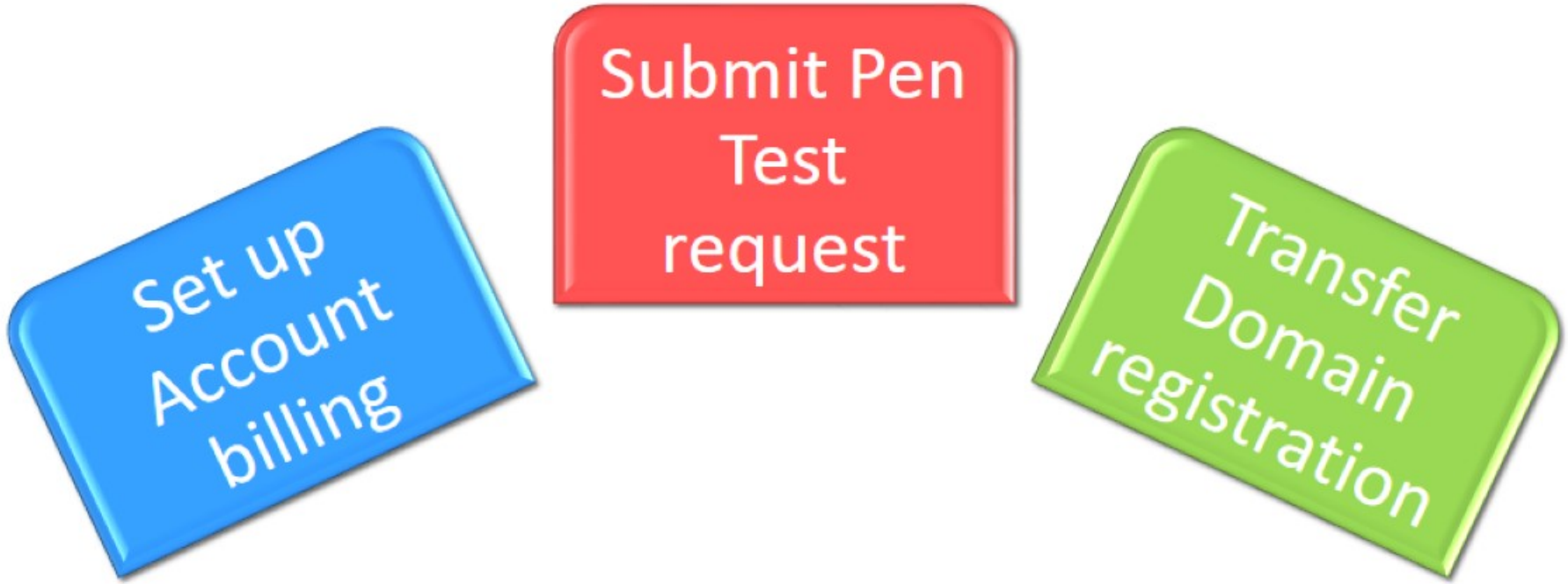
Root Account

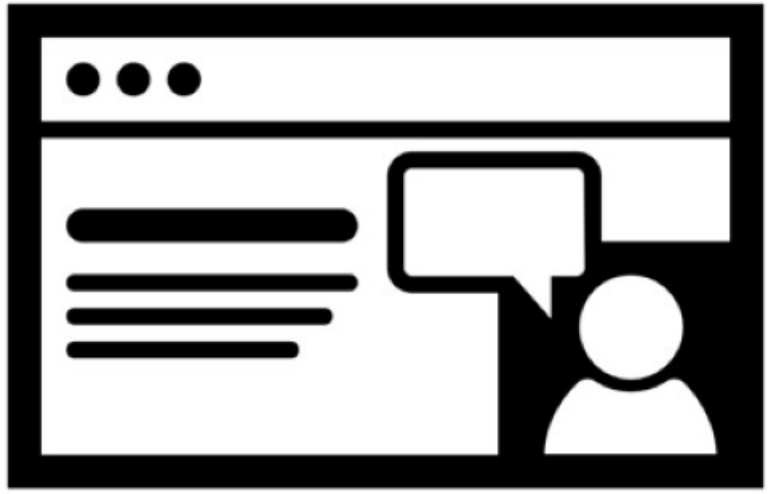
Root Account Creation

When you first sign up with Amazon Web services the first user account created is the Root account

- The root account can't be disabled
- Has full unrestricted access to all account resources
- Should not be used for administration
  - Store root account credentials in a safe location
  - Enable MFA on the root account
  - Disable root account usage by "forgetting" password

Root Account Tasks





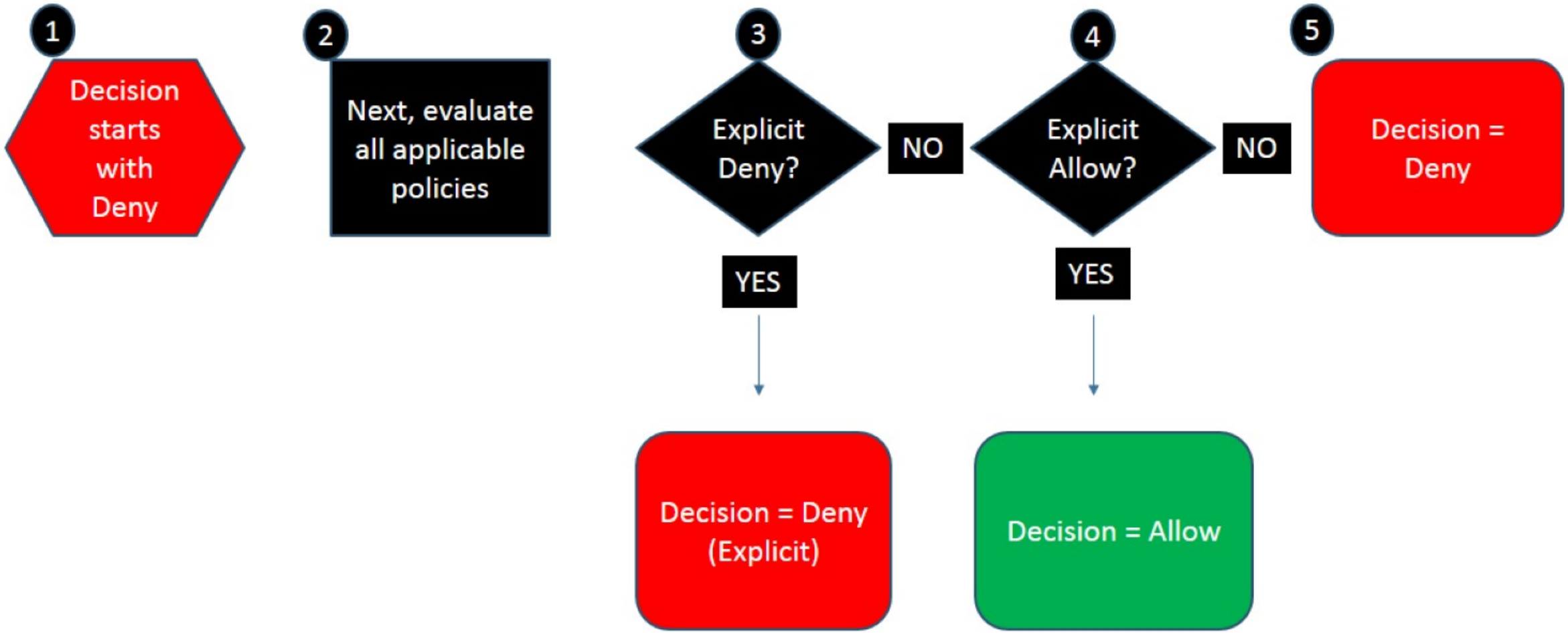
Exercise: The Root Account

IAM

Resolving IAM Permissions

Initially, all requests are implicitly denied

1. All policies are evaluated; if there is an explicit deny, processing stops, and the request is denied
2. If no explicit deny is found, and an explicit allow is found in any policy the request is allowed
3. If there is no explicit allow or deny permission found, then the default deny is maintained

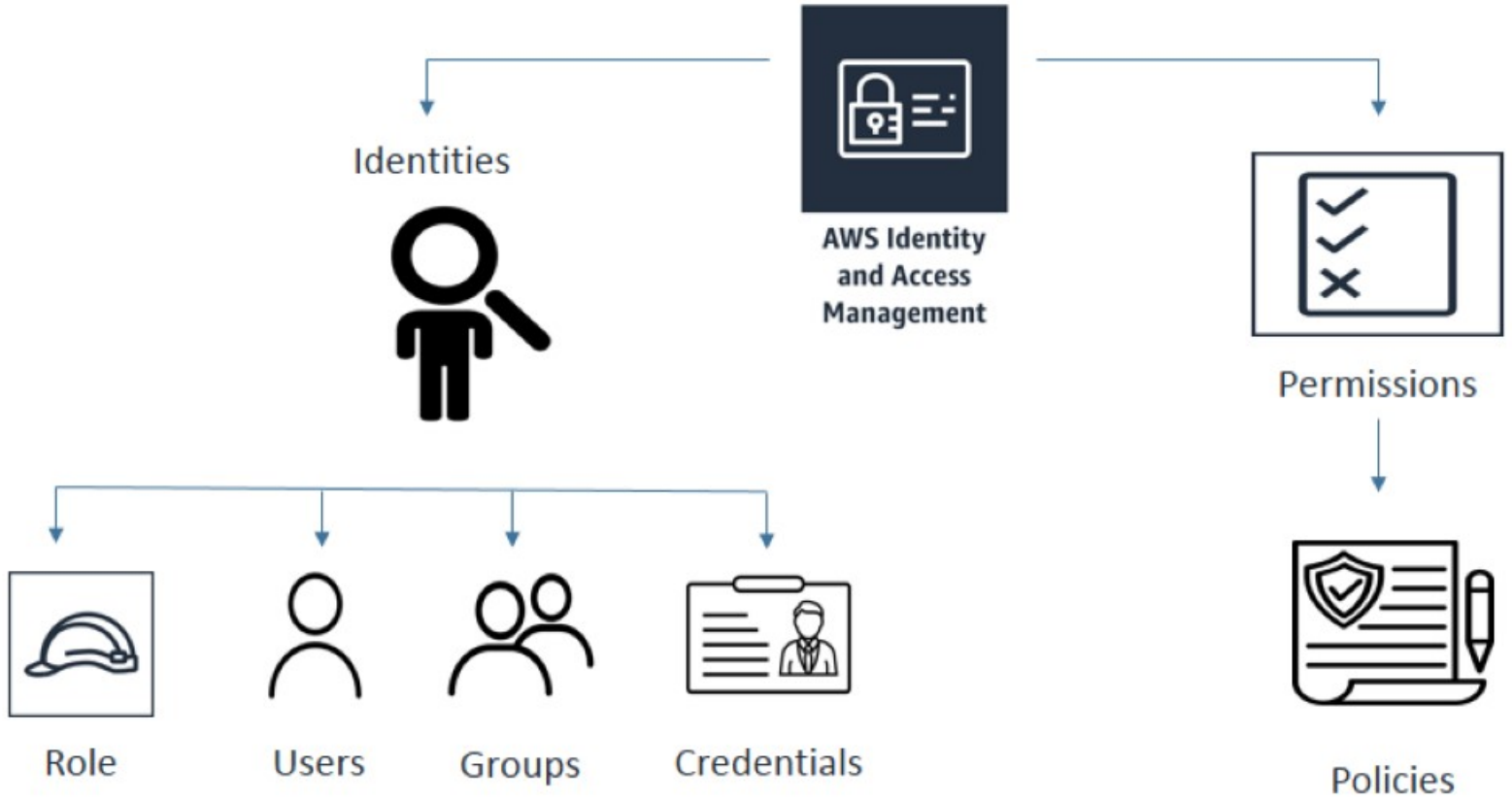


Policy Evaluation Logic

What is IAM?

Permissions to access or control AWS infrastructure

- IAM User – user with logon credentials
- IAM Groups – management of IAM users
- IAM Roles – grant specific privileges to applications and federated users for temporary access


Identities
AWS Identity and Access Management
Permissions
Role
Users
Groups
Credentials
Policies

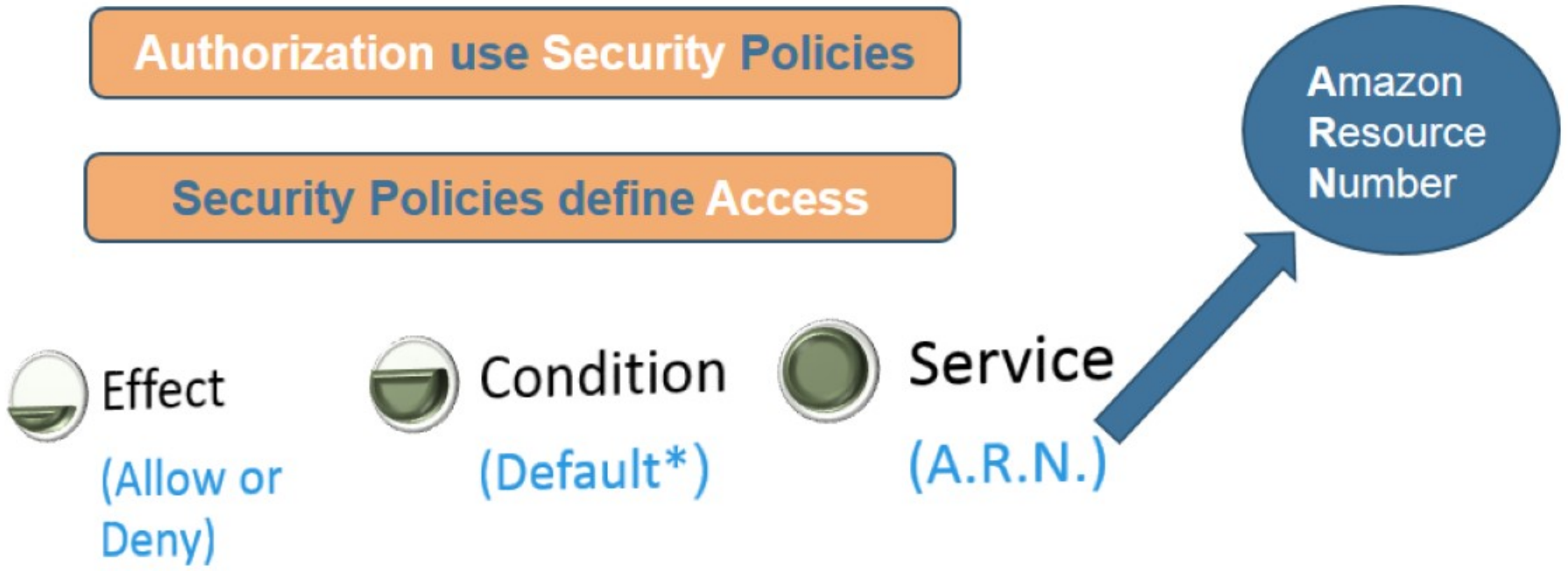
IAM User Authentication

- Username @ Password combination
  - Access key (Access key ID + Secret access key) combination
    - Access key + Session Token
- API calls to AWS must include the access key and session token to authenticate
- Create custom logon URL for easier access
- Add optional Multi-factor authentication

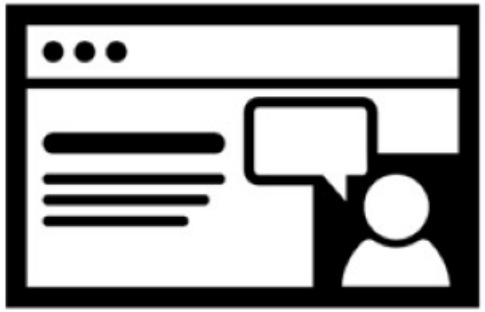
IAM Authorization

Authorization use Security Policies

Security Policies define Access



Conditions define limits, rather than * (Everything)

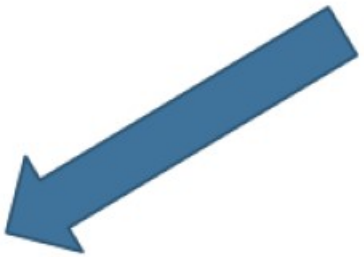


Exercise: IAM Users and Groups

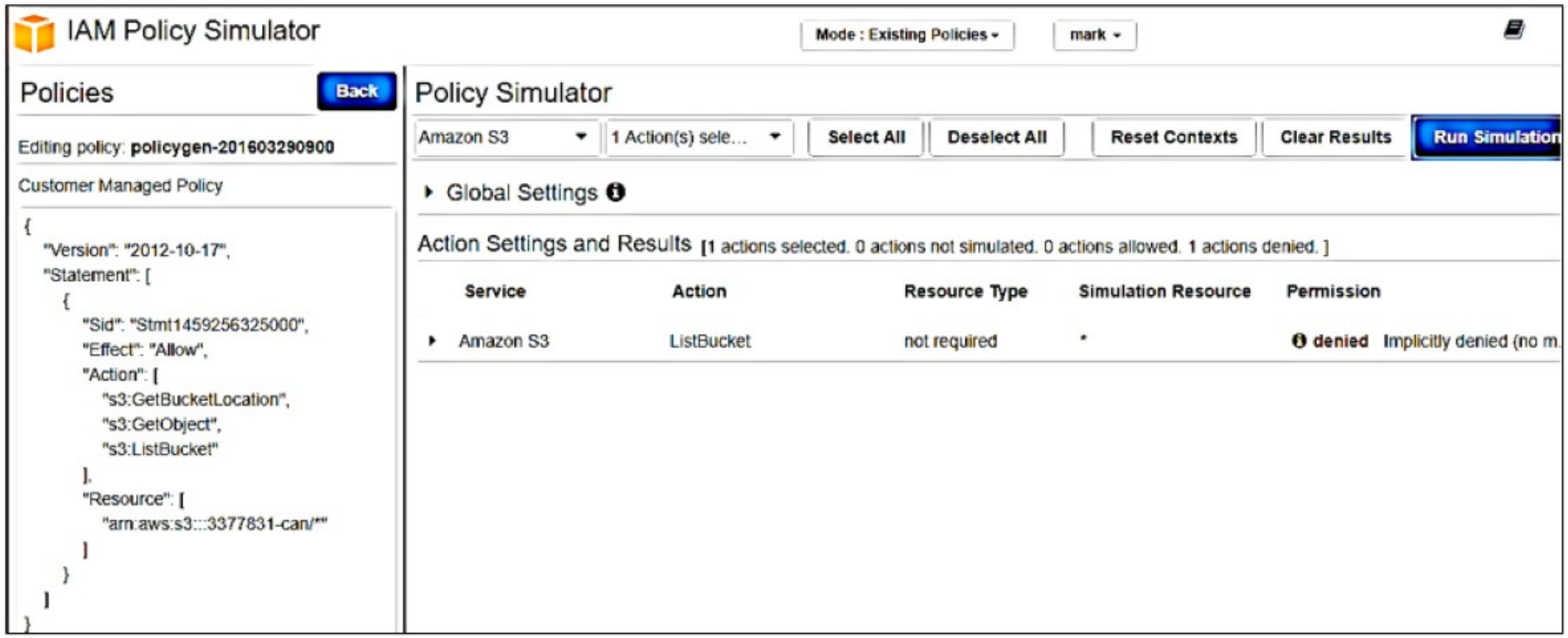
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:ListAllMyBuckets",
      "Resource": "arn:aws:s3:::*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation"
       ],
      "Resource": "arn:aws:s3:::productionapp"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": "arn:aws:s3:::productionapp/*"
    }
  ]
}
```

IAM Policy: S3 Access

IAM Policy: Allow EC2 instances to Attach or Detach Volumes

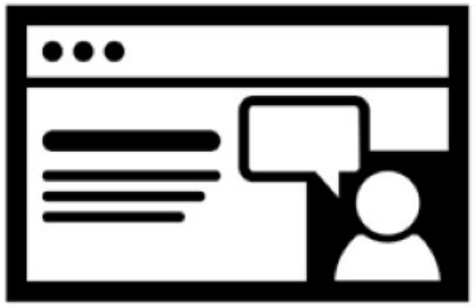


