



What's AWS?

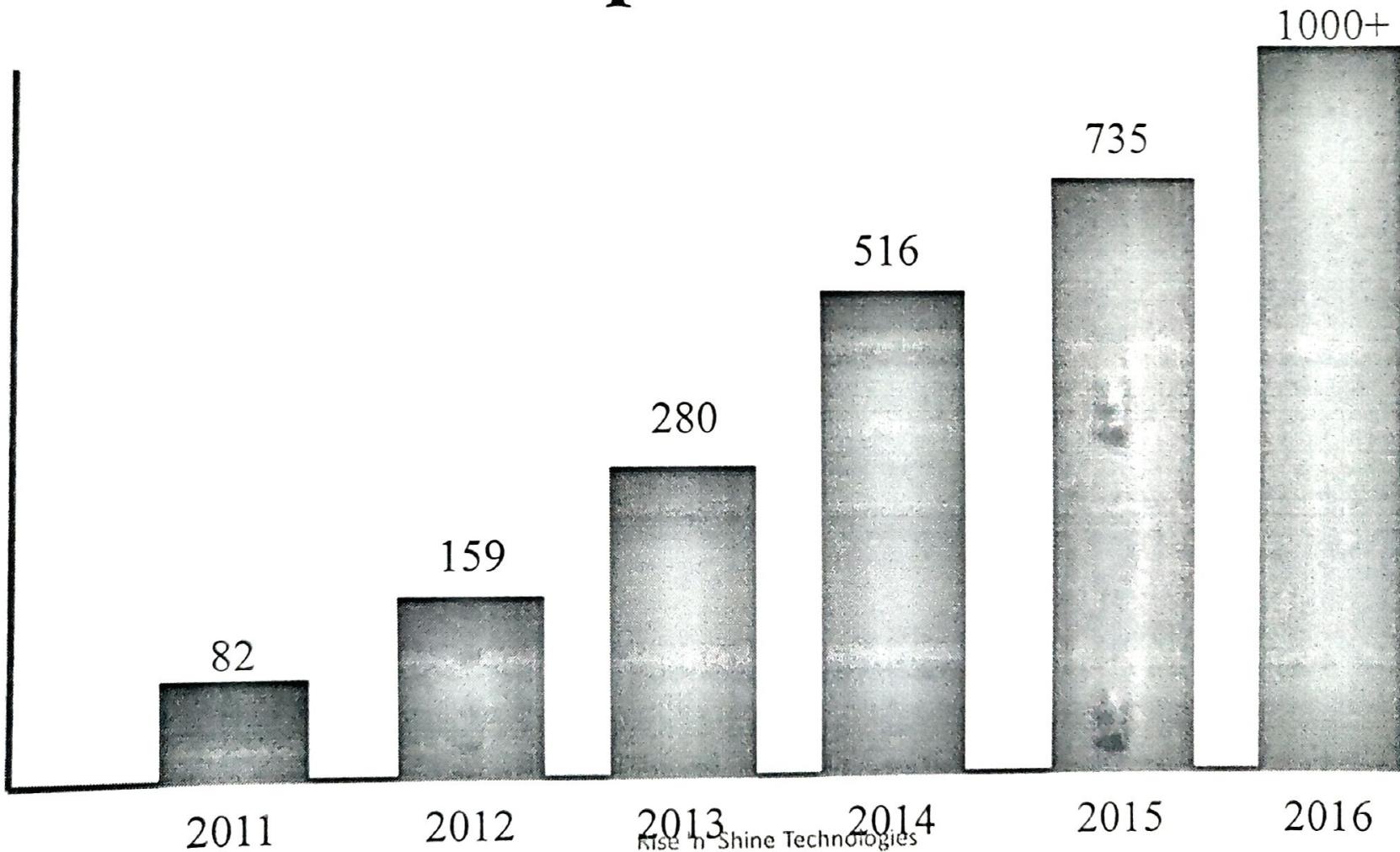


- AWS (Amazon Web Services) is a Cloud Provider
- They provide you with servers and services that you can use on demand and scale easily
- AWS has revolutionized IT over time
- AWS powers some of the biggest websites in the world
 - Amazon.com
 - Netflix

Why Learn AWS?

