**AWS Structure**

AWS provides cloud services.

**AWS Structure:**

1. Account and Service Layer
2. Physical and Network Layer

**ACCOUNT AND SERVICE LAYER:**

This says about how we interact with AWS.

We can interact with AWS using three of below option

1. API (AWS Console)
2. Command Line (AWS CLI)
3. Custom Software Application (AWS SDK)
4. **API (AWS Console)**
   1. Through API we will launch the service we need (VM – EC2 instances, DB service, put something in s3 etc)
   2. API call will be successful only if we have correct permissions
   3. Every user must be granted permission to use a particular api and perform a certain operation

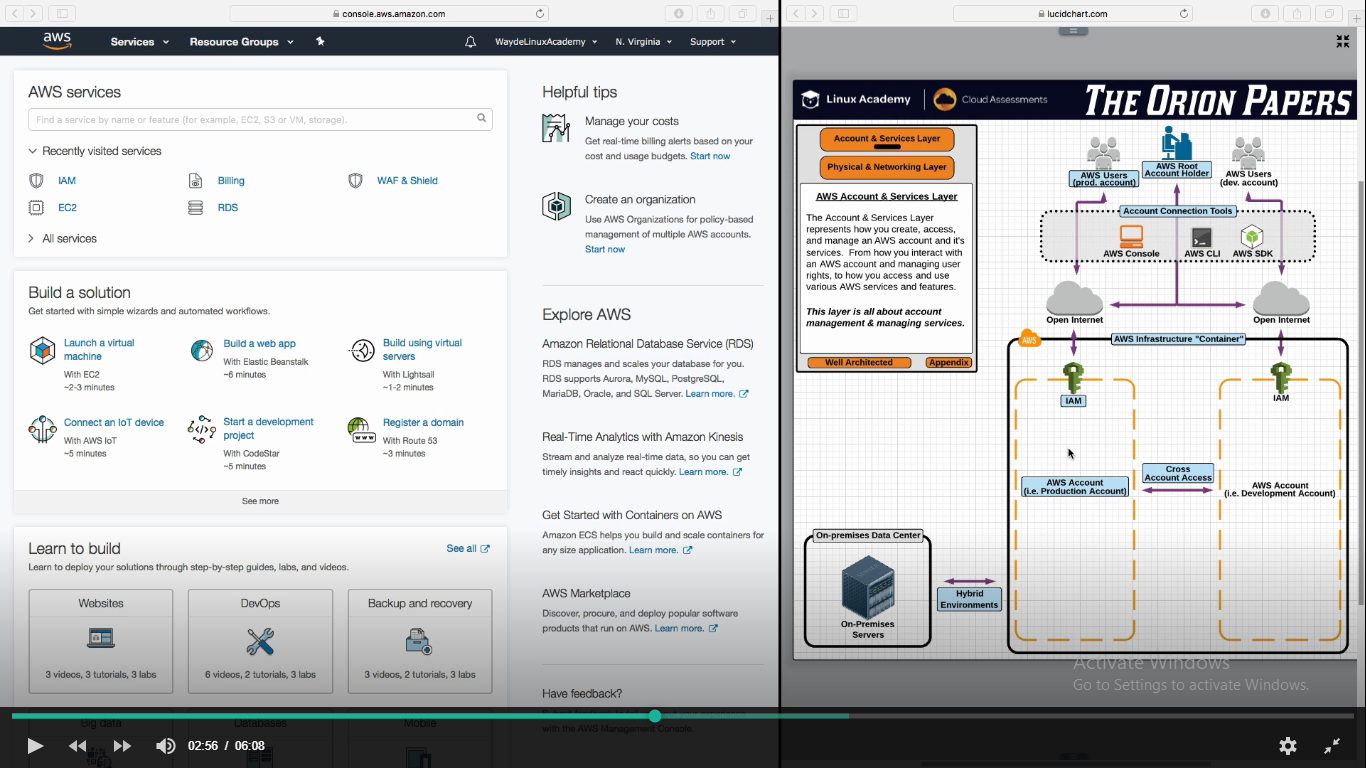
**API** – Application Programming Interface

For a first time we use to login with an email address. Logging with email address means that user is in root account. (That user will have full access to all AWS Services)

Using IAM (Identity and Access Manager) we can create an account which allows user to do some actions under privilege activities

Users are going to be interfacing with AWS over the internet. Yet it is possible to use a private connection known as “**Direct Connect**” which allows a private connection to service API Endoints.