```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ec2:AttachVolume",
    "ec2:DetachVolume"
   ],
   "Resource": [
    "arn:aws:ec2:<REGION>:<ACCOUNTNUMBER>:volume/*",
    "arn:aws:ec2:<REGION>:<ACCOUNTNUMBER>:instance/*"
   ],
   "Condition": {
    "ArnEquals": {"ec2:SourceInstanceARN":
"arn:aws:ec2:<REGION>:<ACCOUNTNUMBER>:instance/<INSTANCE-ID>"
     }
    }
   }
  ]
}
```



Policy Simulator

HTTPS://POLICYSIM.AWS.AMAZON.COM/

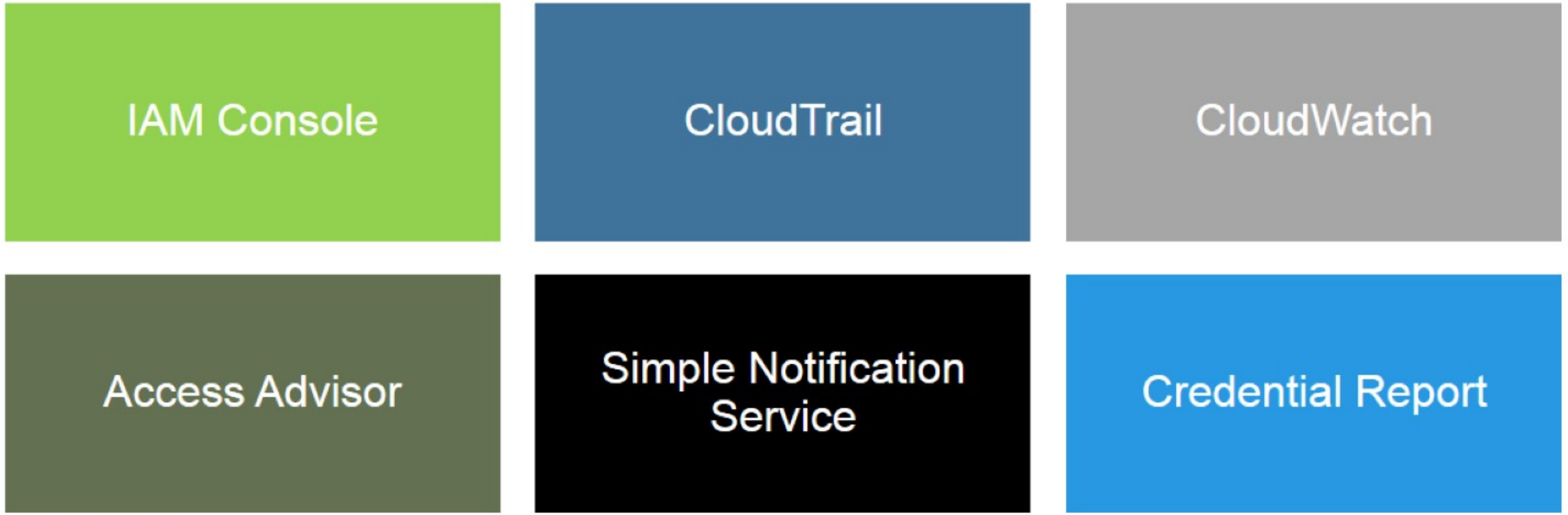


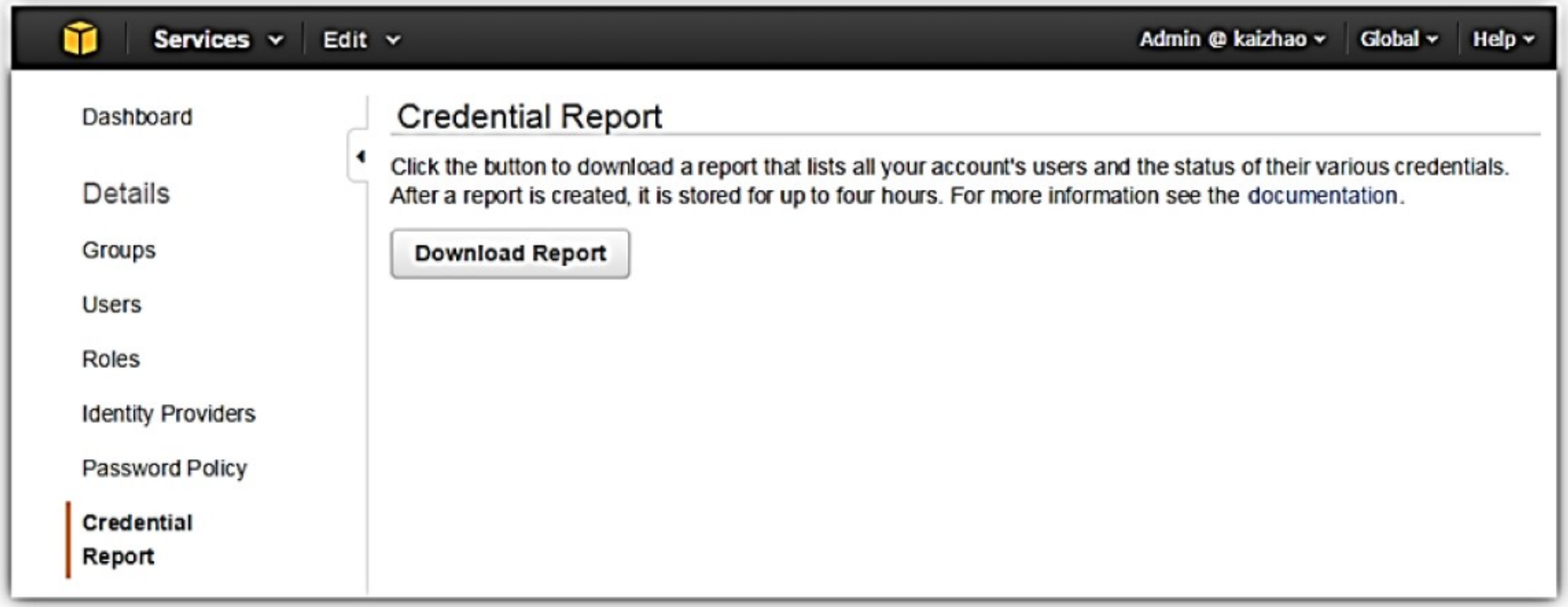
Exercise: Policy Simulator

Password Policy Options

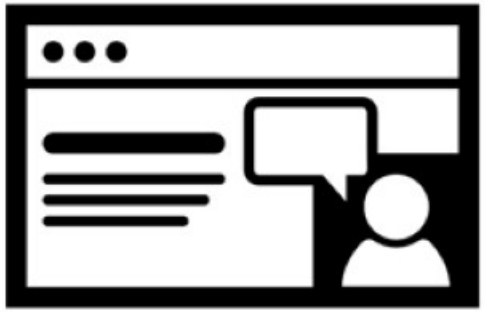
Minimum password length	Complexity	Set password expiration
Allow password reuse	Users can reset their passwords	Password expiration rules

Password Management





Credential Report



Exercise: Password Management

Associating Policies with Principals

(User, Group, or Role)

User Policy

- Policy attached to a specific user, or a specific group of users

Managed Policy

- Master list of pre-created policies that can be assigned per user, or group

Custom Policy

- Written by you, added to your AWS account as custom IAM policy

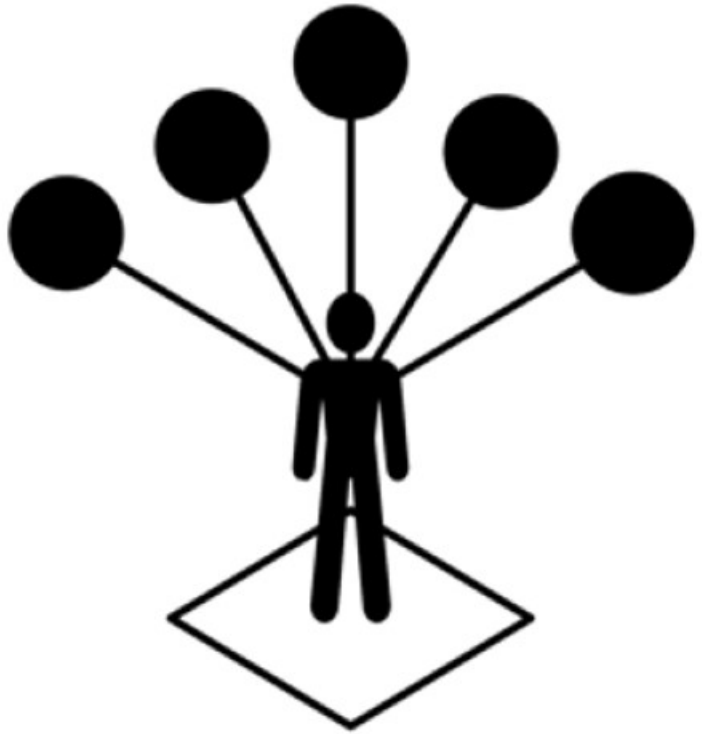
IAM Roles

The STS provides a temporary security token used to access AWS cloud services

Temp security token lifetime – 15 minutes to 36 hours

- IAM Roles are used for:
  - EC2 Roles – permissions for applications
  - Federation – granting permissions to users authenticated by a trusted external system (Google, Facebook)
  - Cross-Account Access – granting permissions to users from other AWS accounts

Using temporary access tokens means no fixed access keys need to be maintained, and rotated



IAM Roles (FYI)

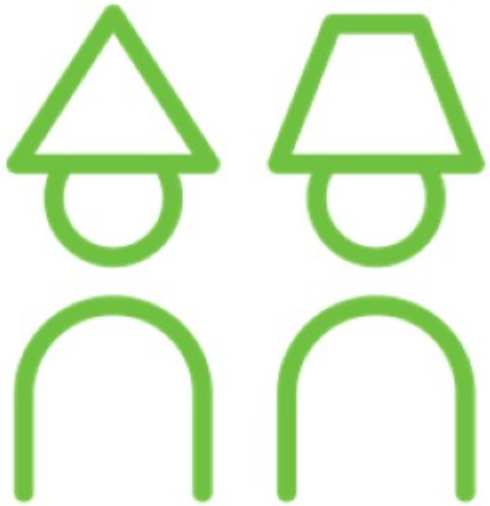
- IAM roles cannot make direct requests to AWS services by themselves
- IAM Roles do not have any credentials
  - Temp credentials are controlled by the security token service

Roles are assumed by:

- IAM users from other AWS accounts
- Applications hosted on EC2 Instances
- Federated users that authenticated externally

IAM Roles with EC2 Instances

- Simplify management and deployment of access keys to EC2 Instances
- Associate an IAM Role to EC2 Instance
- EC2 Instance metadata hosts the temporary security credentials used by application
- Application makes calls to AWS service defined by the IAM role using temporary credentials

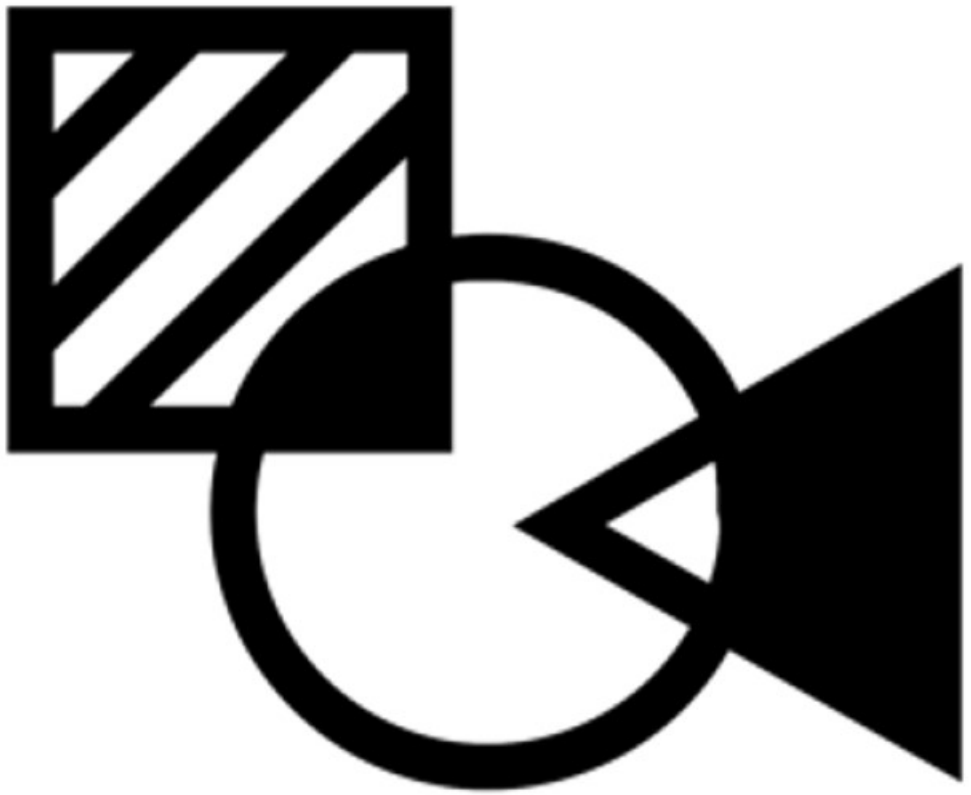


Who can Assume a Role ?

- As an authenticated IAM user: “a person”, “a process”
- A person, or process that has been authenticated by an external trusted service outside of AWS (LDAP or Web authentication)
- After the “actor” has assumed the role, it is provided with a temporary security token that is associated with the security policy of that role

The provided temporary token contains the details for authenticating and access to the AWS service along with:

- Access key
- Session token used for authenticating calls under the assumed role



Roles and Cross Account Access

- Setup permissions with assigned IAM role
- Users then **assume the role** to access resources in linked AWS account

Cross Account Access

Production account = Live applications

Development account = Test applications

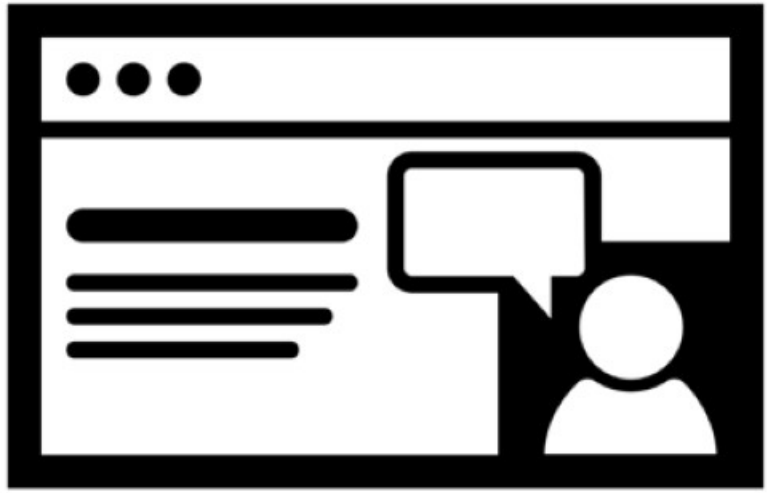
Problem: Developers need to update live applications in the production account

Solution: Use roles to delegate access to resources using Cross Account Access

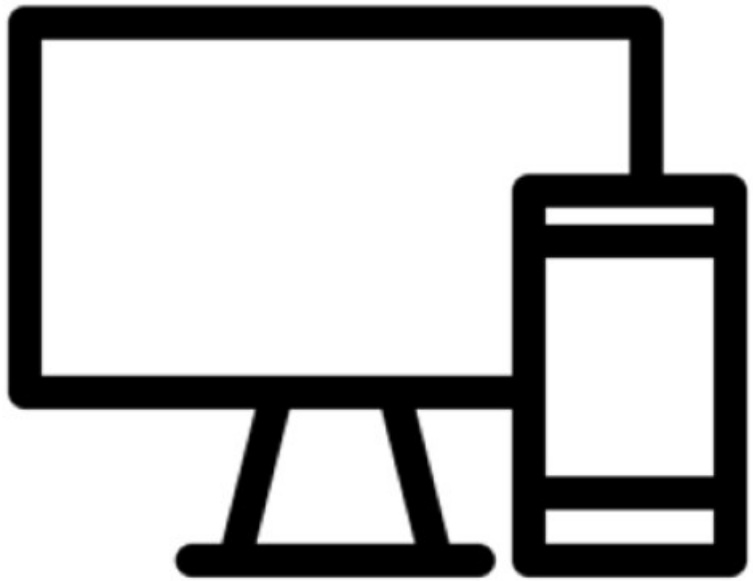
Production account : Trusting account

Development account: Trusted account

Solution: Role for Developers to access app in Production