- Fastest growing cloud computing platform on the Planet
- Largest public cloud computing platform on the planet
- More and more organizations are outsourcing their IT to AWS

AWS new Service Announcements & Updates



About AWS Certifications

Specialty

Security

**Advanced
Networking**

Big Data

Professional
Tier

**Certified Solutions
Architect
Professional**

**DevOps
Professional**

Associate
Tier

**Certified Solutions
Architect Associate**

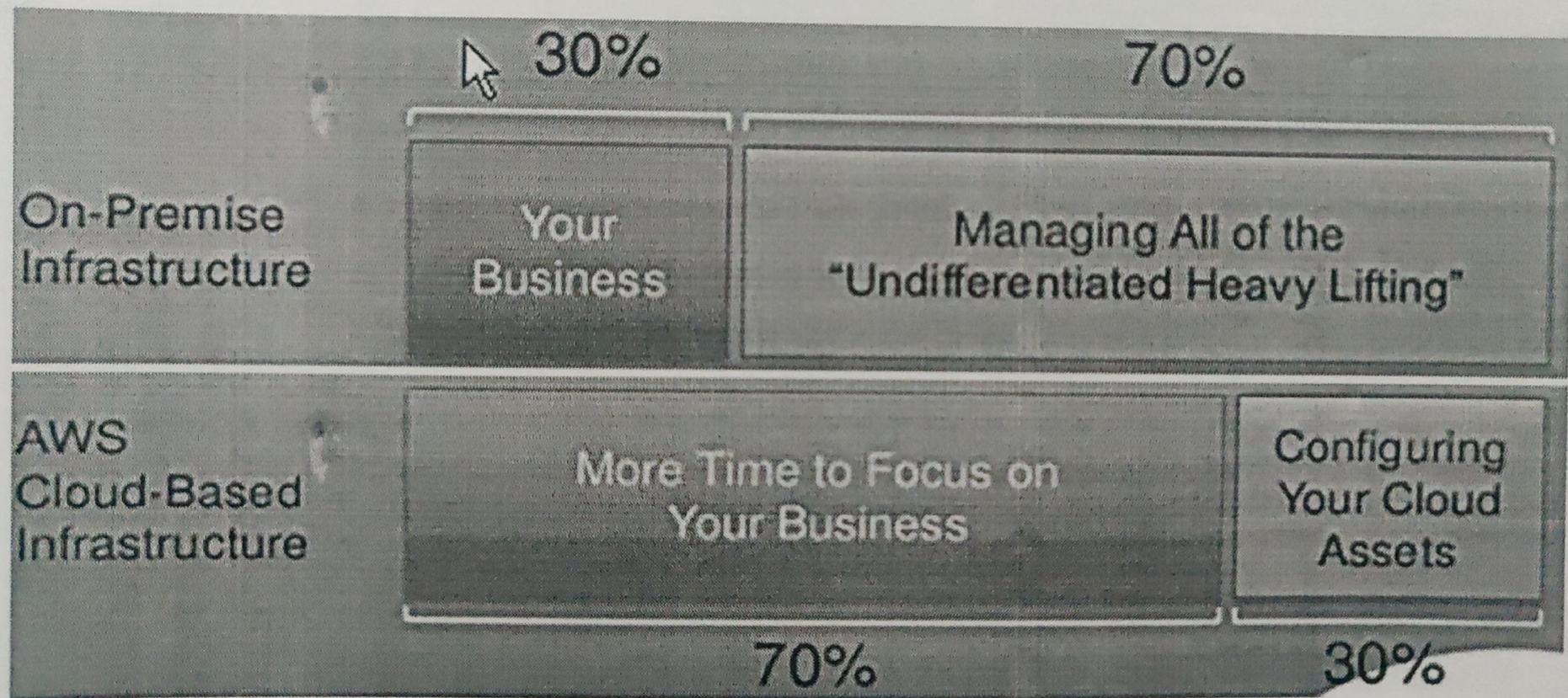
**Certified Developer
Associate**

**Certified SysOps
Administrator
Associate**

Gartner's Magic quadrant

- In Aug 2016, AWS was named as a leader in the IAAS Magic Quadrant for the 6th consecutive Year.
- 90% of the Cloud market by AWS
- 5% of the Cloud Market by Microsoft