**AWS Account and Service Layer**



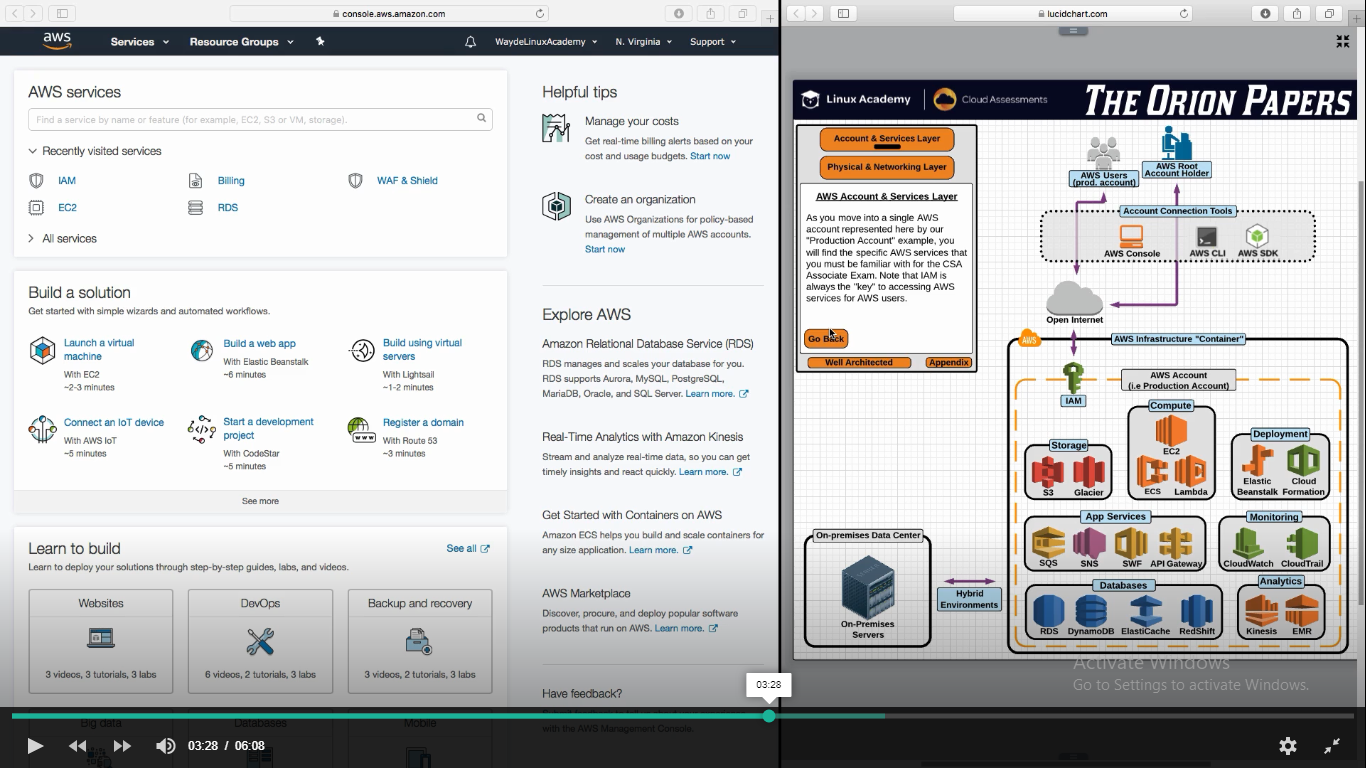
Every user who need to invoke API and perform an operation must be provided access.

Initially when we create an account in AWS we created with email address and it have full privilege to all its services. This user is the root user.

We have to use IAM (Identity and Access Manager) to restrict user and decide what operation/service that user is going to do. An organization will have multiple AWS accounts. As shown in above picture “Production Account” shall not be give access for all users.