Exercise: Create an IAM Role

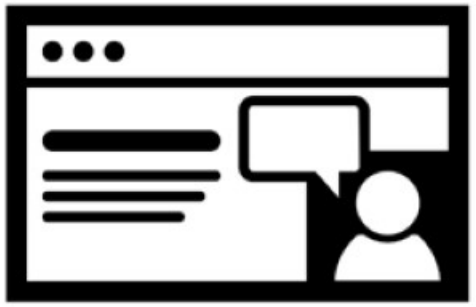


Multi-factor Authentication

MFA adds a second layer of authentication using an OTP, a one-time-password

Uses either a hardware or software device

1. Something you know
2. Something you have



Exercise: Setup MFA

Amazon RDS

Databases at AWS

Relational

Read / Write using SQL commands

Tables (columns, rows)

Online Analytical Processing (OLAP)

High query rate 64K IOPS

Vertical Scaling

Data Warehousing

Central store from other sources

Reports and searches

Online Analytical Processing

Batch processing

NoSQL

Non-relational

Key/value or document stores

Flexible schema

Online Transactional Processing (OLTP)

Horizontal scaling

Amazon RDS

Manage DB instance via API calls

Built using EBS volumes with IOPS

Automated backups and failover

Automatic Snapshots

Multi-AZ Deployments

Scale Up or Out

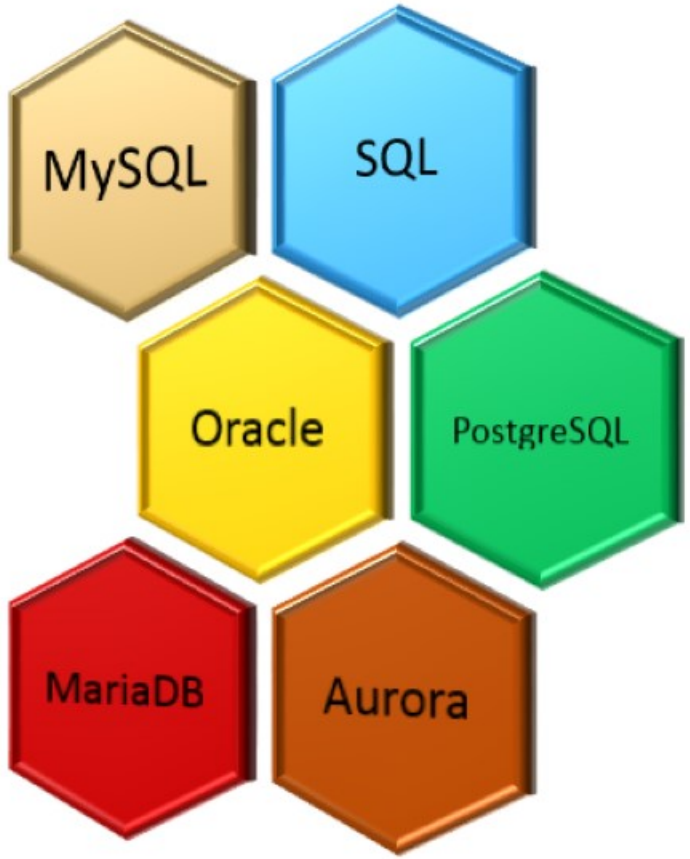
Database Instance (FYI)

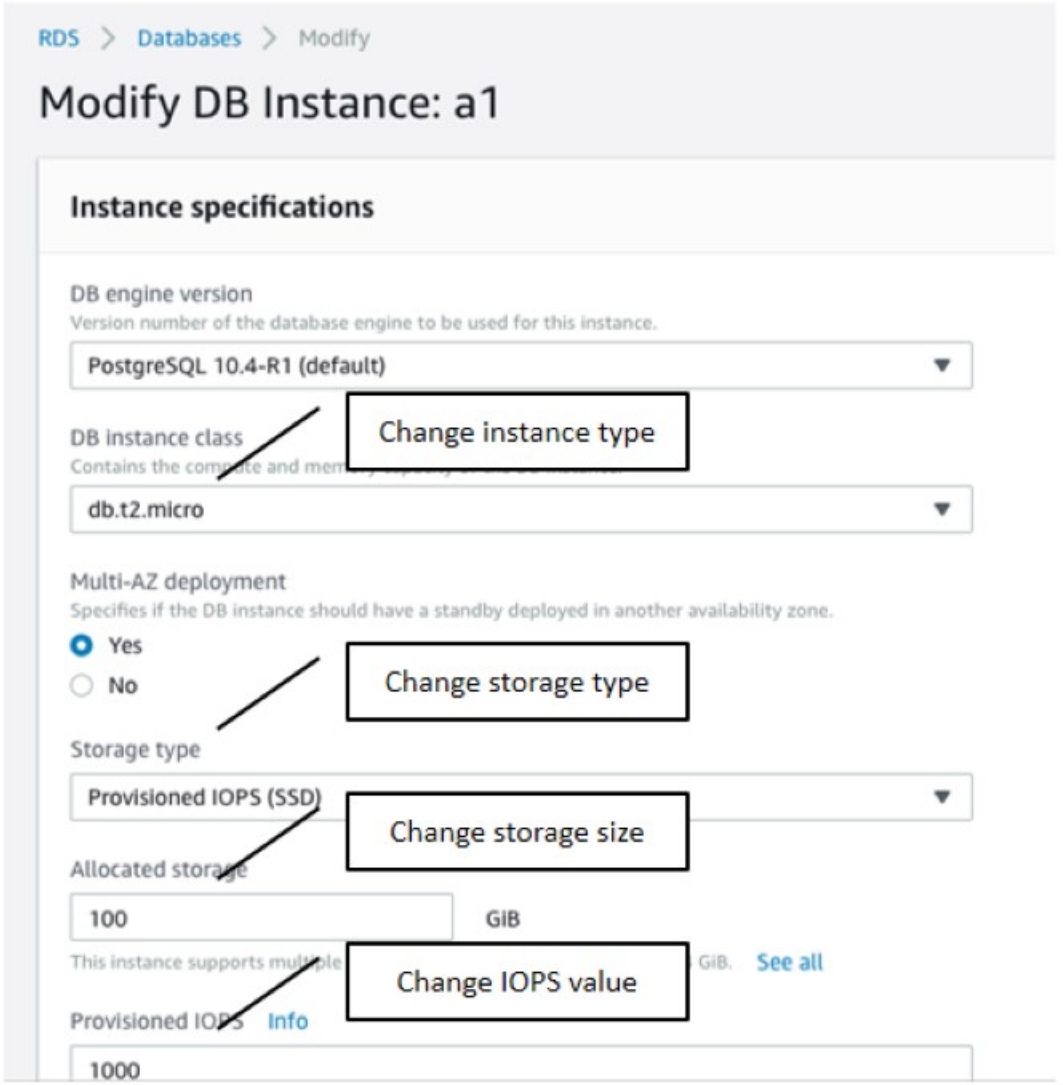
RDS uses managed DB instances (master, standby)

- Deployed on your private network (VPC)
- DB compute and memory needs defined by instance family
- DB instance parameters can be changed or resized
- There is no direct access to the database instance
- Connect to DB using common administration tools (Ex: SQL Tools)

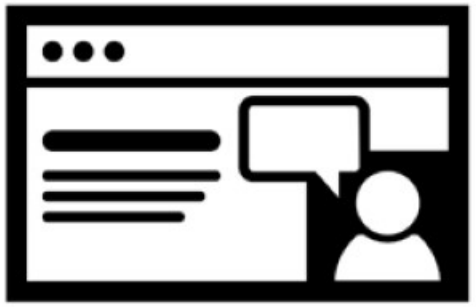
Existing databases can be migrated to AWS using the Database Migration Service

RDS Platforms





Changing DB Instance Parameters



Exercise: Setup RDS

Automating RDS Backups

You define the backup window for rollback time-frame

- RDS service creates volume snapshots of your database instance
- If database is multi-AZ; snapshot is taken from the standby
- RDS service backs up transaction logs every five minutes
- Single AZ deployment – multiple backup copies in AZ
- Multi-AZ deployment – multiple backup copies in multiple AZs

Manual RDS Snapshots

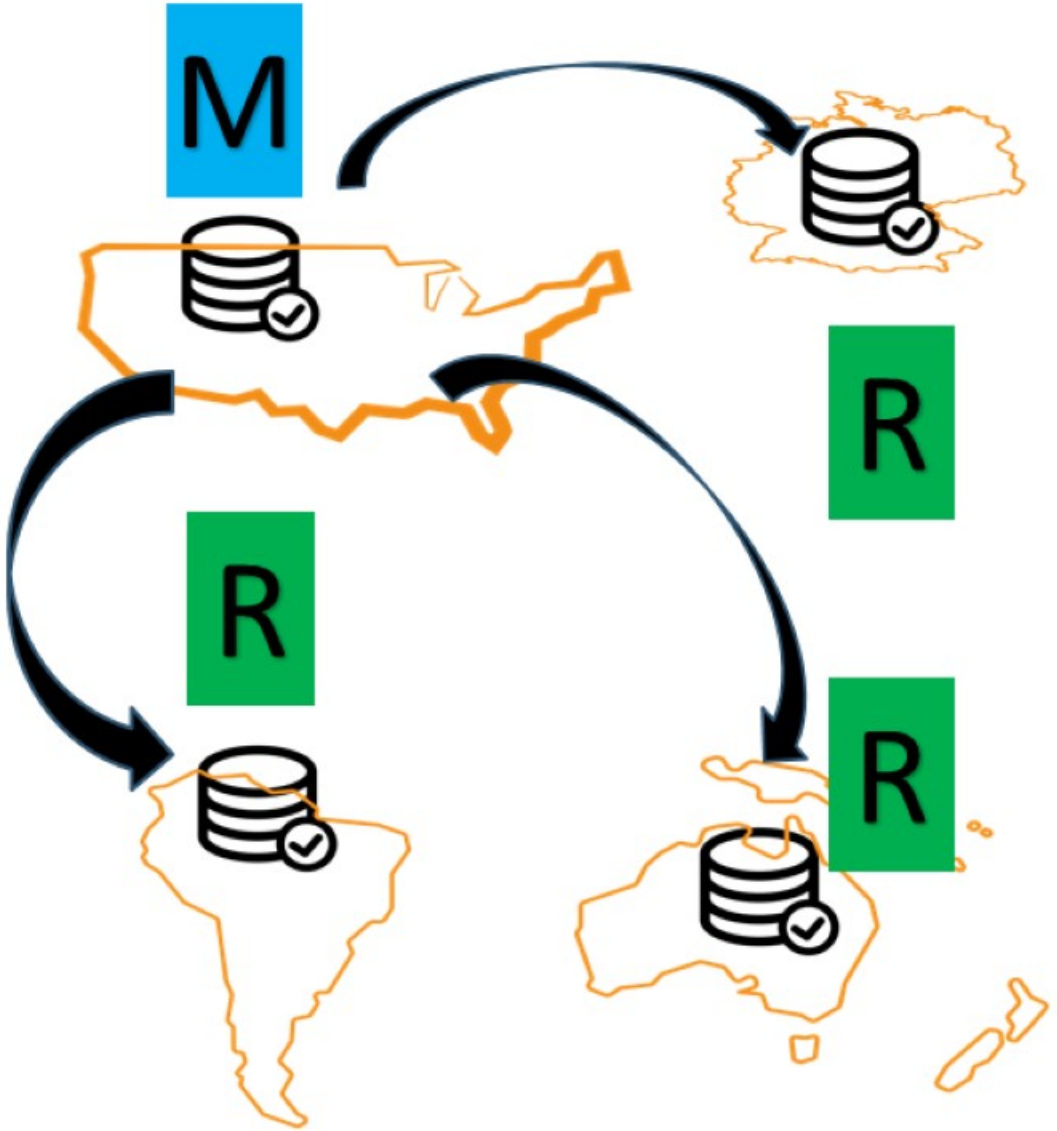
- A complete copy of your RDS database instance
- Independent of scheduled backups
- Can be used to create new RDS instance
- Taken from standby when design is multi-AZ

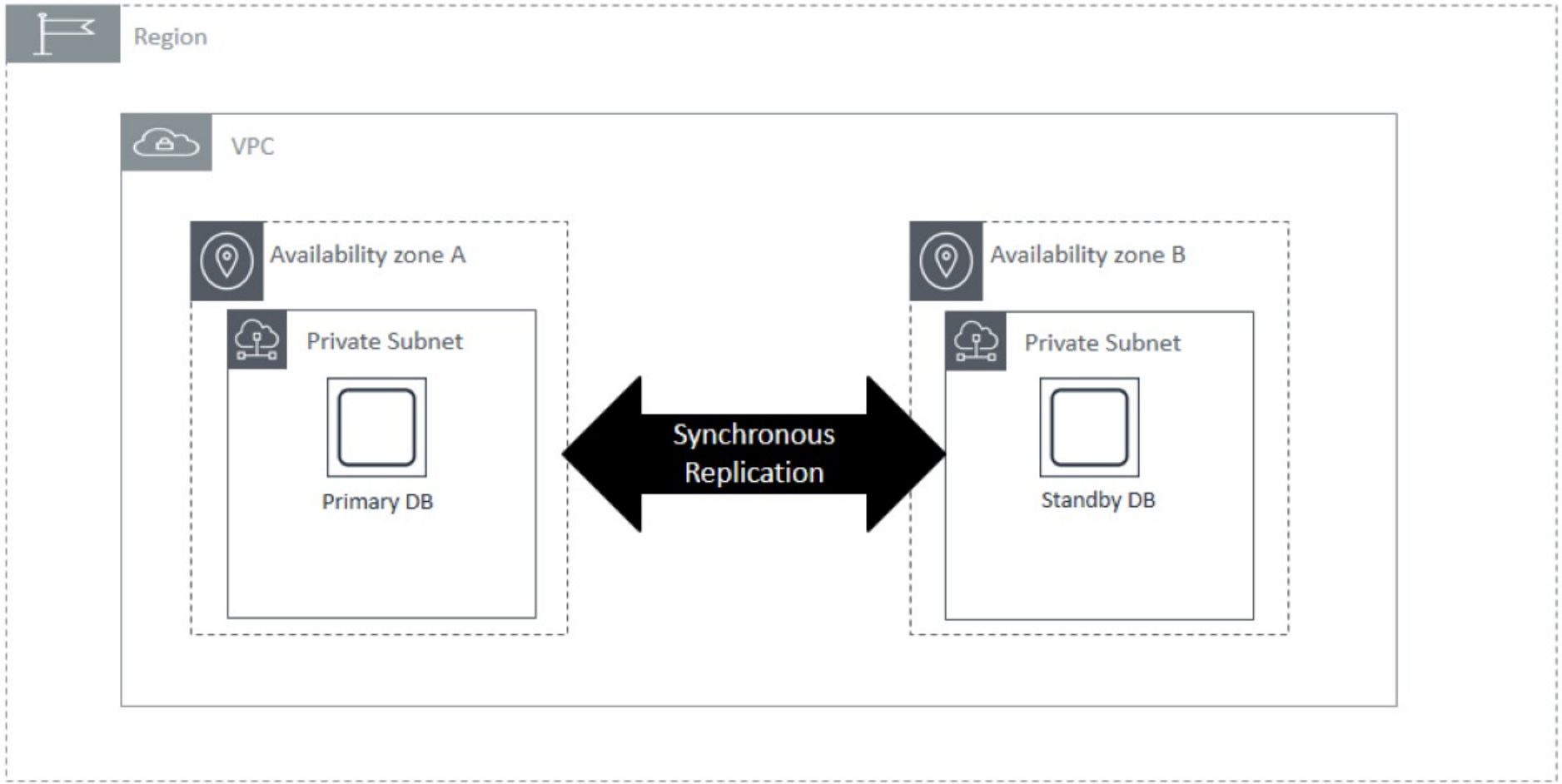
Read Replicas

Bring data closer to your customers within or across regions

Standby database cannot perform reads

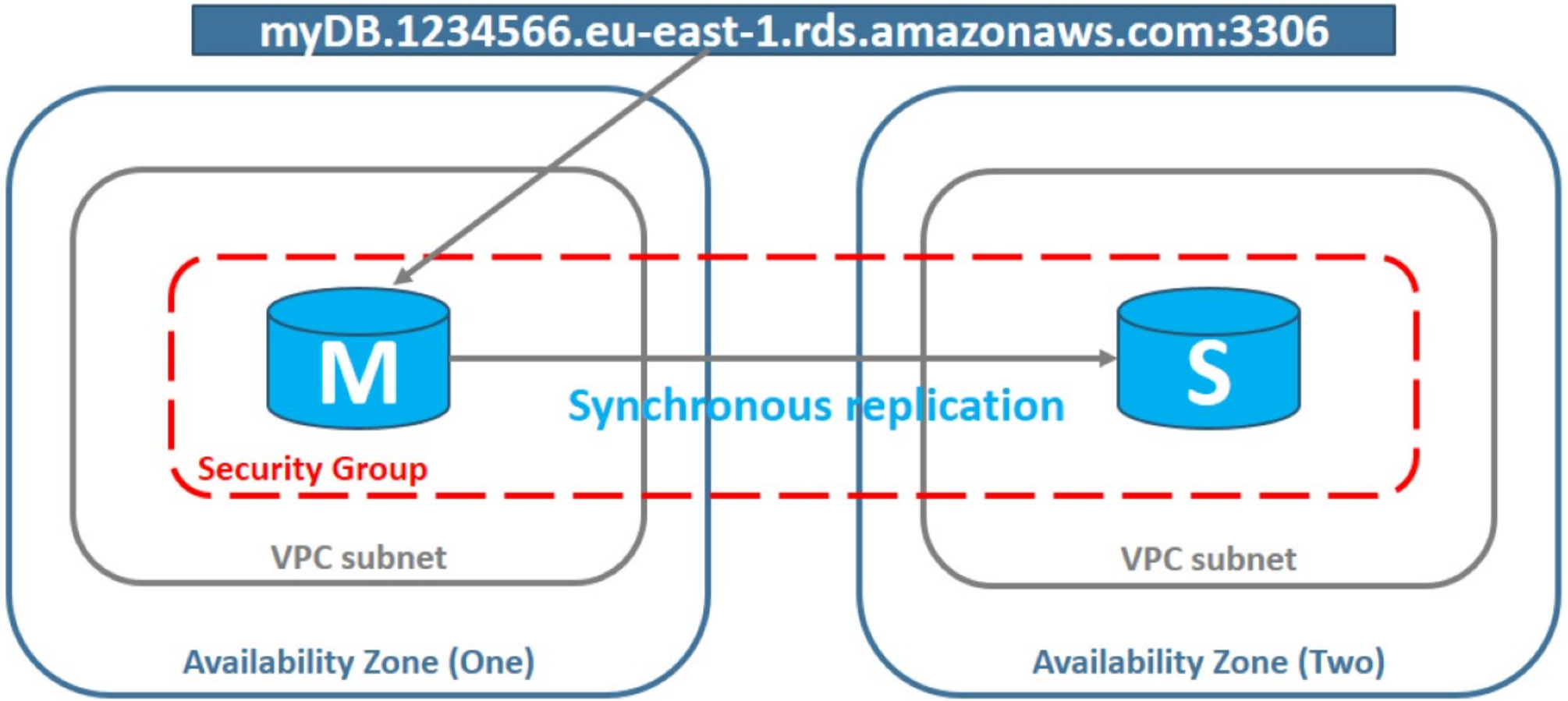
- Remove overload from your master database node
- Promote a read replica to a master node as additional recovery option, or for testing purposes
- Create up to five read replicas per master DB instance
- Available in MySQL, PostgreSQL, MariaDB, and Aurora

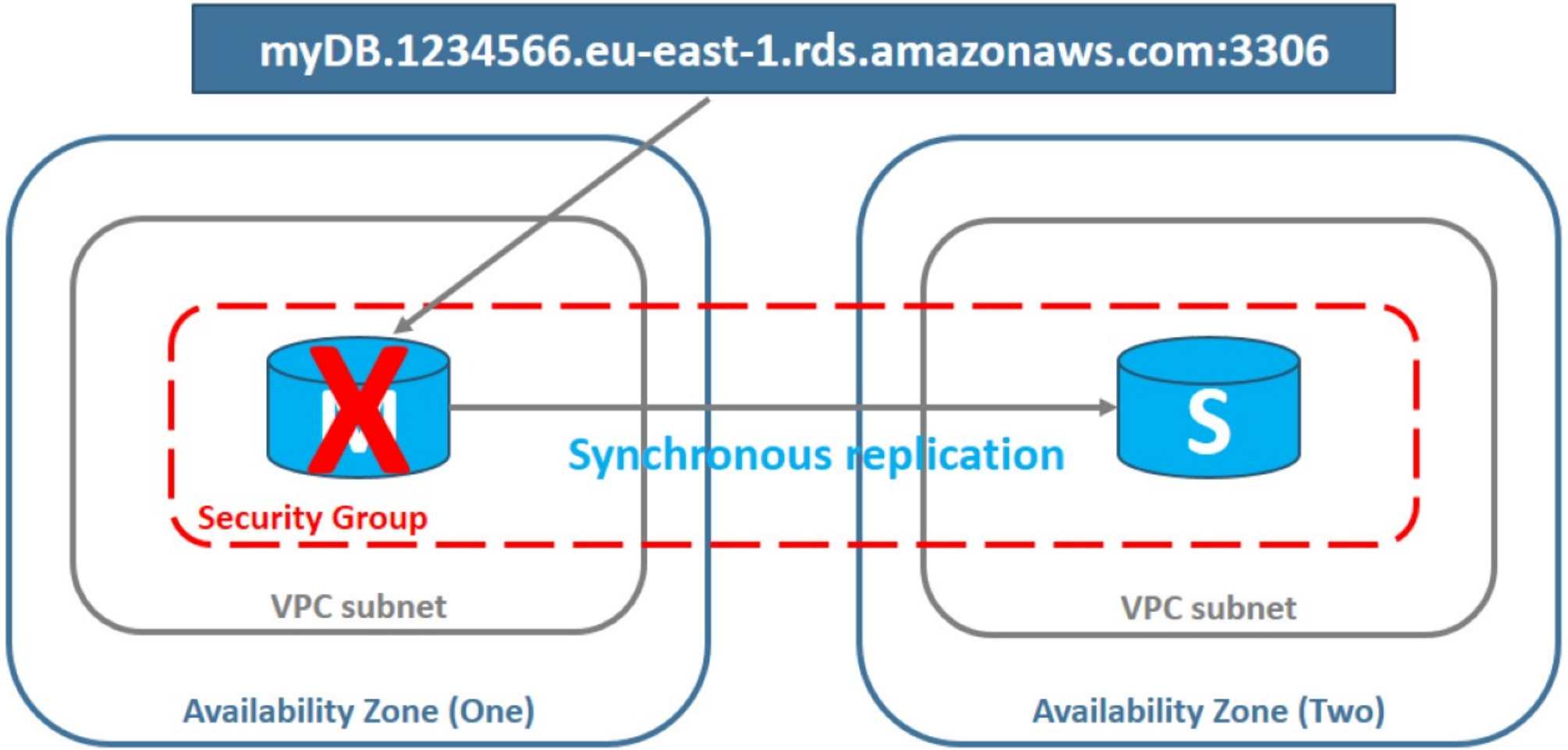

M
R
R
R

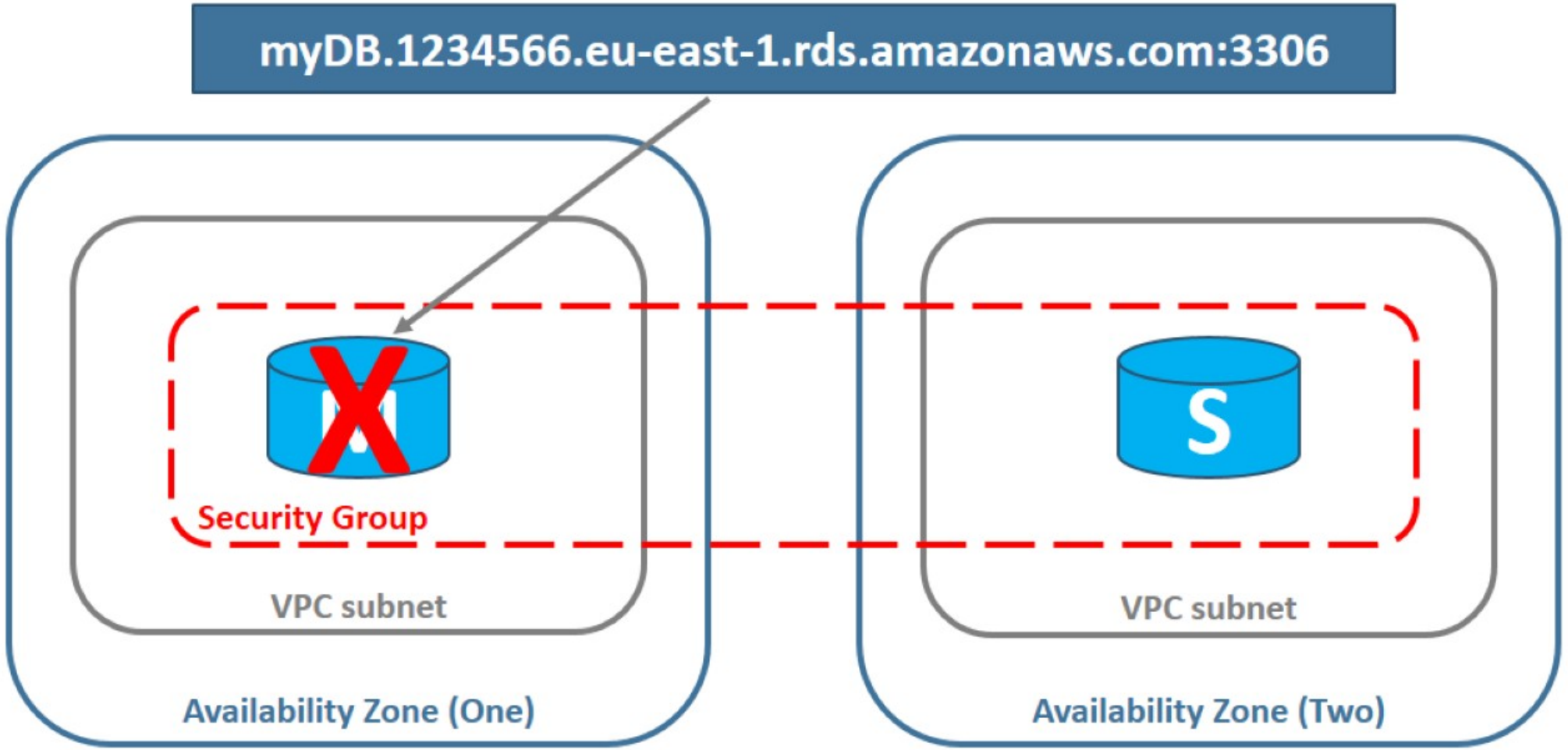


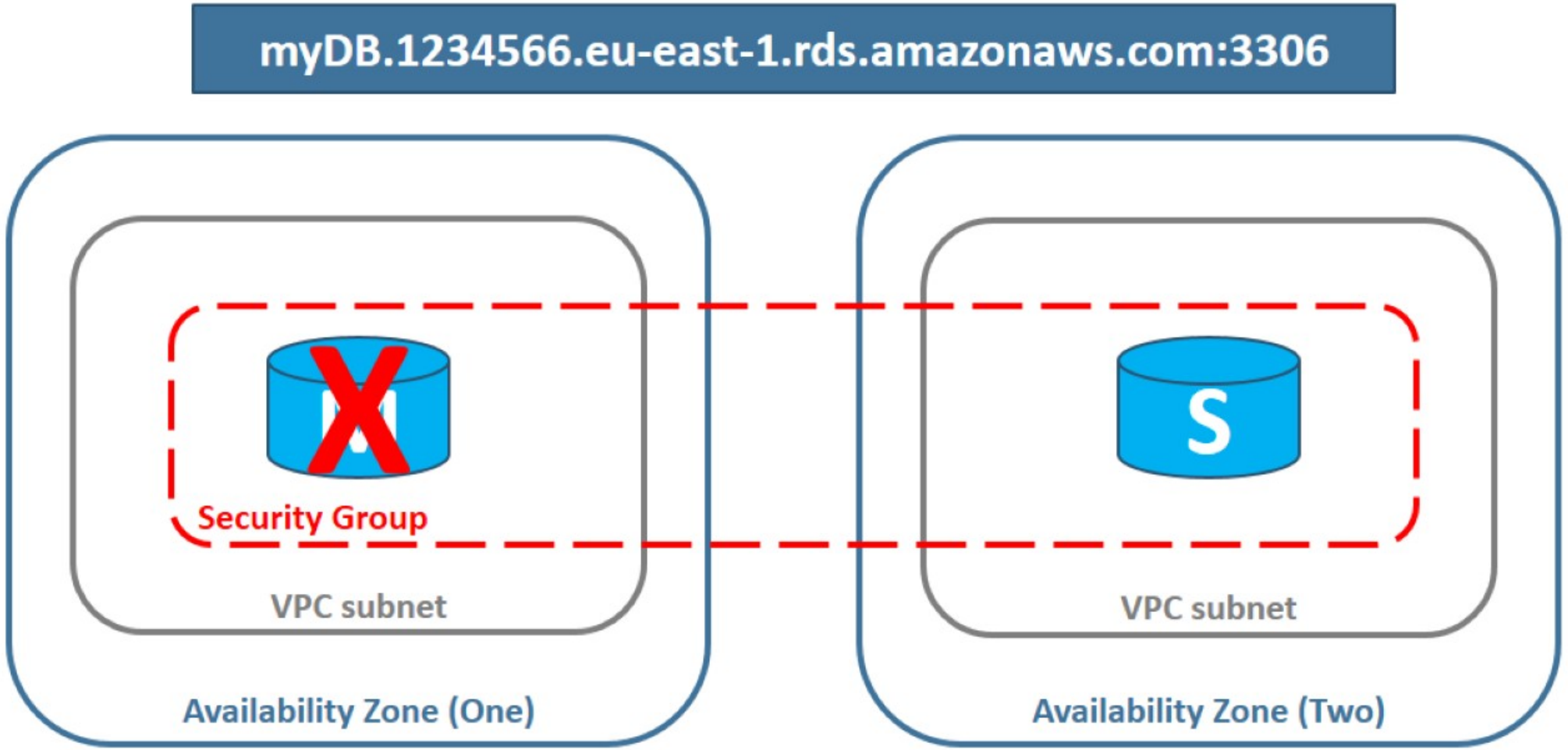
RDS Multi-AZ Design

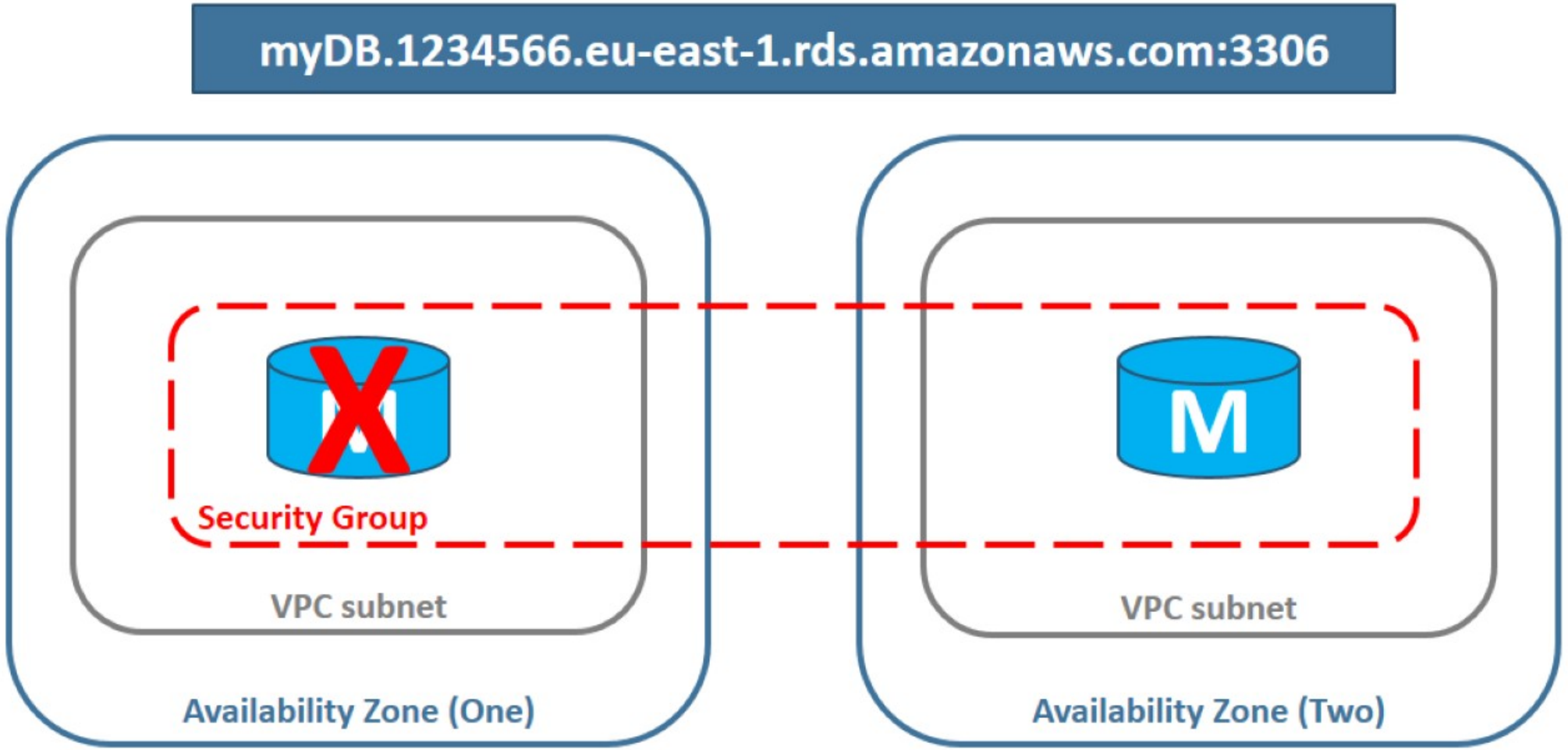
High-Availability: Multi- AZ

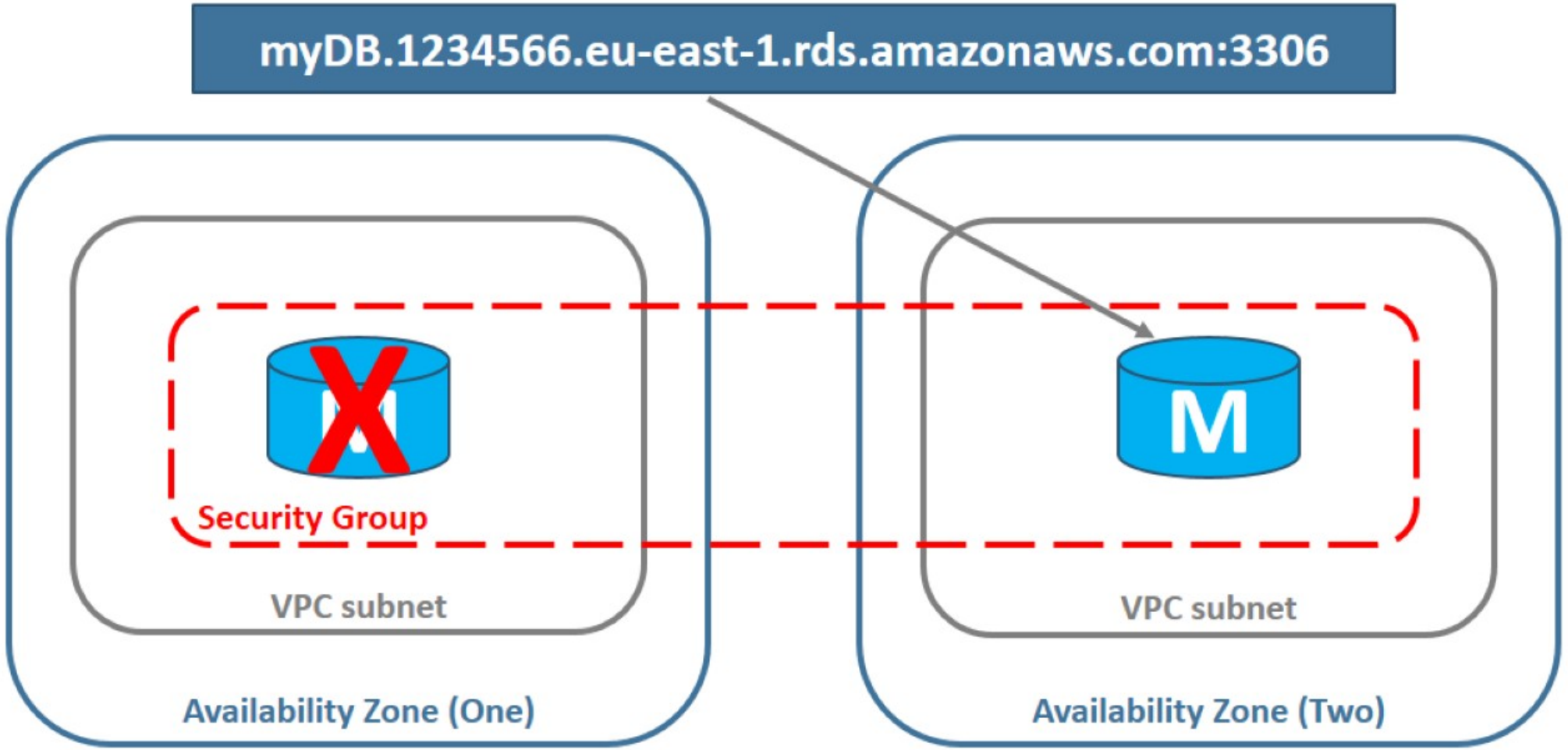


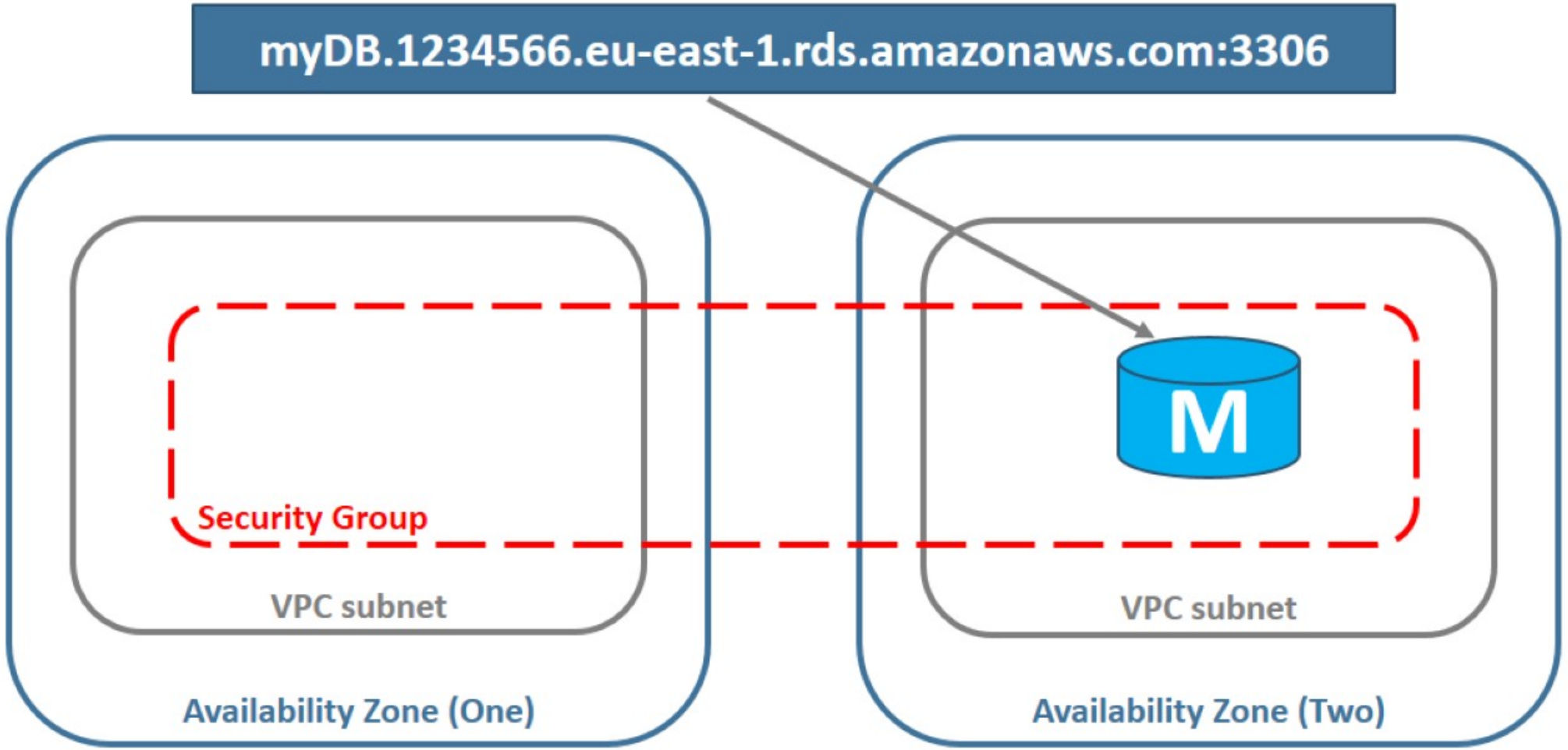

myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
S
Synchronous replication
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)

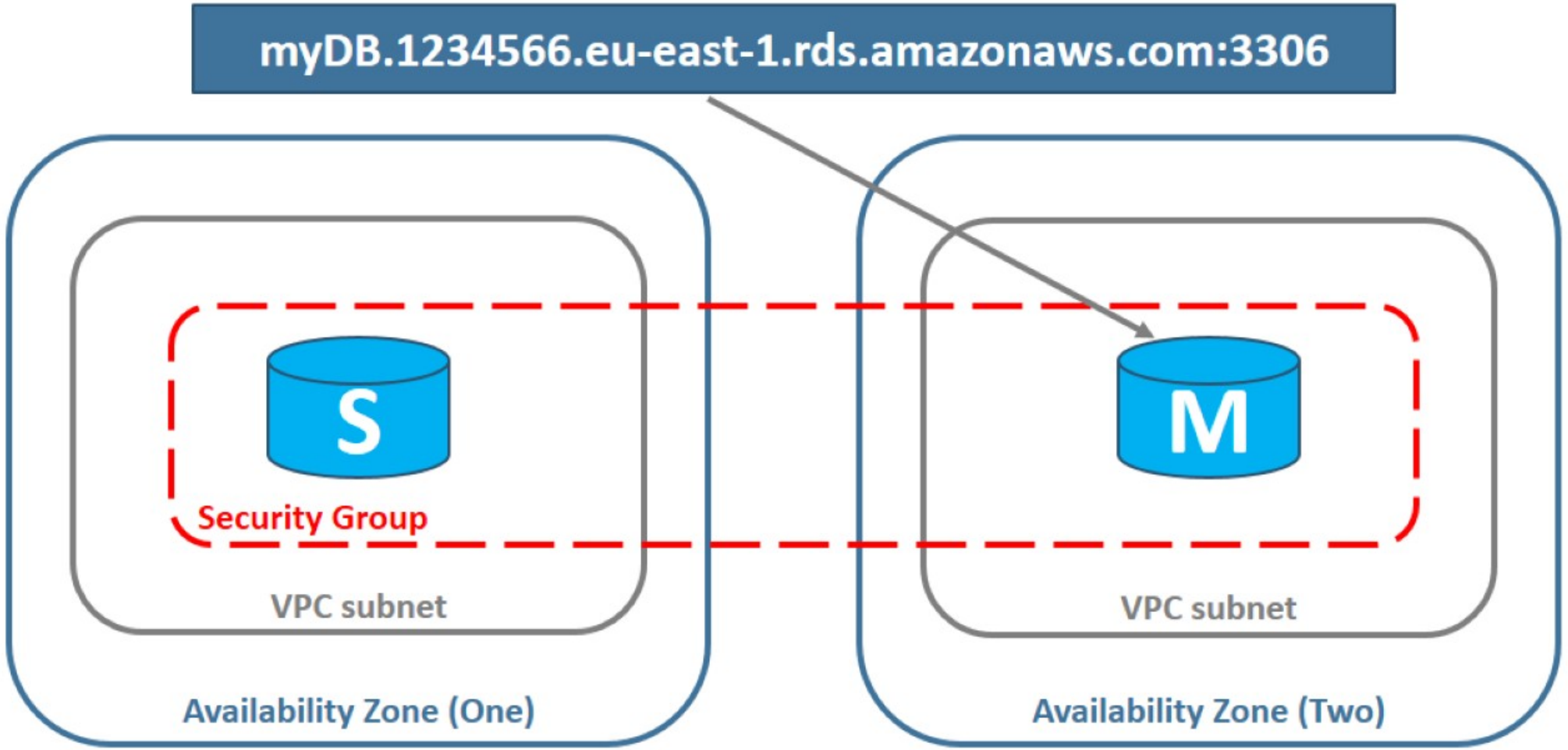

myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
S
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)

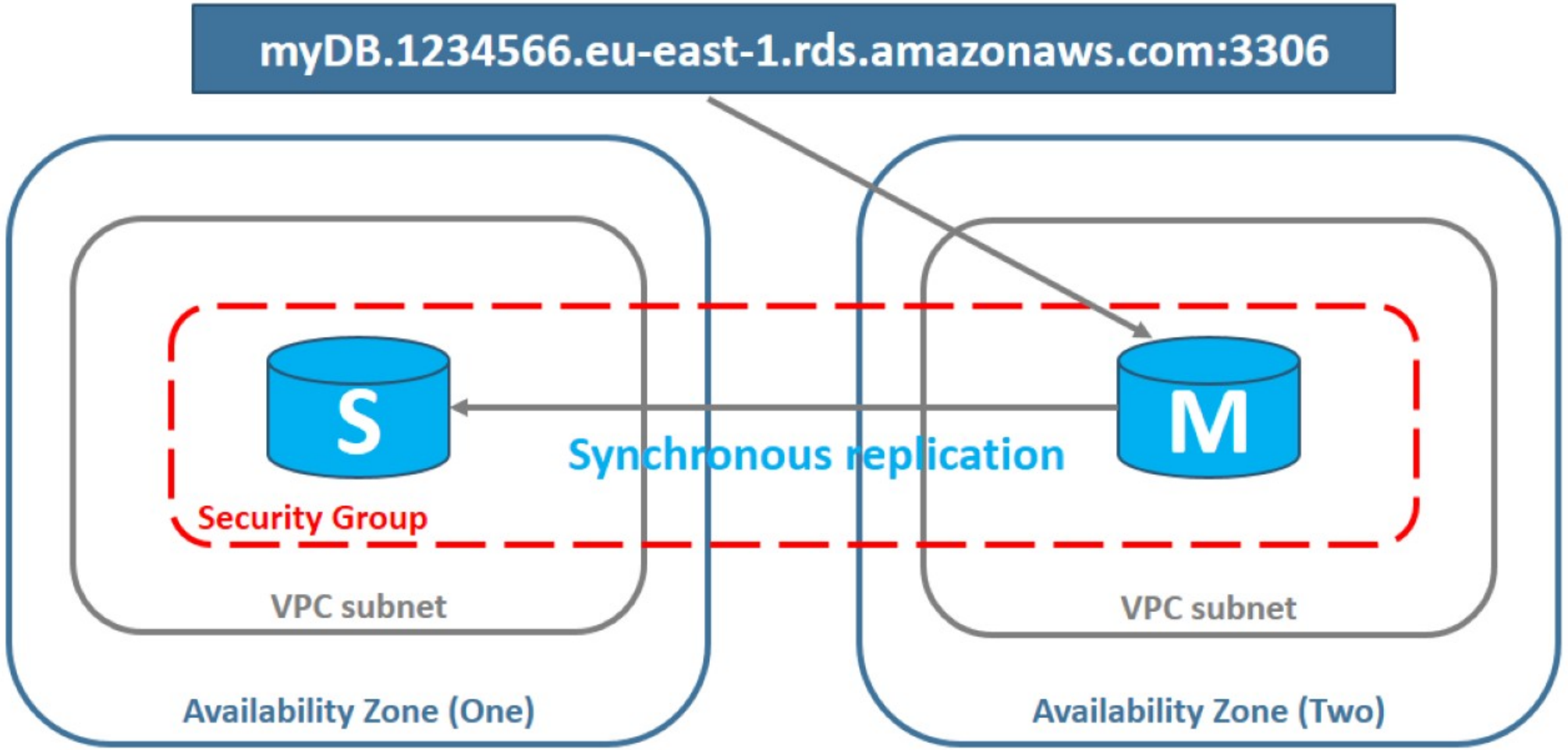

myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
S
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)


myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
Security Group
VPC subnet
Availability Zone (One)
M
VPC subnet
Availability Zone (Two)

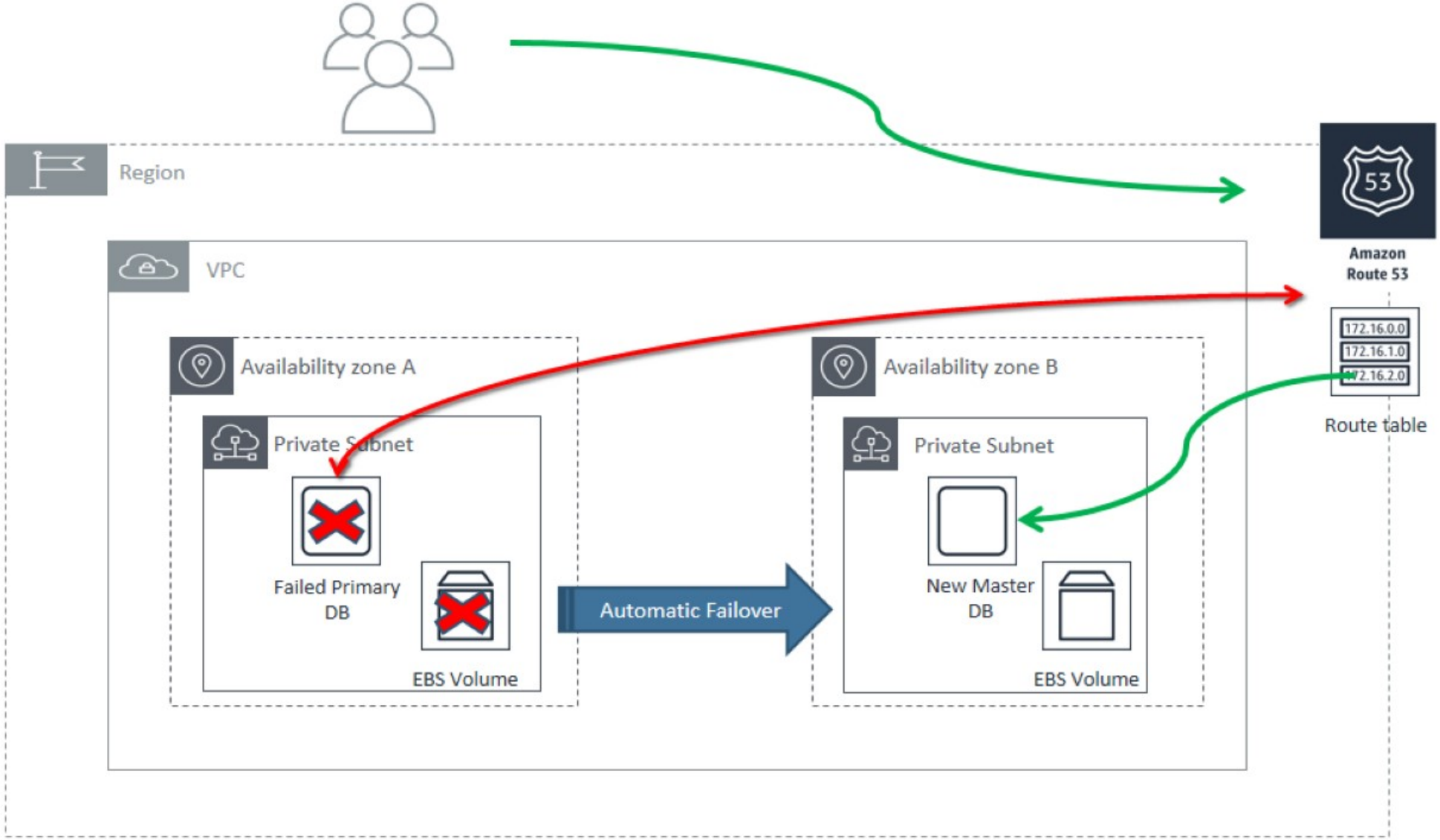

myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