AWS Goal



AWS Platform

Game Development

Artificial Intelligence

Messaging

Business Productivity

Internet Of Things

Desktop & App Streaming

Application Services

Developer Tools

Mobile Services

Analytics

Security & Identity

Mgmt Tools

Migration

Storage

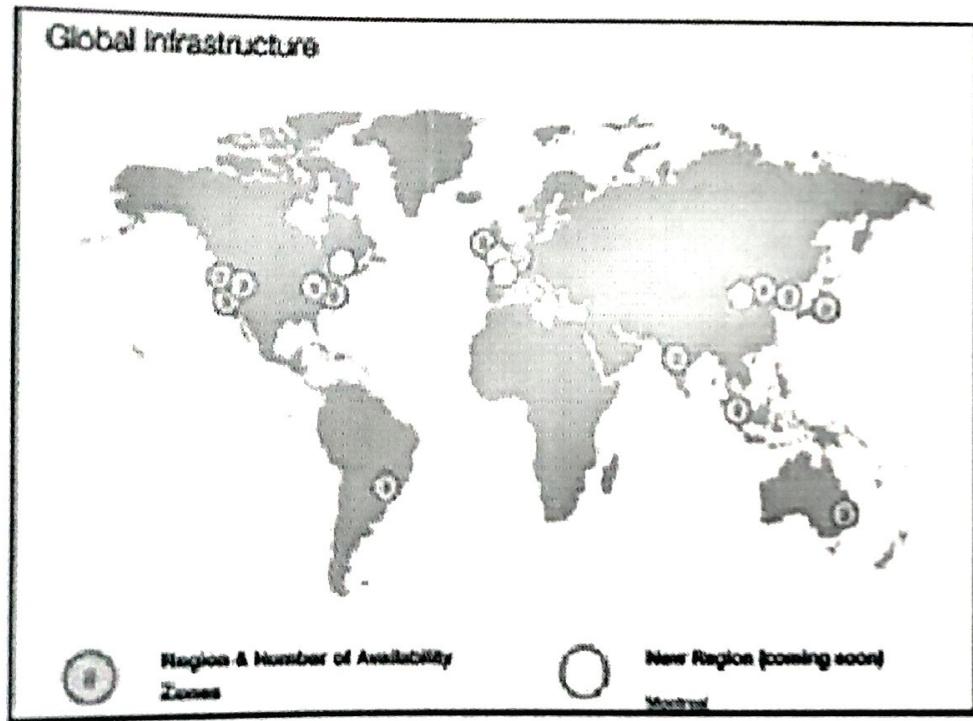
Databases

Networking & Content Delivery

Compute

AWS Global Infrastructure

AWS Global Infrastructure



Region & Number of Availability Zones

US East	China
N. Virginia (6), Ohio (3)	Beijing (2)
Europe	
US West	Frankfurt (3), Ireland (3), London (2)
Oregon (3)	
South America	
Asia Pacific	São Paulo (3)
Mumbai (2), Seoul (2), Singapore (2), Sydney (3), Tokyo (3)	AWS GovCloud (US-West) (2)
Canada	
Central (2)	



New Region (coming soon)

Bahrain

China



France

Hong Kong

Sweden

AWS GovCloud (US-East)