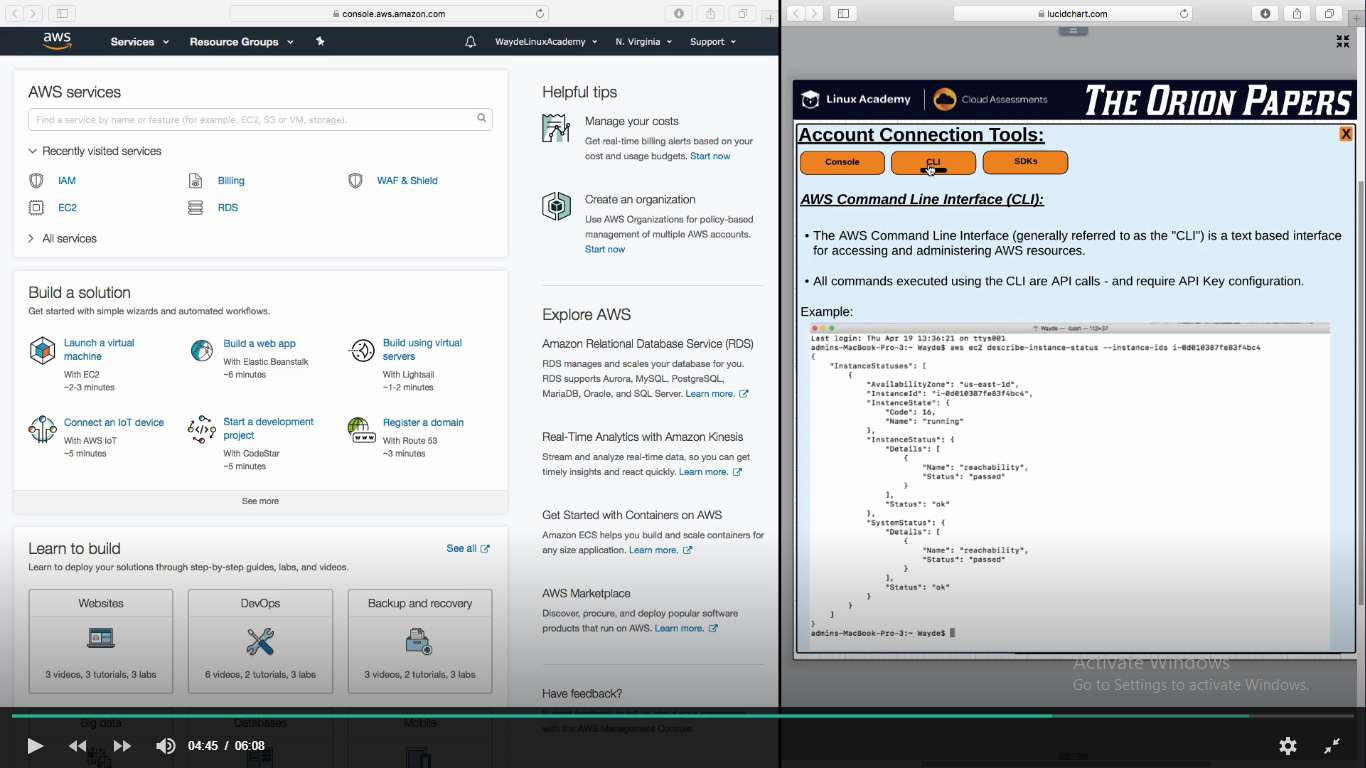
**Cross Account Access** is also possible – Users in one account to access resource in another account.