Security Group
VPC subnet
Availability Zone (One)
M
VPC subnet
Availability Zone (Two)


myDB.1234566.eu-east-1.rds.amazonaws.com:3306
M
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)

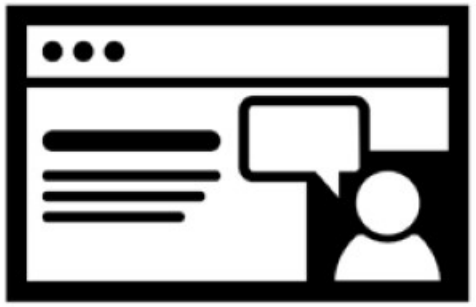

myDB.1234566.eu-east-1.rds.amazonaws.com:3306
S
M
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)


myDB.1234566.eu-east-1.rds.amazonaws.com:3306
S
M
Synchronous replication
Security Group
VPC subnet
VPC subnet
Availability Zone (One)
Availability Zone (Two)

Failover is NOT instantaneous



RDS Failover

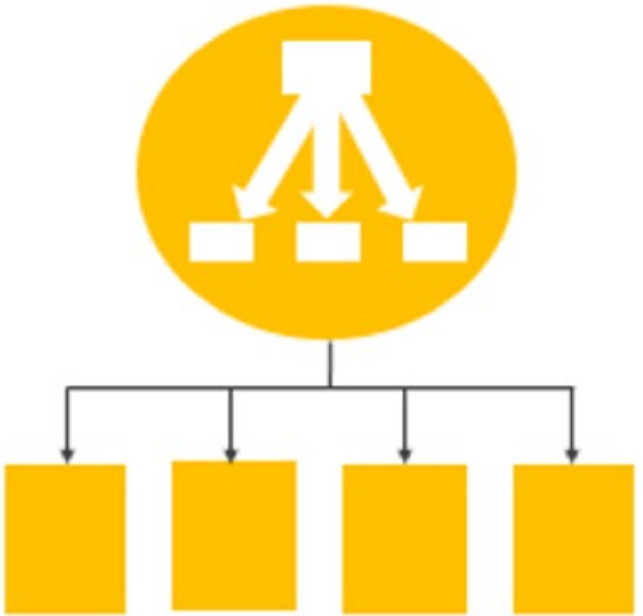


Exercise: Setup MySQL

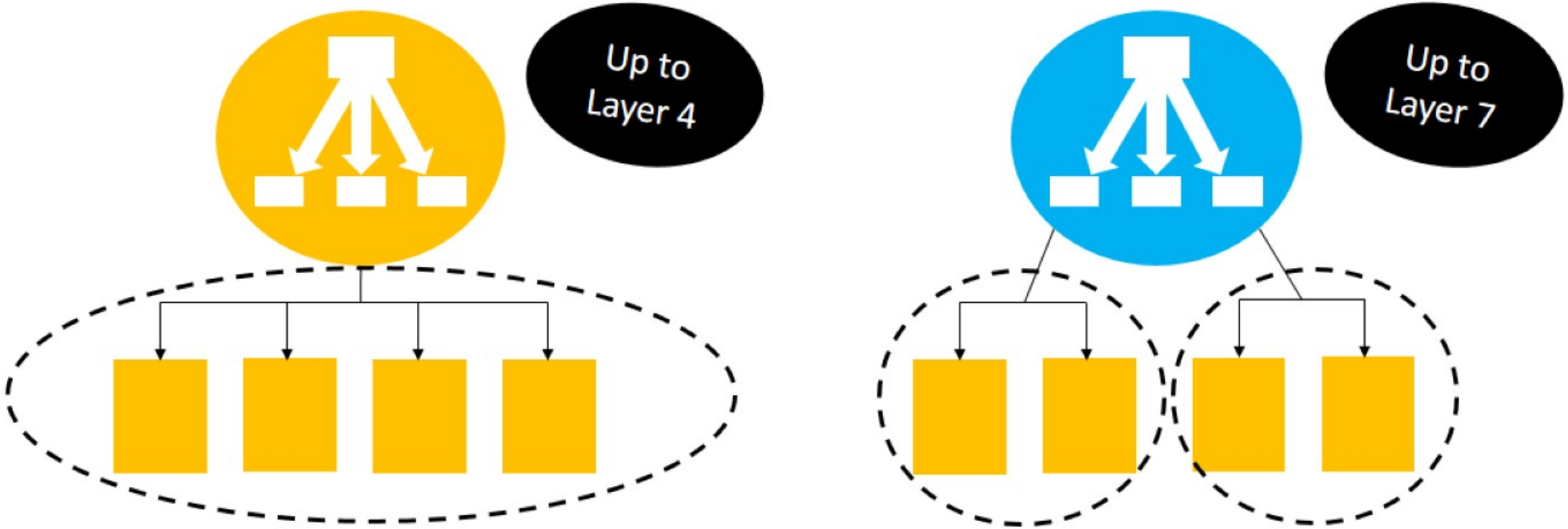
Load Balancing

ELB Load Balancing Service

- Distribute incoming traffic across multiple EC2 instances or containers
- Across one, or multiple availability zones
- Charged for ELB and traffic flow
- Classic load balancer: Public Web tier load-balancing (Classic NW)
- Network load balancer: Private web tier load-balancing (VPC)
- Application load balancer: Use with instances or container-based architecture (Docker)
- Best Practice: Integrate ELB with Auto Scaling groups



Load Balancing Options

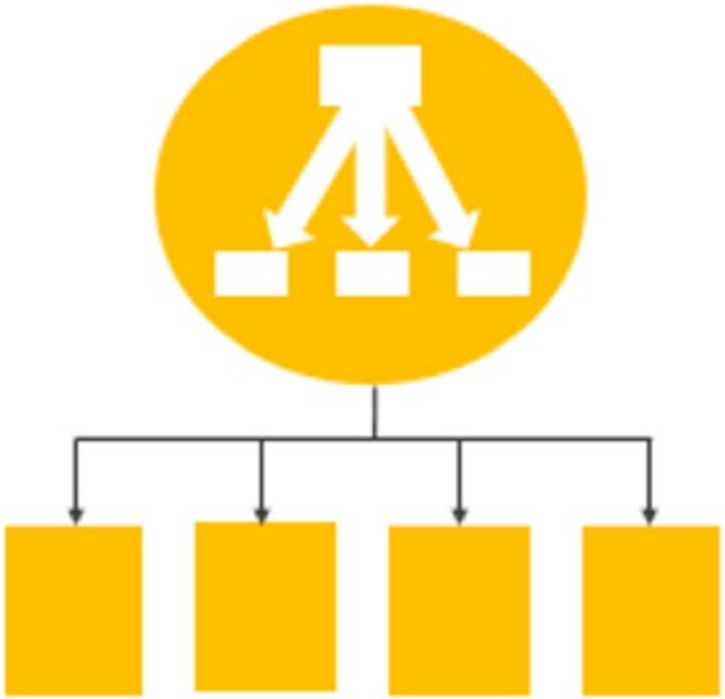


Network Load Balancer

Traffic balanced across instances

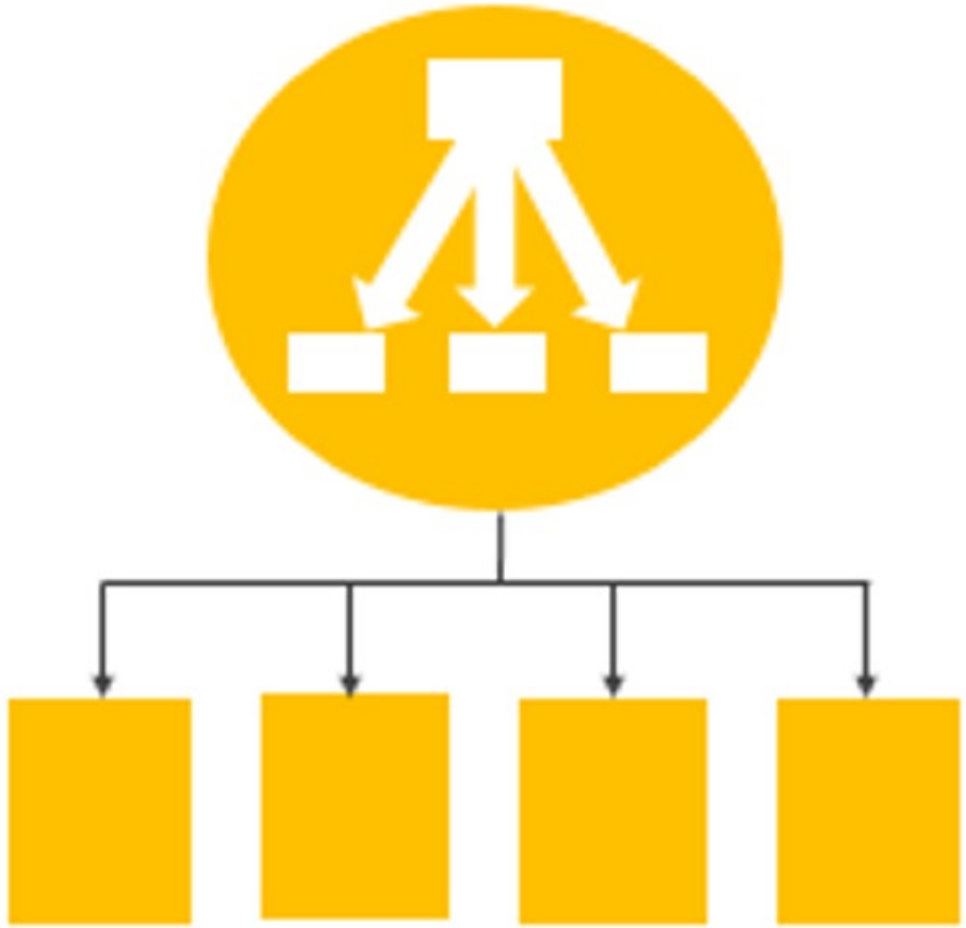
Application Load Balancer

Traffic balanced across targets



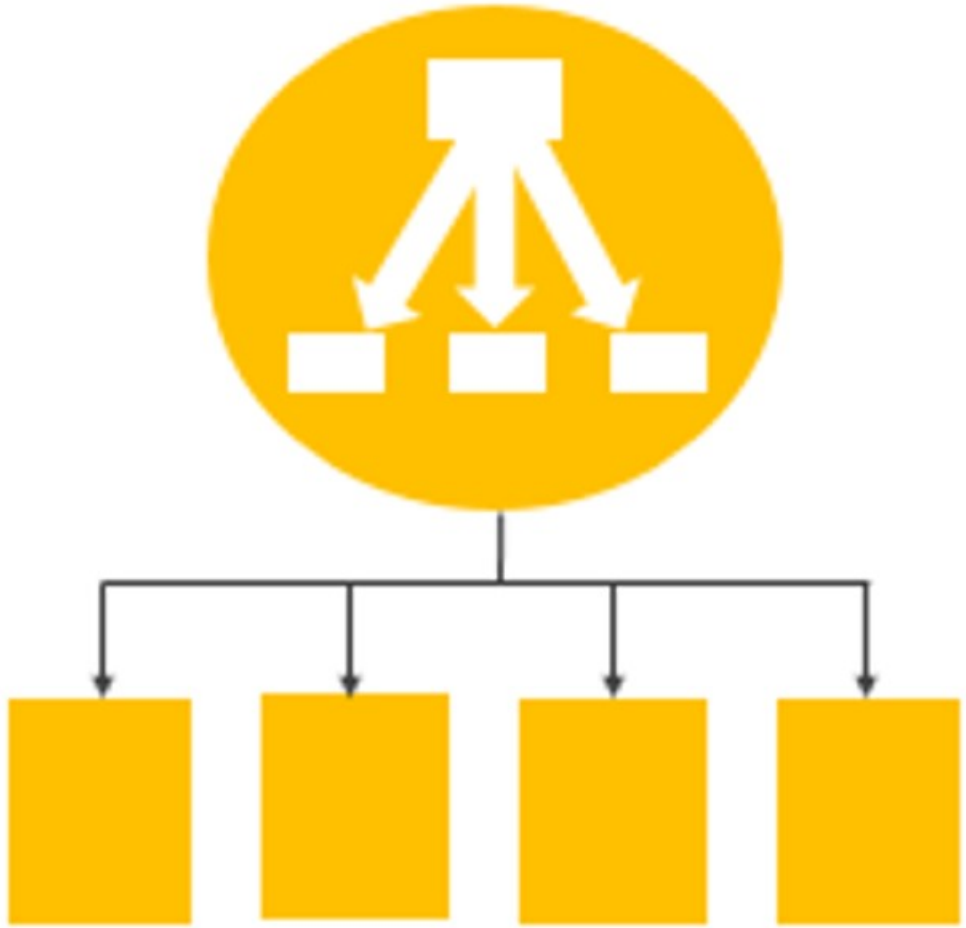
Public facing ELB

- Deployed in a public subnet providing front-end protection
  - Web servers or application servers, hosted in a private subnet
  - Configured to accept traffic only from your load balancer
  - Up to 20 load balancers can be created per region for different clients needs (desktop, mobile)



ELB Design

- The incoming (ingress) node of an Internet facing load balancer has a public EIP address provided automatically (Static public IP)
- AWS DNS (Route 53) resolves the load balancer's domain name
- Request are sent from LB to a healthy registered instance on its private IP address
- DNS round-robin routing is used for TCP listeners
- NLB preserves the client-side source IP: the back-end instances can see the clients IP address

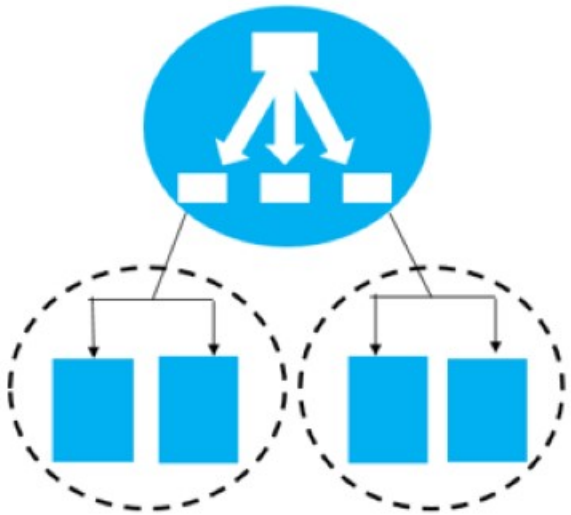


ELB Operation

- Health Checks
  - Unhealthy instances stop receiving traffic; network and application health checks (URL)
- Security
  - Managed by Security Groups and NACLs in hosted VPC
- Fail-over
  - If no healthy targets in AZ, fail-over to another AZ
- Connection draining – deregistration
- Cross-zone load balancing supported
- Integration
  - Auto Scaling, CloudWatch

Application Load Balancer (FYI)

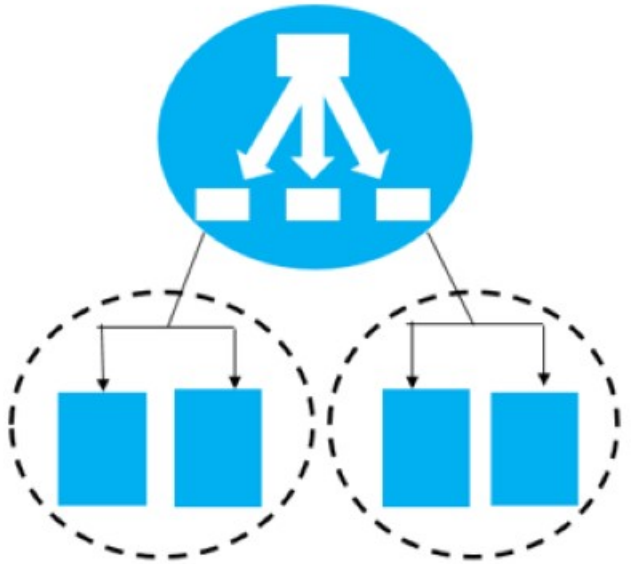
- Load balancing up to Layer7
- HTTPS support
- Performs SSL Offloading
- Authentication (Cognito)
- Sticky session support
- Access logging available
- Web Application Firewall support
- Delete protection

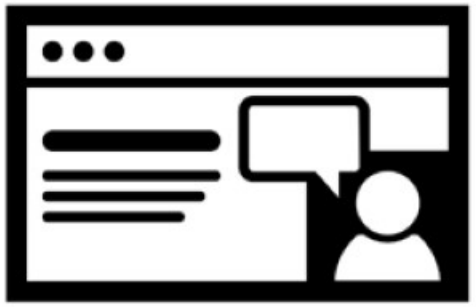


Application Load Balancer Operation

Route requests to target groups (Instances / Containers)

- Traffic is sent to any healthy target in the “target group”
- Supports Web Sockets and HTTP/2
- Routing decisions can be based on content or path of application traffic
- Path based routing: listeners forwarded based on the URL (HTTP/HTTPS)
- Supports routing requests to containers or micro-services

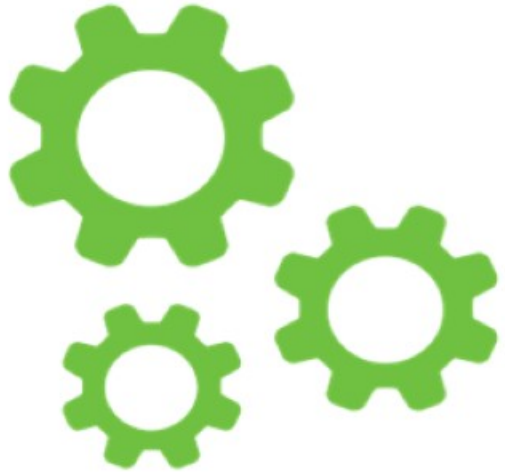




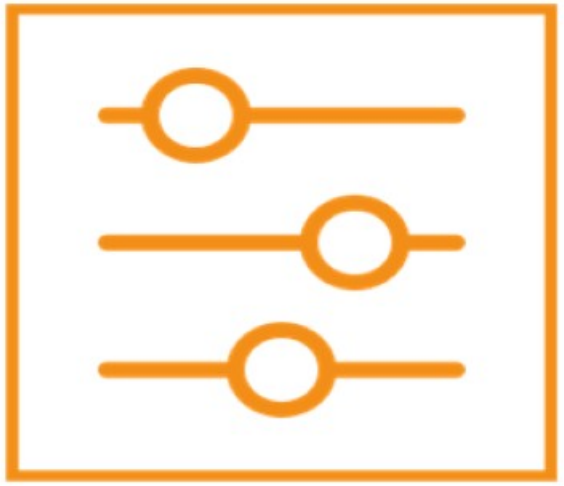
Exercise: ALB Setup

Auto Scaling

Auto Scaling Concepts



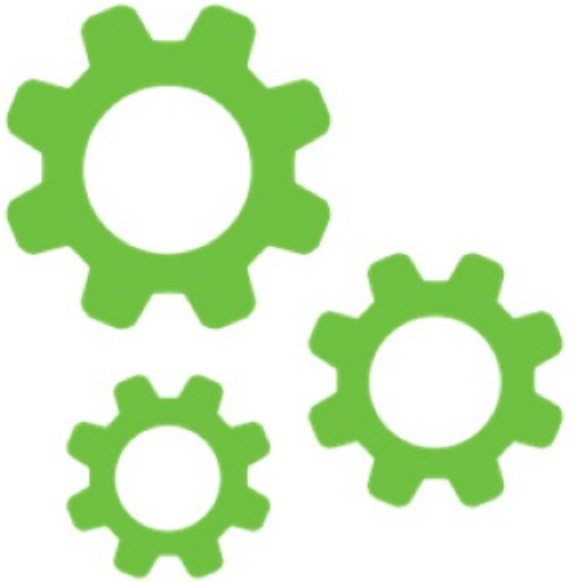
Launch Configuration



Auto Scale Group

Policy

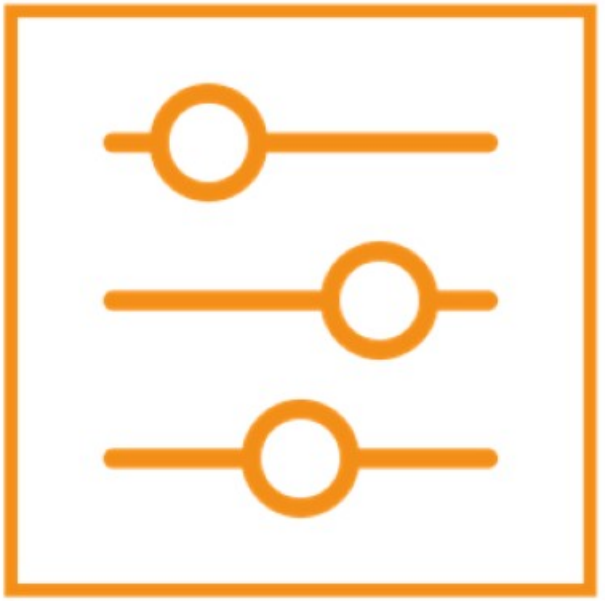
Launch Configuration



- Instance type
- Amazon machine image (AMI)
  - (Golden Image)
- Security groups
  - Firewall protection
- SSH keys
  - Authentication
- Bootstrapping (User data)
  - Automation

Auto Scale Group

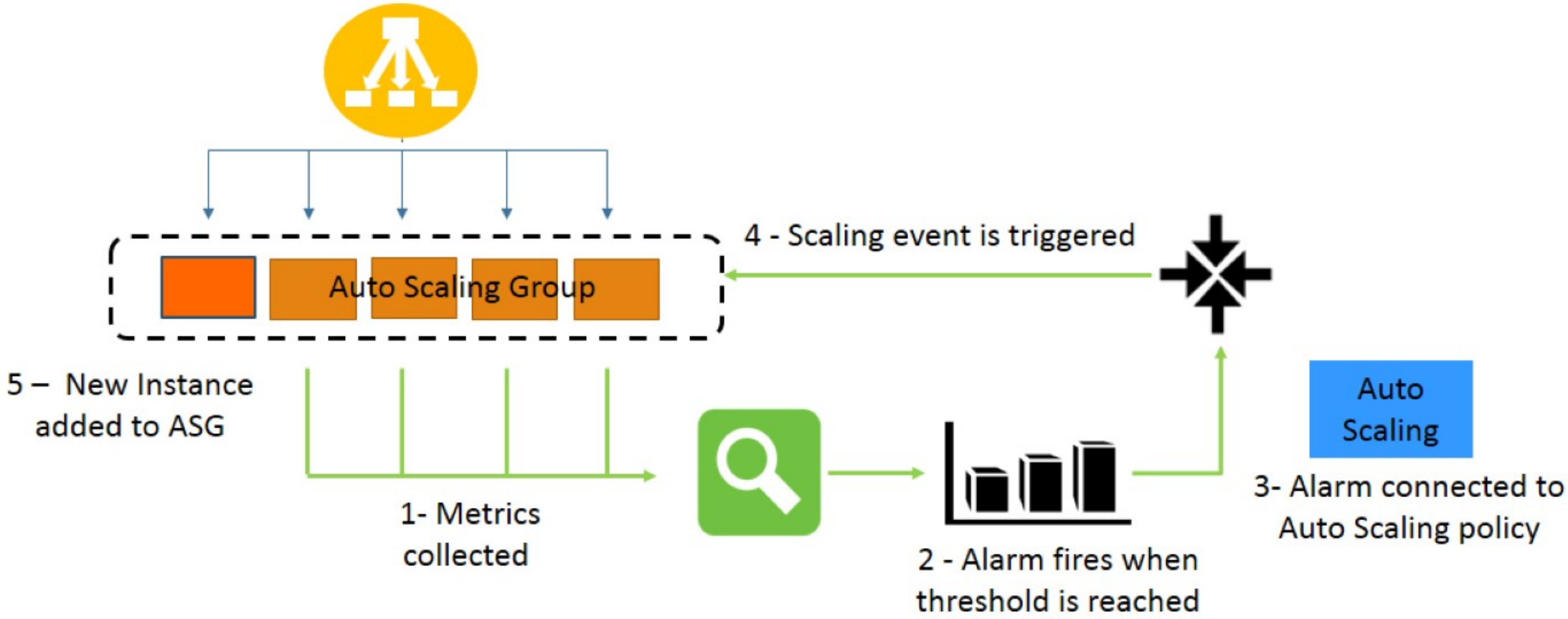
- Desired capacity (to maintain)
- Minimum number of instances
- Maximum number of instances
- Keep capacity balanced across Availability Zones
- Example:
  - Desired number of instances: Example: 2 per AZ
  - Minimum = 2
  - Maximum = 10

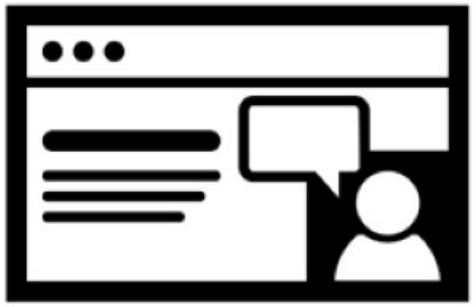


Auto Scale Policy

- When will auto scaling scale out or in?
  - Launch instances when application is under increased load
  - Terminate instances when application is not under load
- Scheduled: Scale out / in at specific times
- Dynamic scaling: scale using CloudWatch metrics
  - Add / remove fixed number of instances +2
  - Add / remove percentage of existing capacity + 20%
- Step Scaling Policies – multiple policies within the same policy
- CloudWatch metrics: (CPU usage, Incoming network traffic)

Auto Scaling with Alarms





Exercise: Auto Scaling

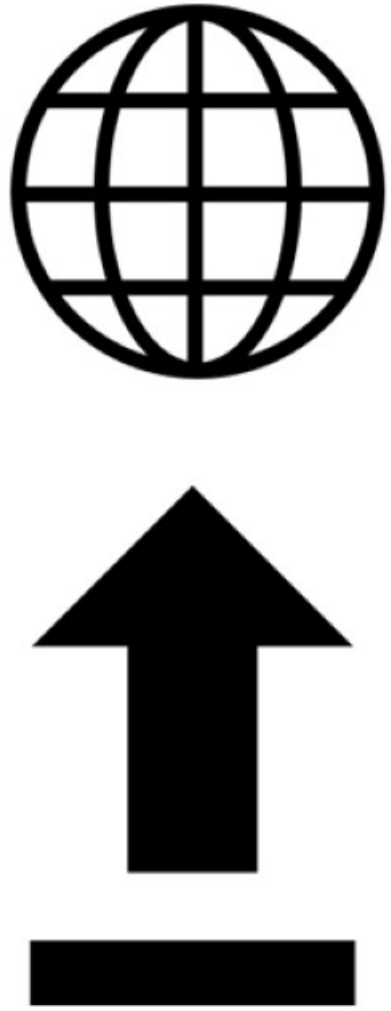
VPN Connections

VPN Connections

- Connect your VPC to external networks using a VPN connection
- AWS Hardware VPN – the hardware devices is on the customer side of the VPN connection
- AWS Direct Connect – Dedicated private fiber connection; hardware VPN connection on the customer side of the VPN. 1 - 10 GIG
- AWS VPN Cloud Hub – useful for multiple branch offices; hardware VPN connection on the customer side of the VPN
- Software VPN – EC2 instance in your VPC running a software VPN appliance

External Connectivity

- The AWS side of the VPN tunnel is the Virtual Private Gateway (VPG)
- The customer gateway (CGW) is either a hardware or software application on the customers side of the VPN tunnel
- The VPN tunnel is initiated from the CGW to the VPC
- VPGs support static and dynamic routing (BGP)
- The VPN connection on AWS's side of the tunnel is created with two tunnels providing high-availability to the VPC on the AWS location



Internet Gateway (IGW)

Allows Internet communication from instances hosted in a VPC on a public subnet

IGW Creation Steps:

1. Attach a IGW to your VPC
2. Create an additional subnet route to direct Internet traffic to the IGW

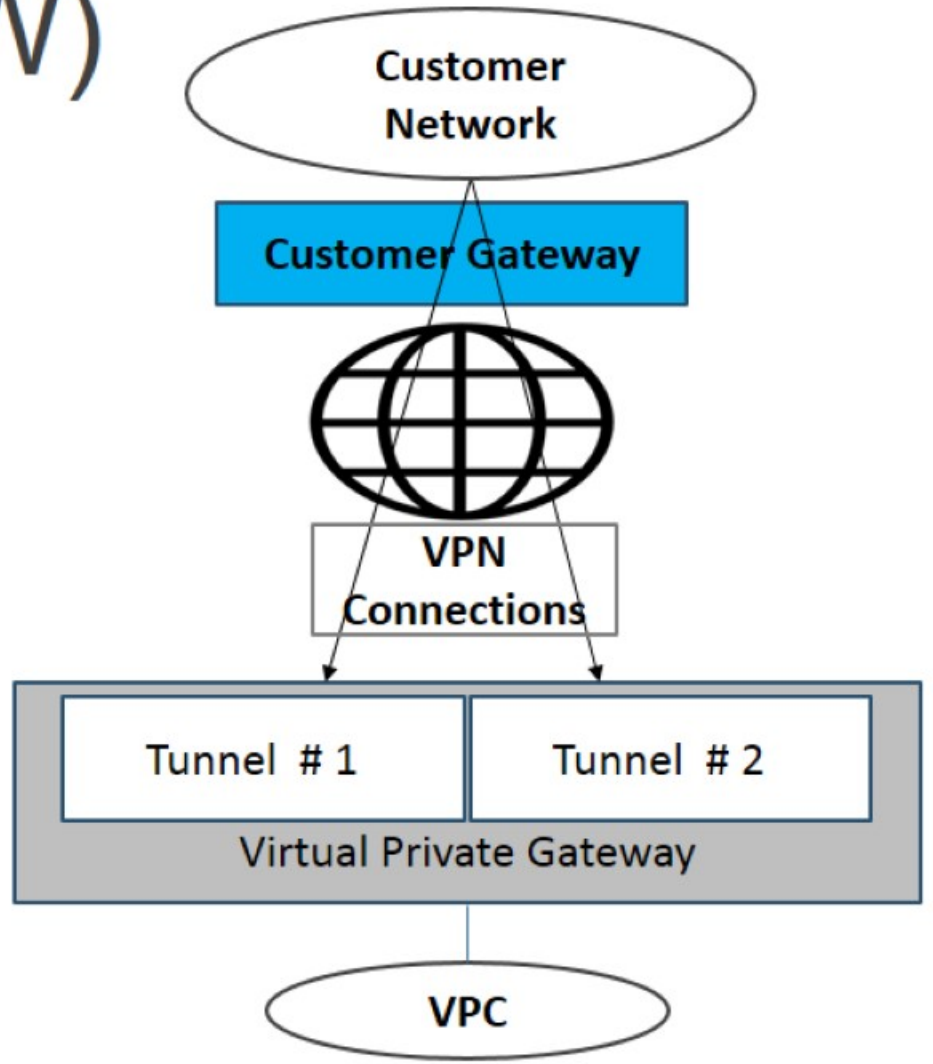
Virtual Private Gateways (VPG)

The VPG allows interconnection to a VPC from a private data center using IPSec tunneling

- VPG is the virtual private network concentrator on the Amazon side of the VPN connection between the two networks
- Data centers can be connected to a VPC with either hardware or software VPN connections

Customer Gateways (CGW)

- A customer gateway is either a physical or software appliance on the customers side of the VPN connection.
- A VPN tunnel is established between a virtual private gateway and a customer gateway
- The connection will be dynamic (BGP) or static in design
- On the AWS side, each VPN connection is created using two tunnels providing high availability to the VPC

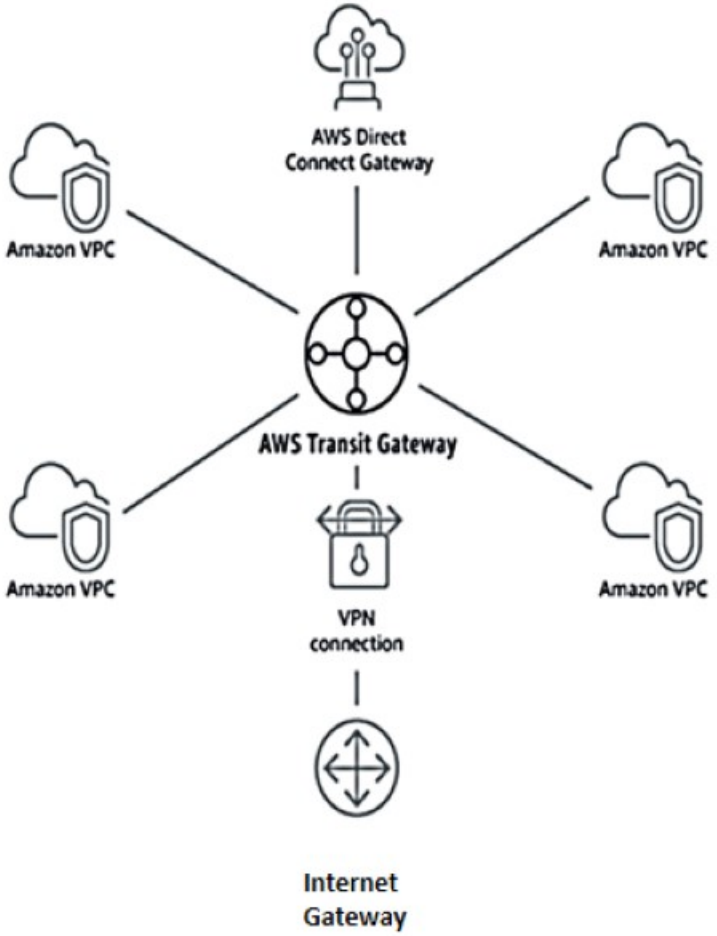


Direct Connect

Private connectivity between AWS and your Datacenter

- Industry-standard 802.1 Q VLAN
- Can be partitioned into multiple virtual interfaces
- One connection can be used for public and private access to resources

- Direct connect speeds from 1 to 10 Gb per connection
- For additional capacity add additional connections
- Can replace standard VPN connection across the Internet



Transit Gateway

Allows you to connect a single gateway device that routes communicates to the networks that are connected to the transit gateway using a hub and spoke model.