PROD Account and its related accesses

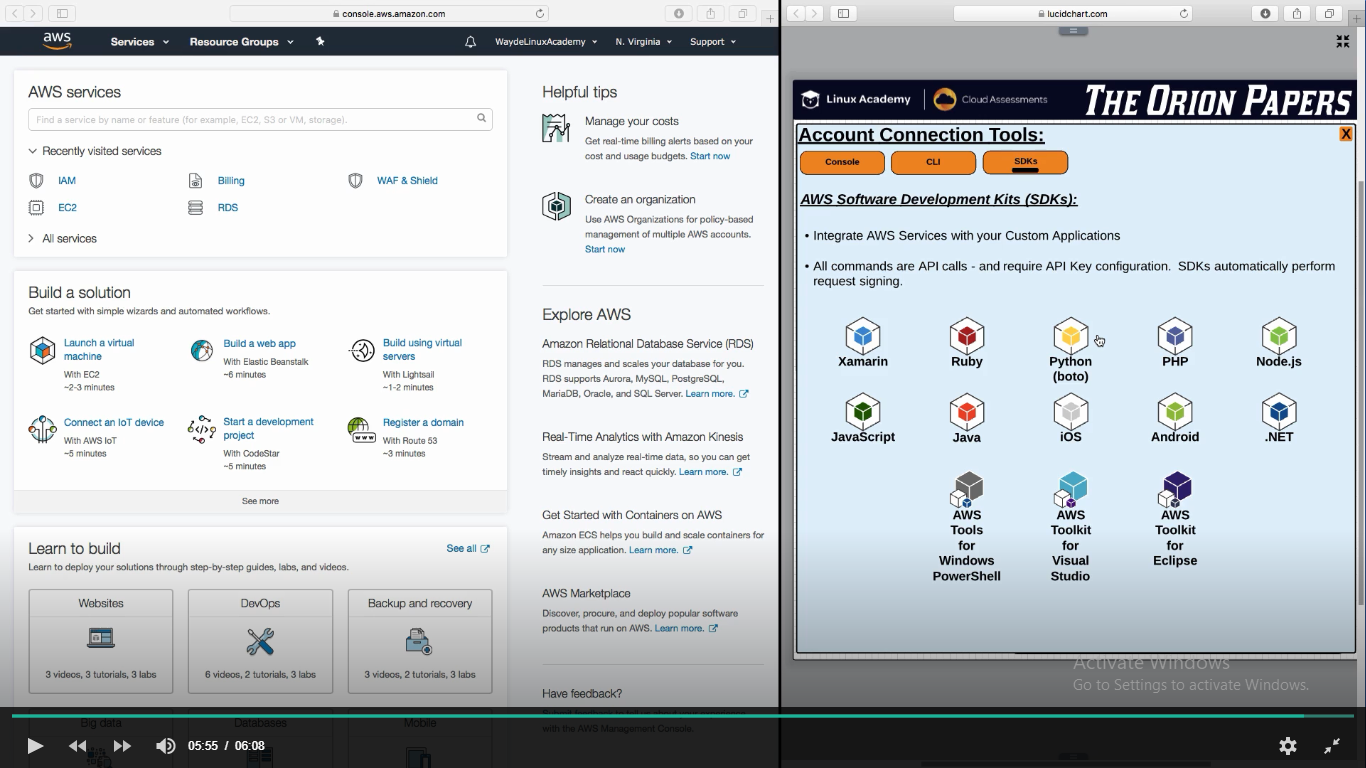


1. **Command Line (AWS CLI)**
   1. Have to install in each user’s desktop
   2. Operators use it for creating automated scripts
   3. Eg. To run a backup job on a schedule

Eg: To run a script to get the status of an instance as show below



1. **Custom software application (AWS SDK)**
   1. Used by Developers to develop an application which interface with AWS
   2. Below means make API call to connect with AWS Console
   3. It can able to make calls to API only for which it have access



**Physical and Network Layer:**

1. AWS Regions
   1. Region will have two or more availability zones
   2. This makes high availability possible for the application
   3. Each availability zones are separated by 10+ miles
   4. Actual physical data centers are grouped in this availability zones
   5. Have 18+ active regions across the world
   6. All service (EC2, S3, etc) will not be available in all regions
   7. Prices vary with region
      1. High/low tax in that geographic location
      2. High/low cost of labours
2. AWS Edge Location
3. Regional Edge Cache