Any VPC connected to the transit gateway is automatically routed to the connected VPC, direct connect gateway and customer gateway routes.

Each TG can connect to 5000 VPCs

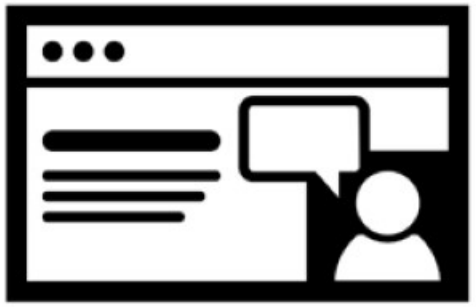
Network traffic can burst up to 50 Gbps

CloudFront

CloudFront (CDN)

Content delivery network providing global delivery of requested content

- Global network of 100 Edge locations, 23 countries and 49 cities
  - Integrated with Route, 53, S3, EC2, ELB and custom origin servers as back-end origin resources
  - Supports dynamic content, on-demand and live streaming (RMTP)
- Direct user requests based on the device and country (geo-targeting)
- Integrated with Amazon Elastic Transcoder and Lambda Edge, AWS Web Application Firewall and AWS Shield
- Securely serve private content using signed URLs, signed cookies, or Origin Access Identities (OAI)



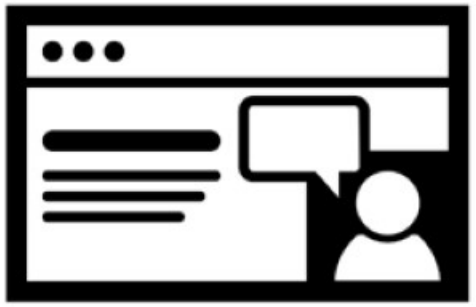
Exercise: CloudFront

CloudWatch

CloudWatch

Built-in monitoring service for AWS cloud resources

- Collect and track metrics
- Supports EC2, Dynamo DB, RDS, ELB, SQS, SNS, and more
  - Auto Scaling Groups – 7 metrics
  - ELB – 13 metrics
  - Route 53 – health checks
- Monitor your system and application log files
- Set alarms on errors found in log files and react to changes
- Set billing alarms and be alerted by CloudWatch
- Alarms have three states: OK, ALARM, INSUFFICIENT DATA



Exercise: CloudWatch

AWS Config

AWS Config

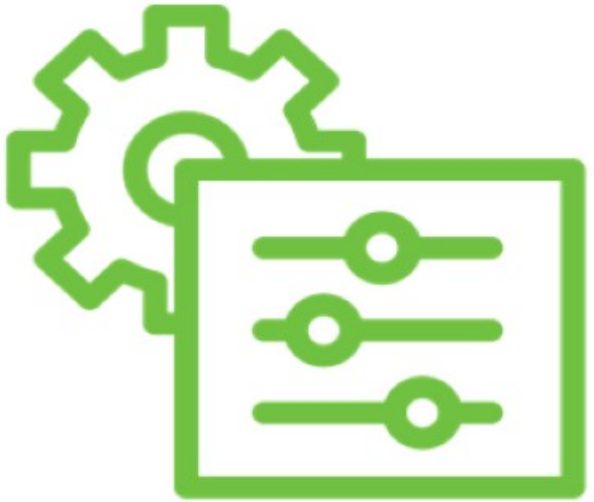
Records changes and current status to resource inventory

- Notify when resources are created, modified, or deleted
- Configuration items track resource attributes and the relationships, current configuration, and related events
- Custom Config rules for remediation
- Config can also capture:
  - Software inventory on EC2 instances
  - Patch levels
  - Application versions

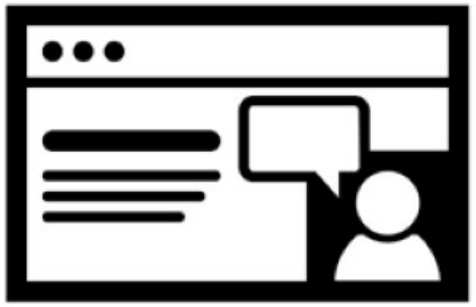
Config Use Cases

Analyze security

Auditing for compliance



Change management



Exercise: AWS Config

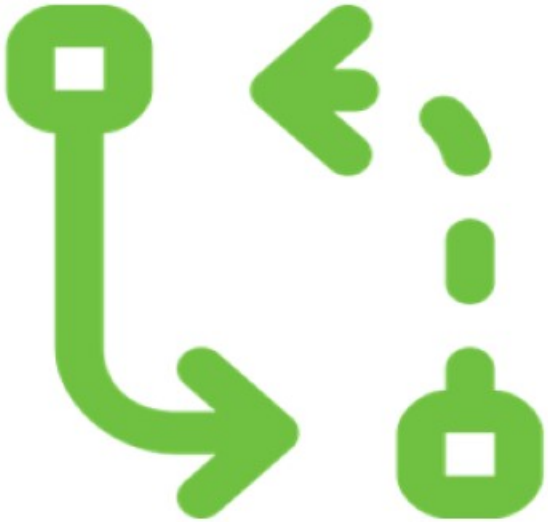
CloudTrail

AWS CloudTrail

Record of API requests and responses

- Who did what and when from where
  - Recorded info includes:
    - Identity of the API request
    - Time of the API call
    - Source IP address of the API
    - The response returned by the AWS service
- Monitor your systems, applications and custom log files
  - Logs saved in S3 bucket
  - Receive notifications using Simple Notification Service

CloudTrail Use Cases



Track changes to resources

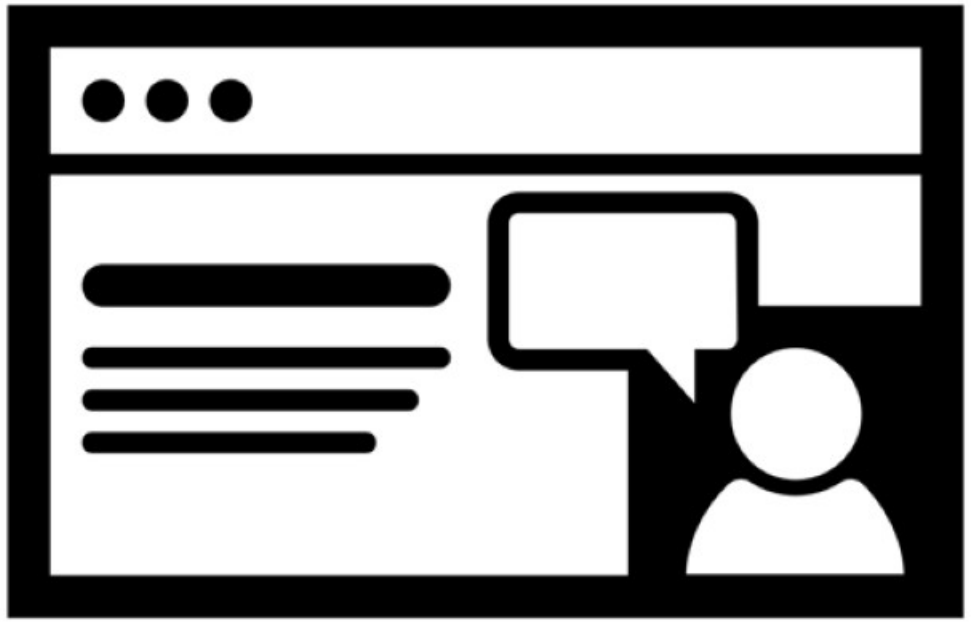
VPC security groups, NACLs

Compliance

API call history

API Activity History

Lookup activity over the last 90 days



Exercise: CloudTrail

Trusted Advisor

Trusted Advisor

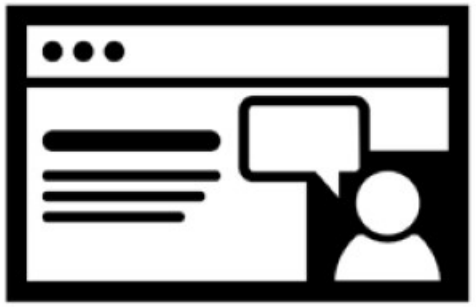
Analyzes your AWS environment against "Best Practice" checks

- Provided for the following categories:
  - Cost optimization
  - Improve security
  - Review fault tolerance
  - Optimize Performance

No Problem

Investigate

Take Action



Exercise: Trusted Advisor

DevOps Tools

Kinesis

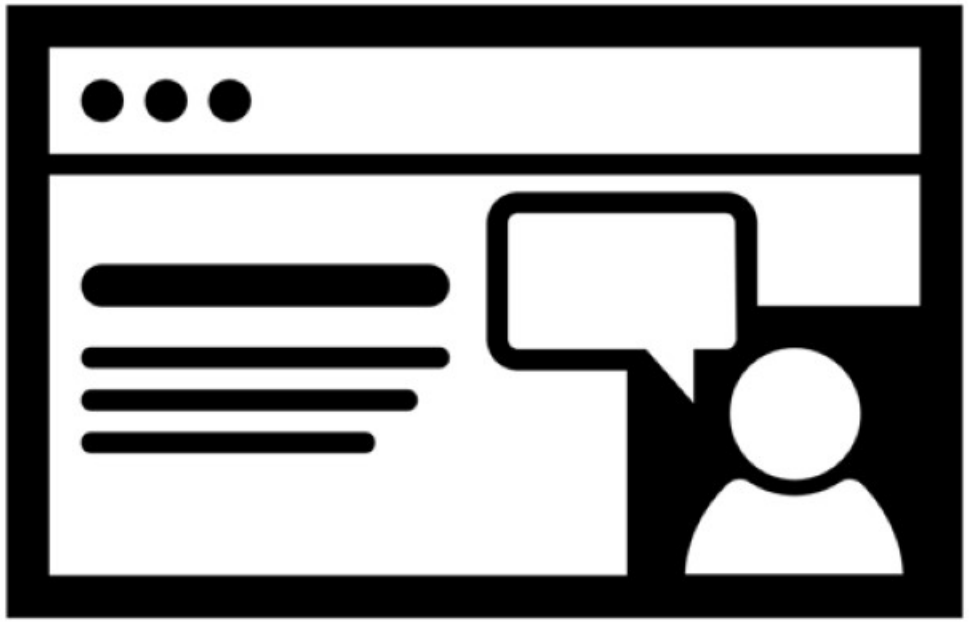
Development platform for handling massive streaming data

- Load and analyze streaming data
- Kinesis Firehose – Receive massive volumes of streaming data stored in S3, Redshift or ElastiSearch
- Kinesis Streams – Complex analysis of streaming data in real-time
- Data can be stored in S3, DynamoDB, Redshift, or EMR

CloudFormation

JSON template for automating resource stack creation