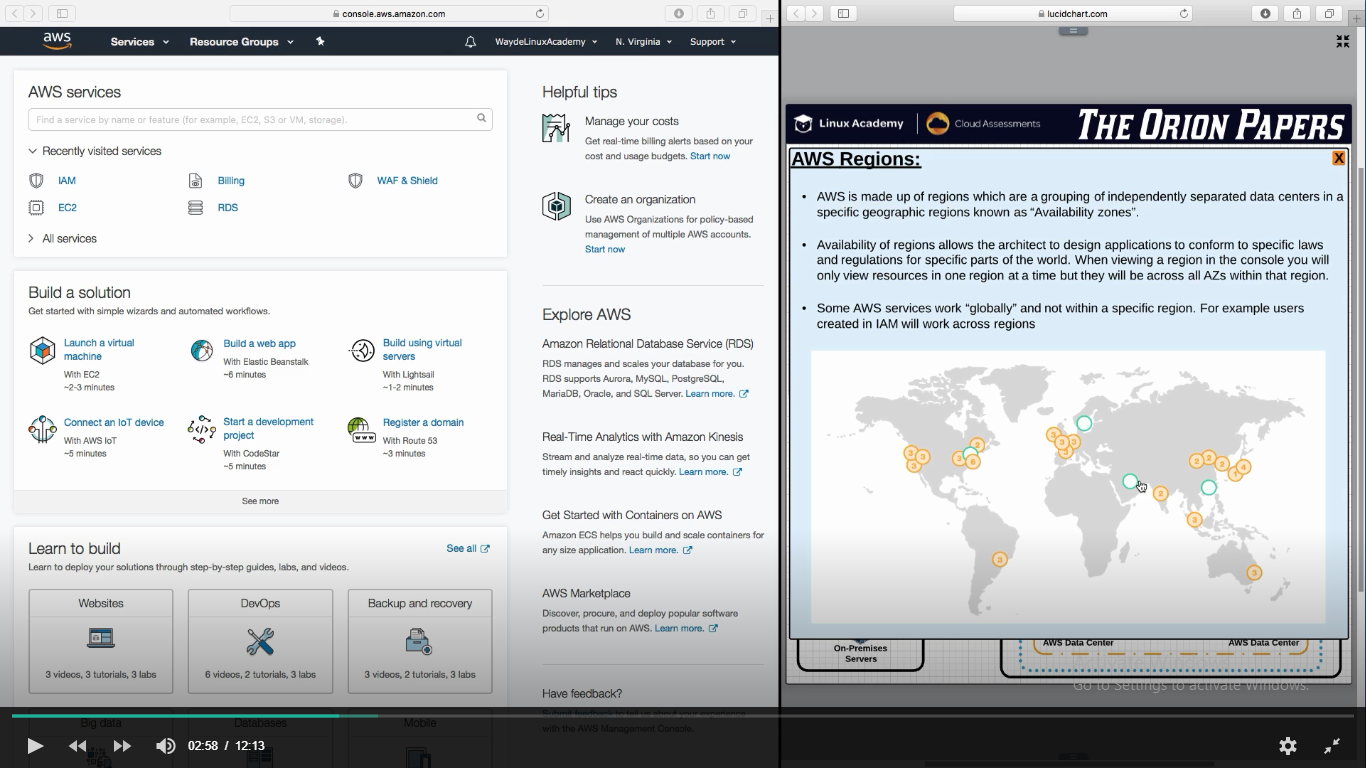
**AZ – Availability Zones**

This speaks about where the AWS servers are located and how we access them

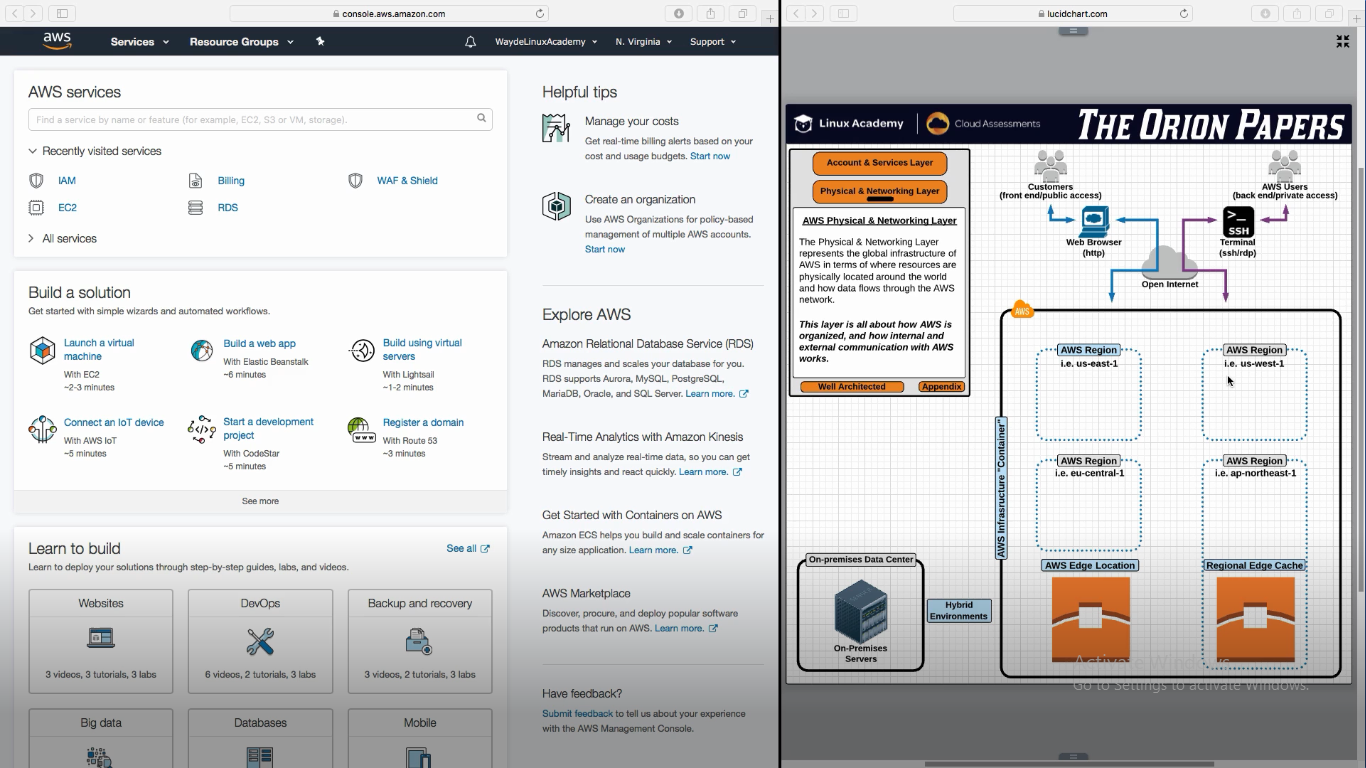
**Below are the active AWS regions**

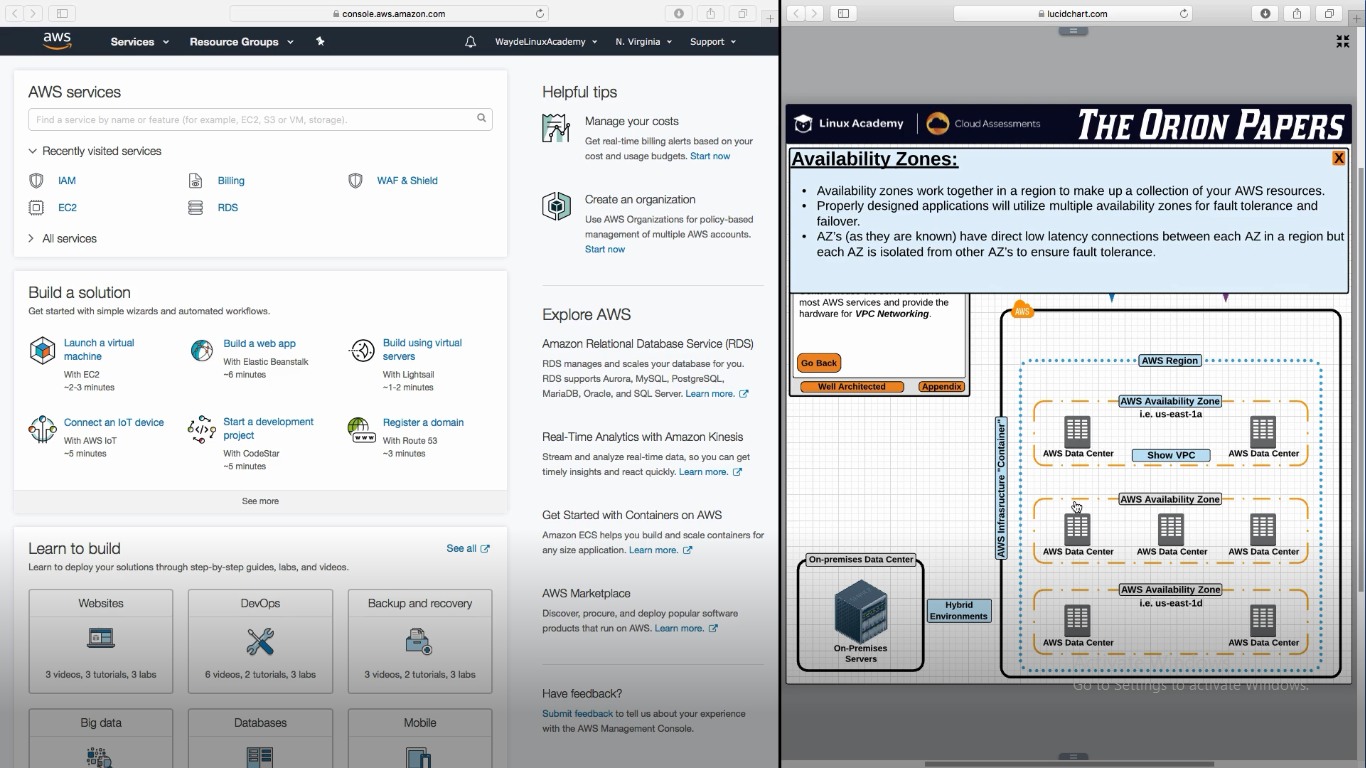
**Data 70** regulation. If a customer is in EU, their data’s also should be in EU region