- Provision, update, and delete resources using CloudFormation
- Infrastructure as code
- Two main components:
  - Templates – JSON, or YAML text file describing the resources to be built
  - Stack – cloud formation uses the template file to build resources in your AWS account. The running resources are called a stack



Exercise: CloudFormation

Elastic Beanstalk

Automated deployment of web applications and containers

- Upload application
- Configure ELB, instances, and auto scale automatically

Simple Notification Service (SNS)

Fully managed push notification service

Publish messages from service or application

Push messages to devices and services via API

Delivered once, or many times to subscribers

Multiple messages to multiple recipients

Use Case: Monitoring , mobile applications

Simple Queue Service (SQS)

- Fully managed message queuing service
- Integrates with Amazon S3 Bucket (Logs or Storage)
- Content changes trigger message to queue
- Automated response from queue subscribers
- Dead letter queues: No lost orders
- Visibility timeouts: I'm working on it

Elastic Transcoder

- Media transcoding in the AWS cloud
- Transcode / Convert media files into playback formats supported on devices, phones, and PC's
- Presets provided for popular output formats

Create transcoding job:

1. Source Files
2. Transcoding rules
3. Destination

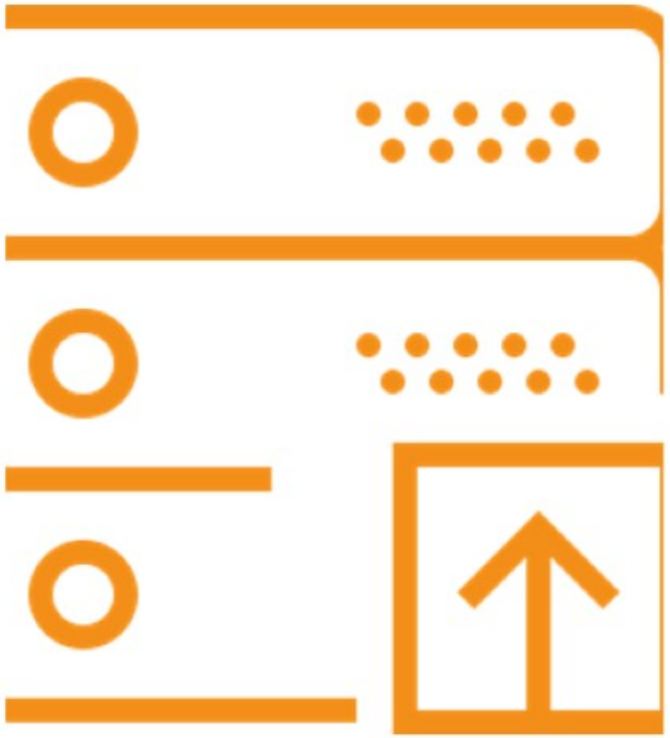
Lambda

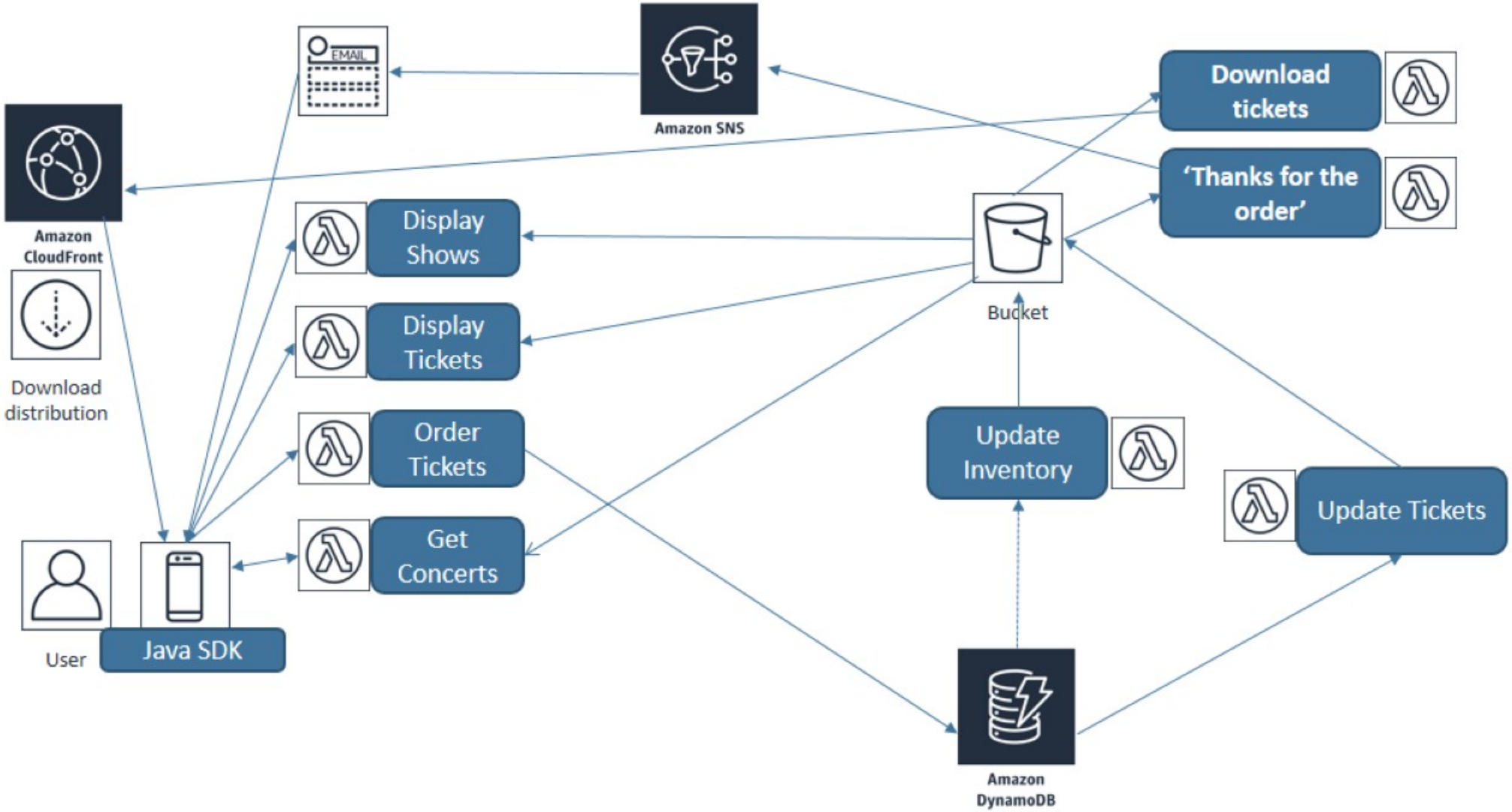
Lambda

Run code without provisioning or managing servers

- Upload your code, define resources, define processing time
- Lambda code is triggered from other AWS services, or Web / Mobile app
  - Code executes in response to triggers
  - Charges apply for every 100 ms your code executes, and number of executions

- Use S3 to trigger a Lambda function after data is uploaded into bucket
- When data is added to a Dynamo DB table execute a lambda function to validate the data
- Use an IoT Dash button




EMAIL
Amazon SNS
Download tickets
'Thanks for the order'
Amazon CloudFront
Display Shows
Display Tickets
Order Tickets
Get Concerts
Bucket
Download distribution
Update Inventory
Update Tickets
User
Java SDK
Amazon DynamoDB

DynamoDB

DynamoDB

Optimized for compute

Scale horizontally

Built for OLTP at scale

Fully-managed NoSQL

Document or key-value

Fast and consistent

DynamoDB FYI

- Document data model supports using JSON on documents stored in Dynamo DB tables
- Dynamo DB is designed with automatic synchronous data replication across three facilities in a region
- Dynamo DB supports cross region replication across regions with Global tables
- Dynamo DB integrates with Lambda using triggers
  - Triggers execute a custom function when item levels change in a table
- There is a downloadable version of Dynamo DB that can be used to test your applications locally

Dynamo DB Structure

The data model consists of tables, items, and attributes

- Each table is a collection of items (Auto parts)
- An item could be: muffler
- Attributes could be: stainless steel, Mazda, M5
- Attributes can be single or multivalued

Dynamo DB Primary Key

The primary key of each table must be defined; this provides unique identification for each item in the table

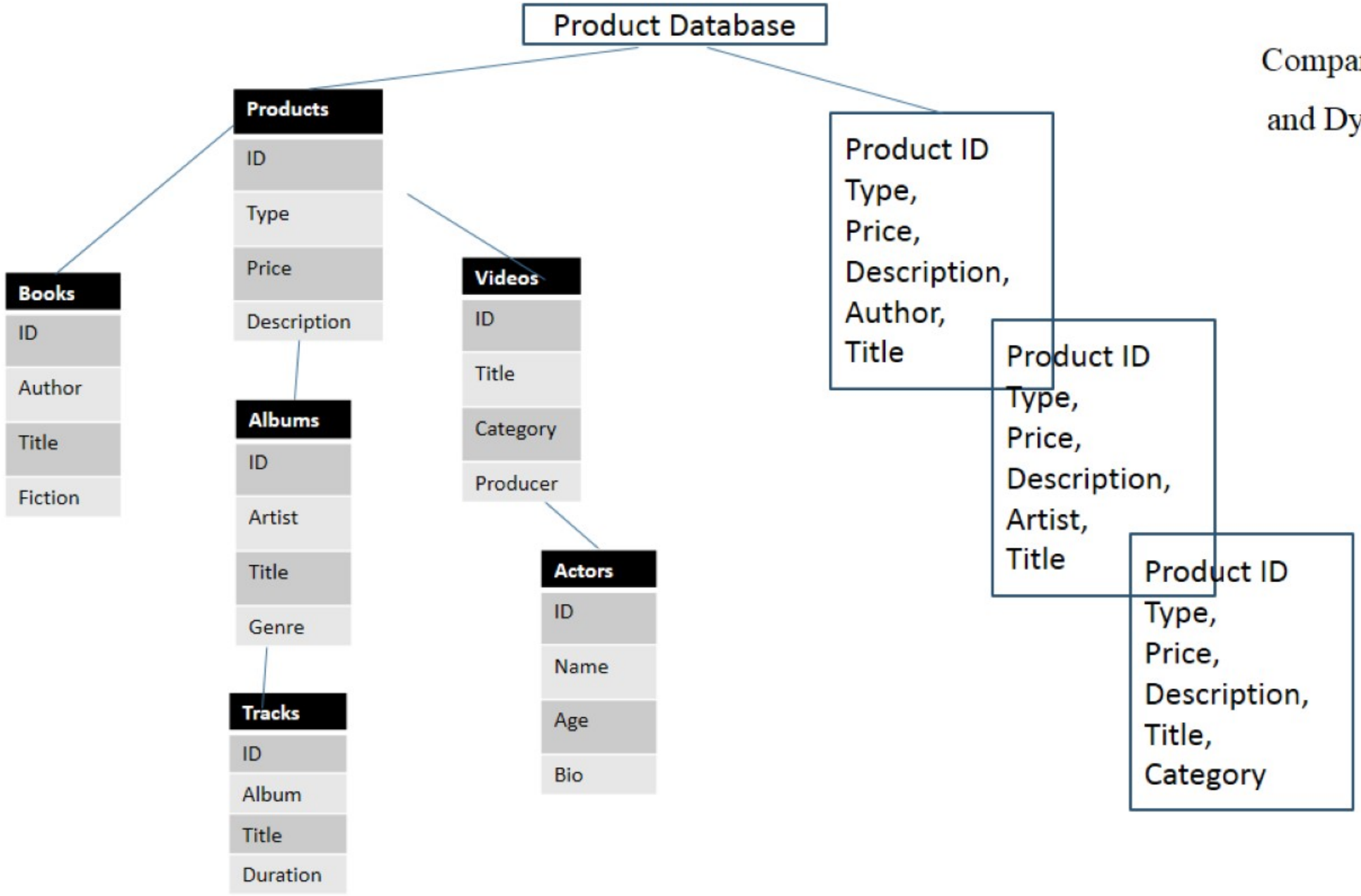
1. **Partition key consisting of one attribute**
   - Example: Table: People
   - Partition key: Personal ID
   - Each item in the table must have this attribute to help uniquely identify each item in the table from all other items
2. **Partition key and sort key (Two attributes)**
   - Example: Table: Music
   - Primary key: Artist
   - Sort key: Song title