Each region operates Autonomously

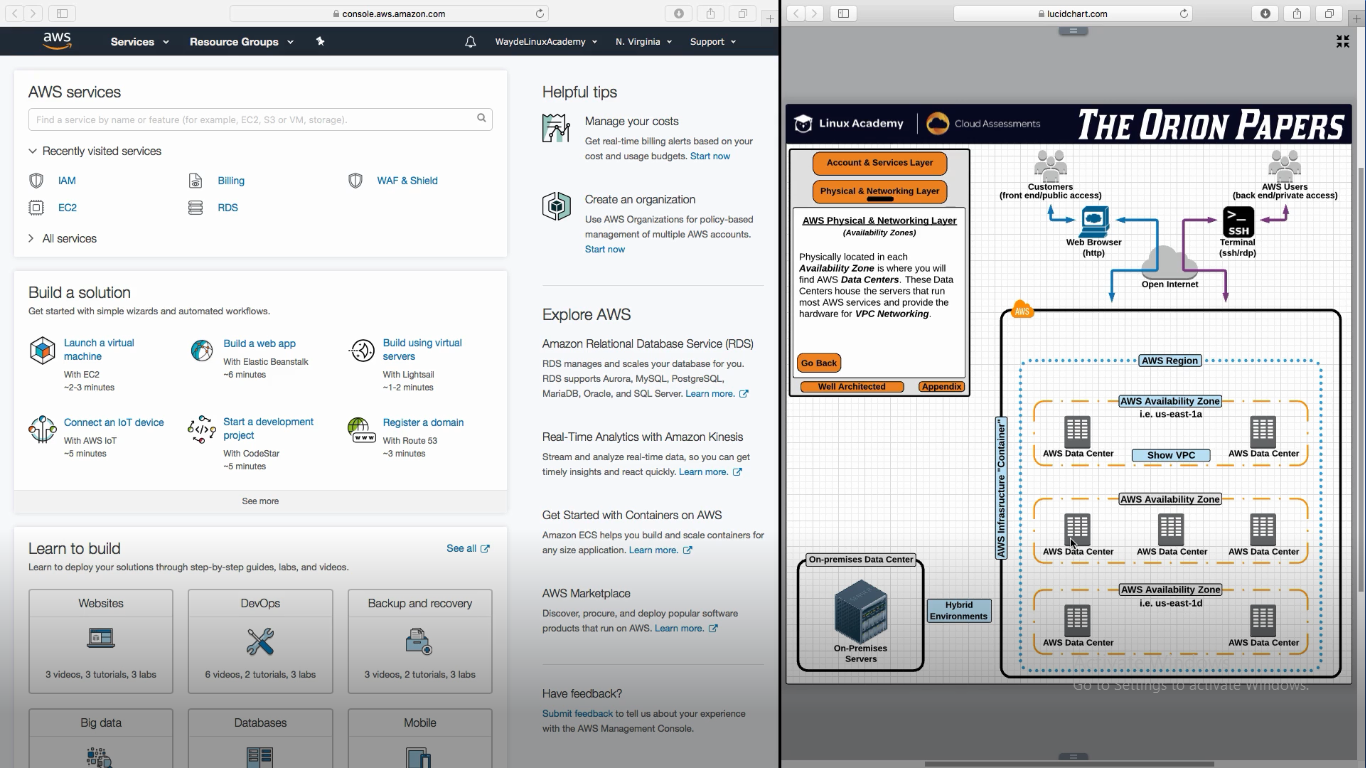


**Physical and Networking Layer**

****

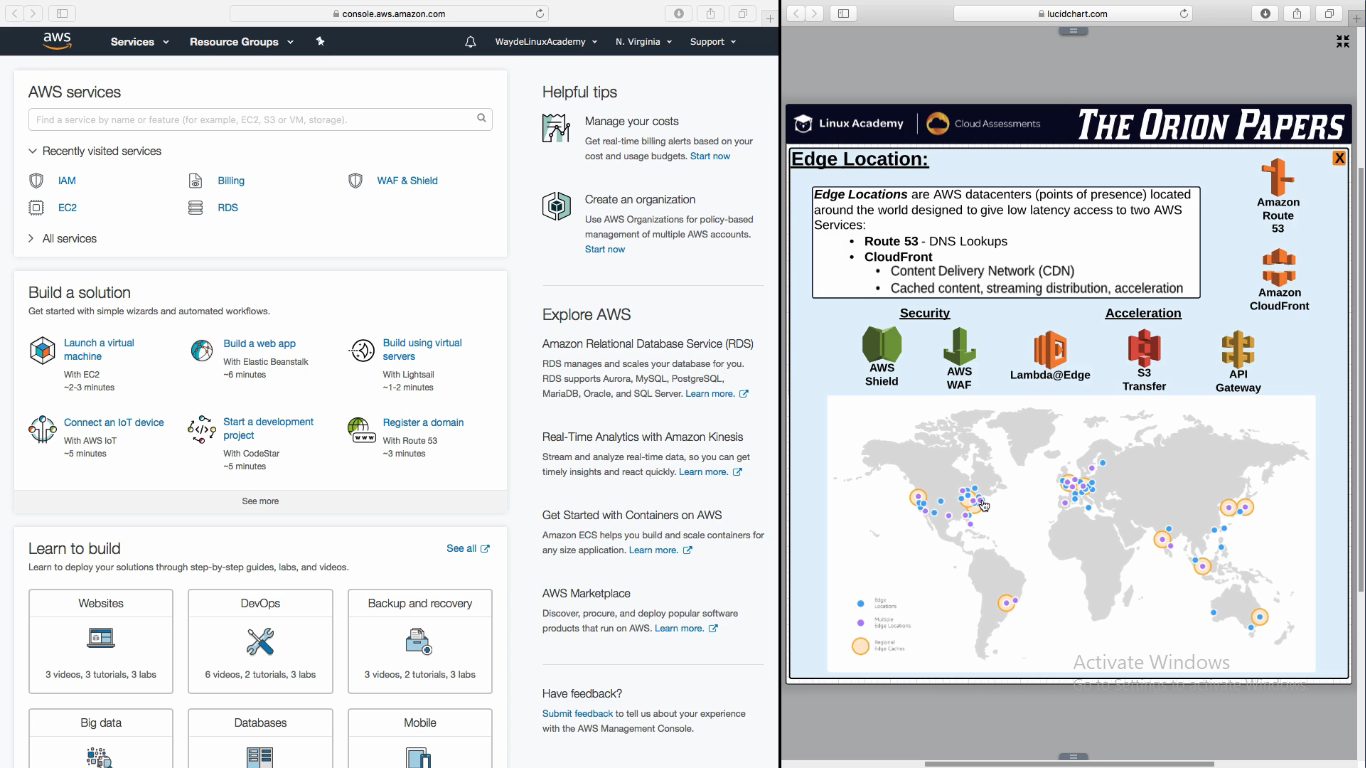
****

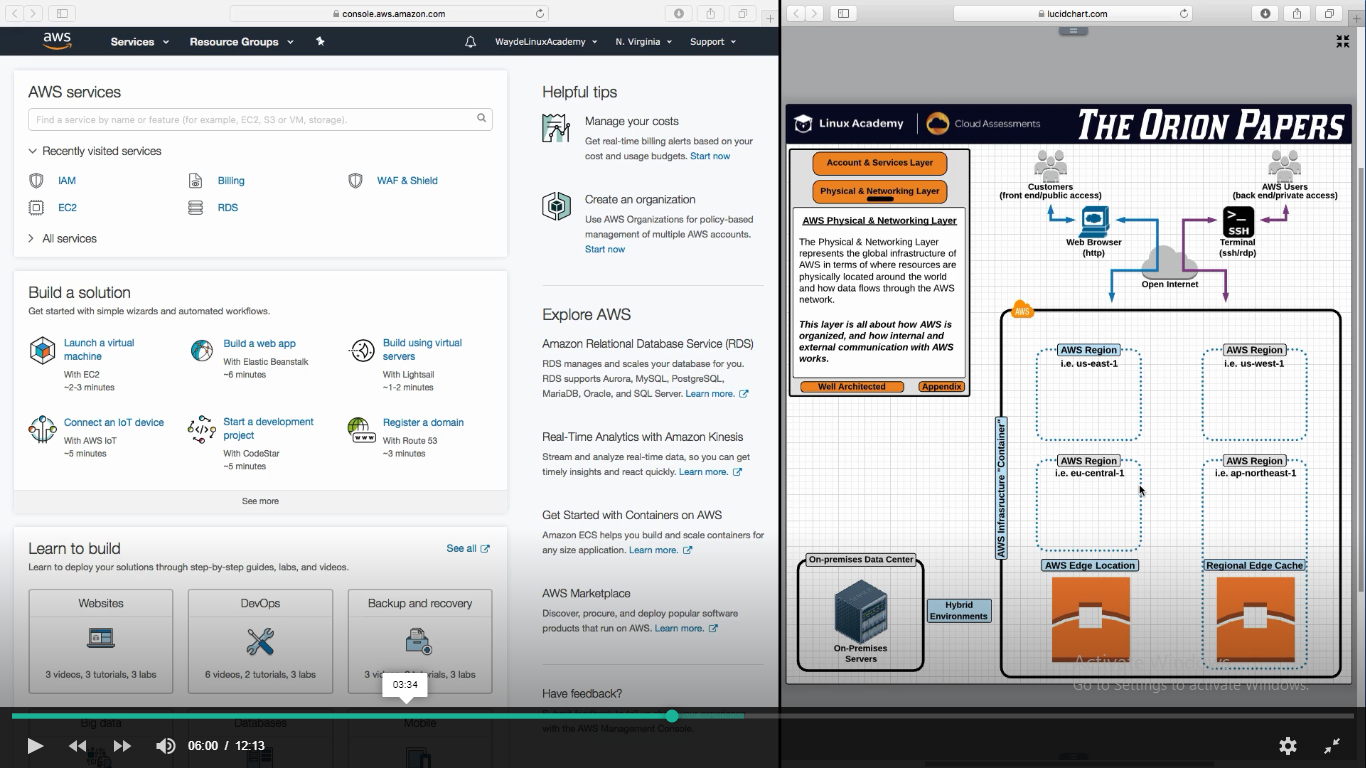
Data Centers in each AZ is connected by fiber optic which provides less latency

****

**AWS Edge Location -**Another set of Data Center

In addition to region, AWS have another set of data Centers called as AWS Edge Location. These are not affiliated to regions. It runs two services

****

****

1. **Route 53 :**
   1. It is the name of the service used for DNS lookups
   2. DNS lookup (if we type aws website with uname and password, browser will lookup to Route 53 to get back an IP address for aws website). To make this lookup quickly, Route 53 runs on edge location
2. **CloudFront:**

**Content Delivery Network (CDN)** – Frequently accessed data are stored here

* 1. Reduce latency, streaming, accelerates performance
  2. IT accelerate all AWS Services
     1. Eg : video streaming by keeping the file in the nearest edge location

Apart from CDN, CloudFront have lot of features. Below are they

* 1. Security and Acceleration
     1. **AWS Shield**: Block the attackers when they try to attack the application. It is a DDOS Mitigation service.
     2. **AWS WAF** : Web Application Firewall – Block application layer attack

Cloud Front can be used to accelerate other AWS Services

* + 1. **Lambda Compute Service** : If we put the code (Lambda function) in it, it will automatically execute it. We can make it to run in edge location and it will be updated in the region
    2. S3 Transfer
    3. API Gateway
       1. Our application will be running on any of the regions
       2. API Gateway allows us to define an API for our services for users to access our application API
       3. API Gateway allows us to define an endpoint API for our custom application and have it propagated to all edge locations
       4. So when user access our application, it comes to edge location and accelerates the response

**Regional Edge Cache:**

Another type of datacenter location. This is a feature of Cloud Front.

Cloud front runs on Global Edge location

If edge location data is full, it comes to regional edge cache.