   - Each item in the table must have these two attributes to help uniquely identify each item in the table from all other items

Secondary Indexes

Secondary indexes allows you to query tables using an alternate key

- There are two types of secondary indexes
  - Global secondary index – a partition and sort key that is different from the current primary key defined on the table
  - Local secondary index – an index that has uses the partition key currently defined on the table with a different sort key
- Up to five Global secondary indexes and local secondary indexes can be defined per table
- Example : Global secondary indexes – Genre and CD Title

Comparing SQL and DynamoDB



Read/write capacity mode

Select on-demand if you want to pay only for the read and writes you perform, with no capacity planning required. Select provisioned t
if you can reliably estimate your application's throughput requirements. See the DynamoDB pricing page and DynamoDB Developer (

Read/write capacity mode can be changed later.

- (•) Provisioned (free-tier eligible)
- ( ) On-demand

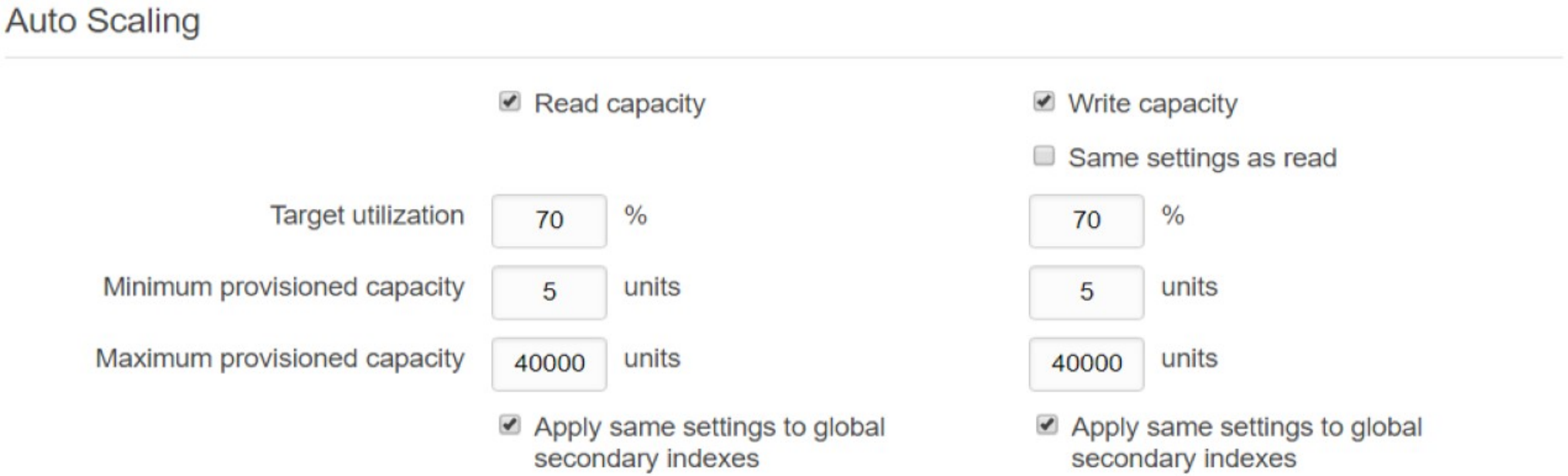
Last change to on-demand mode: No read/write capacity mode changes have been made.

Next available change to on-demand mode: You can update to on-demand mode at any time.

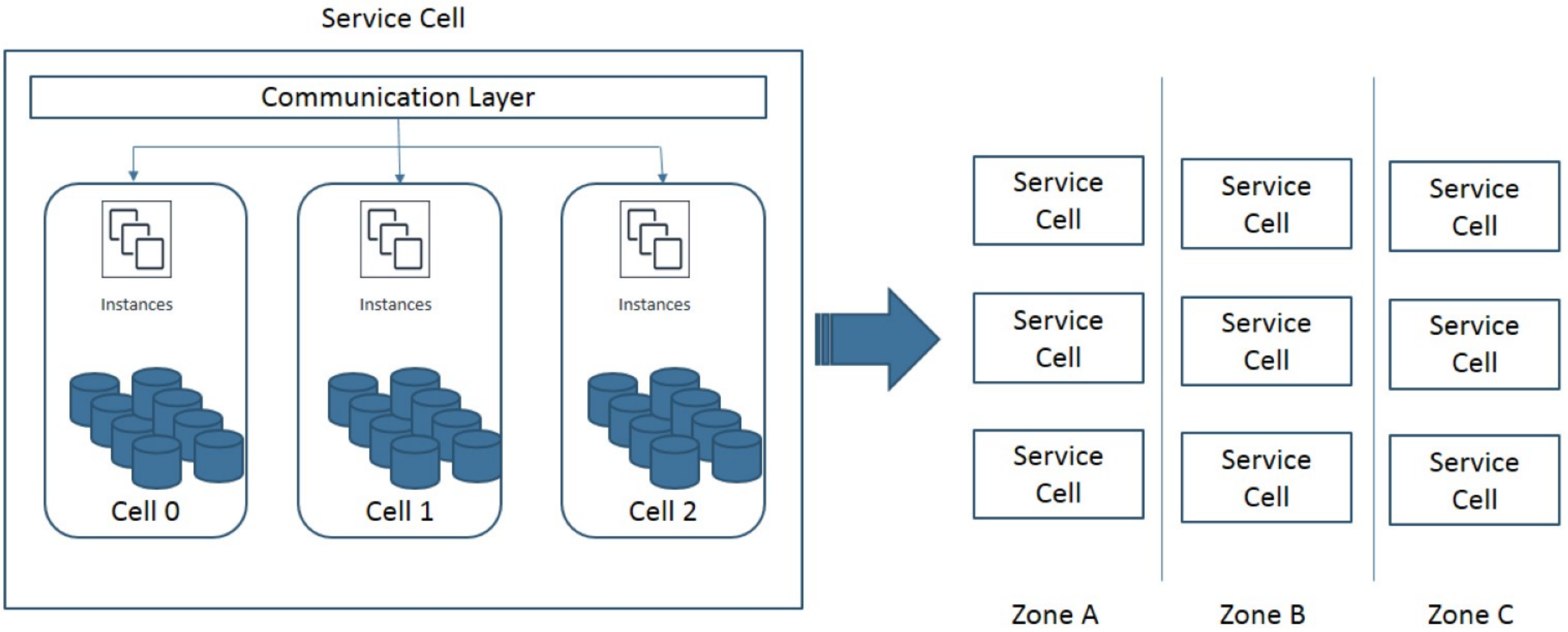
Provisioned capacity

	Read capacity units	Write capacity units
Table	5	5
Estimated cost	$2.91 / month (Capacity calculator)	

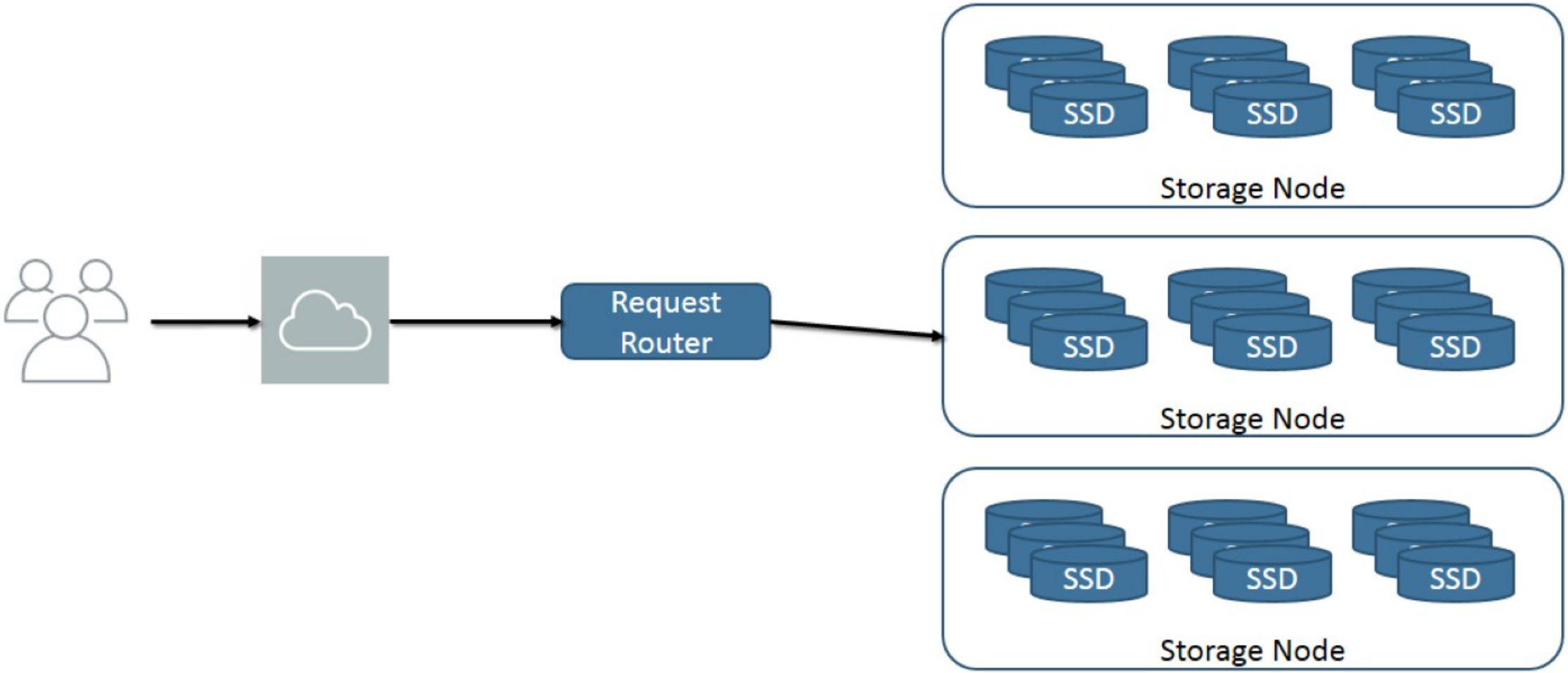
Adjusting Table Capacity



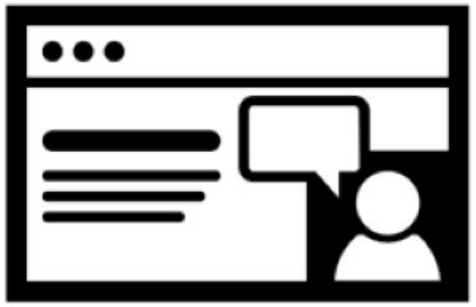
DynamoDB Auto Scale Settings



DynamoDB Cell-based Design



DynamoDB Storage Node Design



Exercise: DynamoDB

Aurora

Aurora FYI

Relational database engine with 5 times the performance of MySQL

- Fully managed 6-way replication across 3 availability zones
- MySQL compatible, PostgreSQL compatible
- Database engine integrated with SSD Virtual SAN
- Minimal database storage 10 GB; can scale to 64 TB in 10 GB chunks
- Computer resources can be scaled to 32 virtual CPUs and 244 GB Memory
- Automated backups are enabled by default; Manual snapshots allowed
- Up to 15 read replicas with asynchronous replication with optional cross-read region replicas and cross-region failover support

Configuration

Estimate your monthly costs for the DB Instance using the AWS Simple Monthly Calculator

DB engine

Aurora - compatible with MySQL 5.6.10a

Capacity type Info

- (•) Provisioned
  You provision and manage the server instance sizes.
- ( ) Provisioned with Aurora parallel query enabled Info
  You provision and manage the server instance sizes, and Aurora improves the performance of analytic queries by pushing processing down to the Aurora storage layer (currently available for Aurora MySQL 5.6)
- ( ) Serverless Info
  You specify the minimum and maximum of resources for a DB cluster. Aurora scales the capacity based on database load.

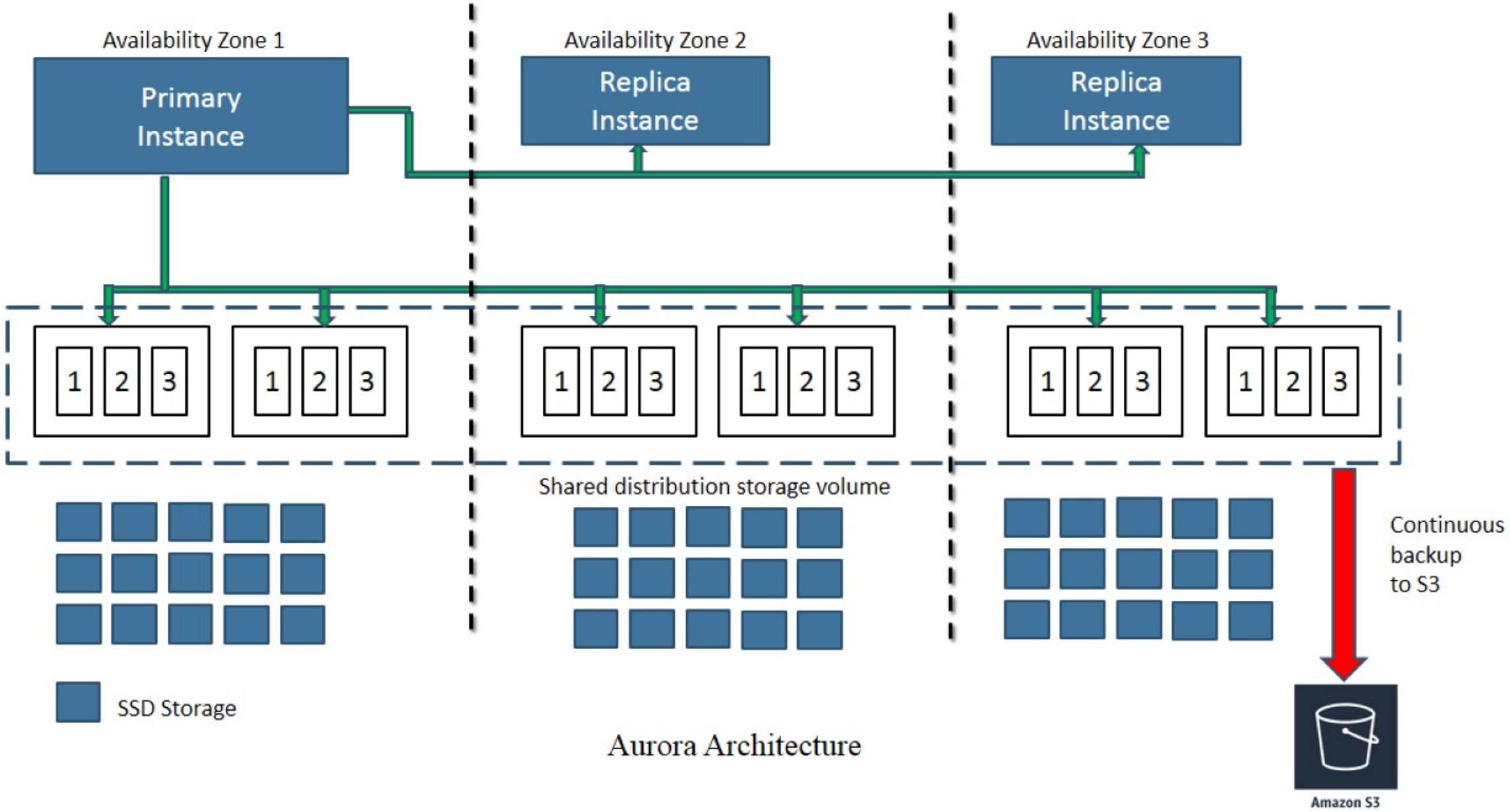
DB instance class Info

db.r4.xlarge — 4 vCPU, 30.5 GiB RAM

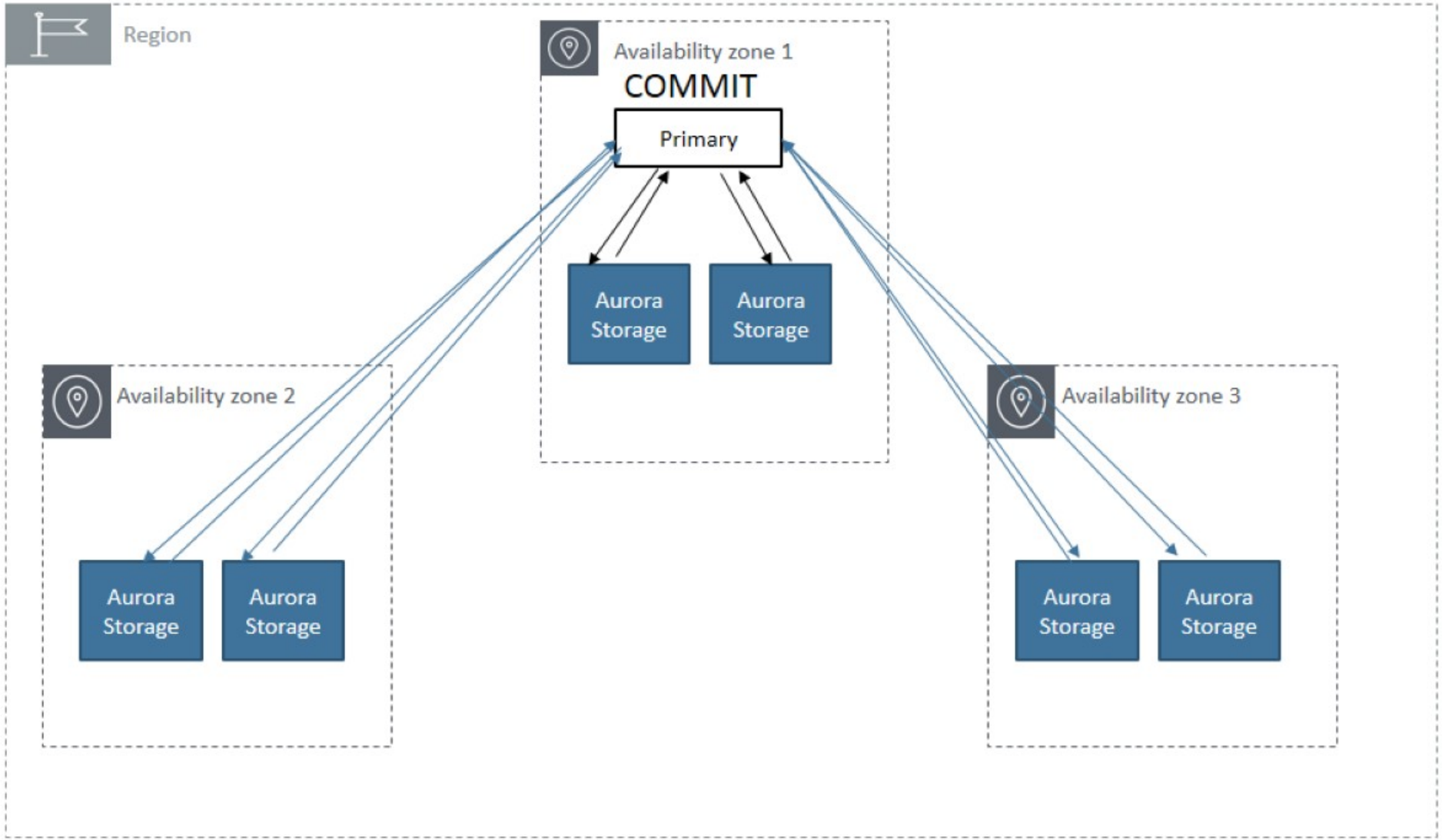
Multi-AZ deployment Info

- (•) Create Replica in Different Zone
- ( ) No

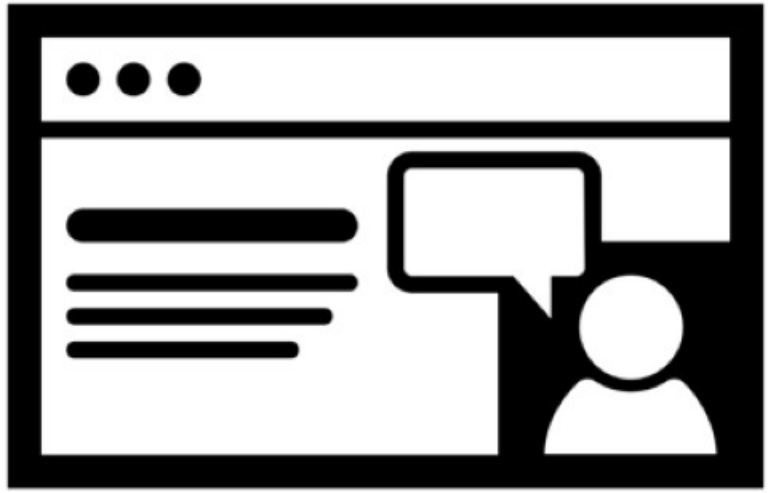
Aurora Deployment Options



Aurora Architecture



Aurora Cluster Design



Exercise: Aurora

What we covered

- Identity and Access Management Administration
- Deployment and Management of Relational Database Services, DynamoDB, and Aurora
- AWS Management Tools and what they are designed to do
- ELB load-balancing and Auto Scaling
- CoudWatch, CloudTrail, AWS Config
- Trusted Advisor
- DevOp Tools

Q and A / Wrap-up