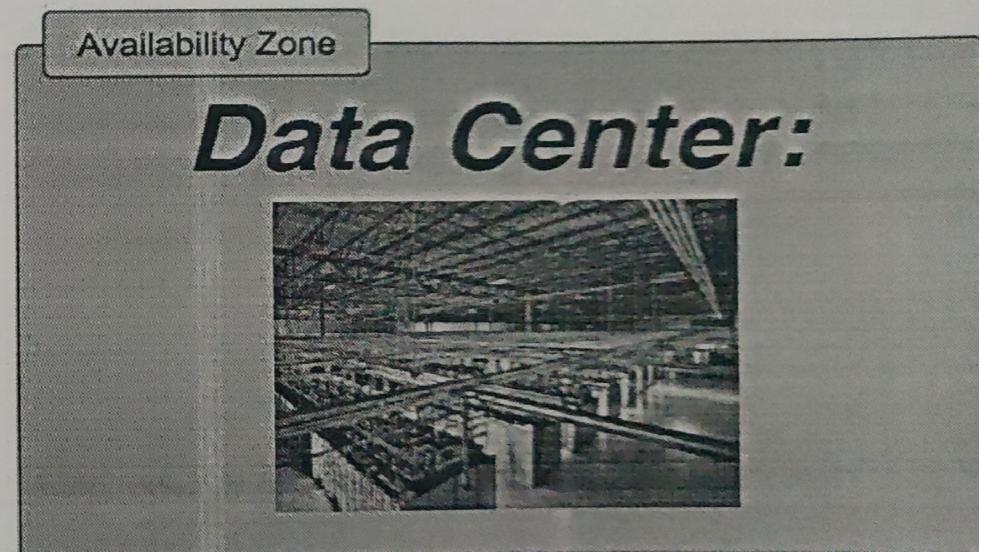
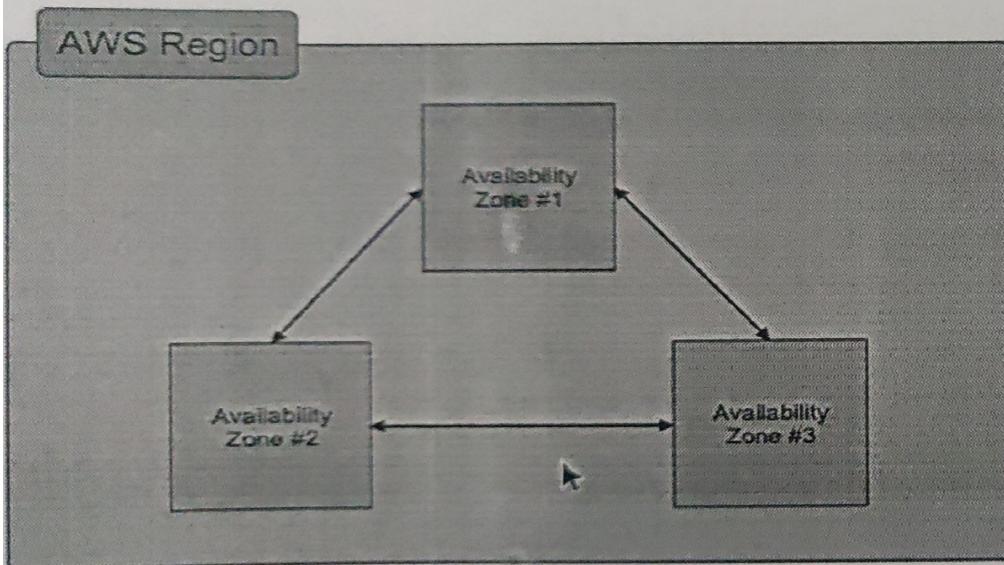
22 Regions & 69 Availability Zones – by 2019
3 More Regions & 9 More Availability Zones

What is a Region & AZ

- A Region is a Geographical area.
Each Region consists of 2 (or more) Availability Zones.
- An Availability zone(AZ) is a Simply a Data Center.
- A list of regions and their corresponding codes is provided here for your reference.
- The code is basically how AWS refers to its multiple regions;

Region	Name
US West (Oregon) Region	us-west-2
US West (N. California) Region	us-west-1
US East (Ohio) Region	us-east-2
US East (N. Virginia) Region	us-east-1
Asia Pacific (Mumbai) Region	ap-south-1
Asia Pacific (Seoul) Region	ap-northeast-2
Asia Pacific (Singapore) Region	ap-southeast-1
Asia Pacific (Sydney) Region	ap-southeast-2
Asia Pacific (Tokyo) Region	ap-northeast-1
Canada (Central) Region	ca-central-1
China (Beijing) Region	cn-north-1
EU (Frankfurt) Region	eu-central-1
EU (Ireland) Region	eu-west-1

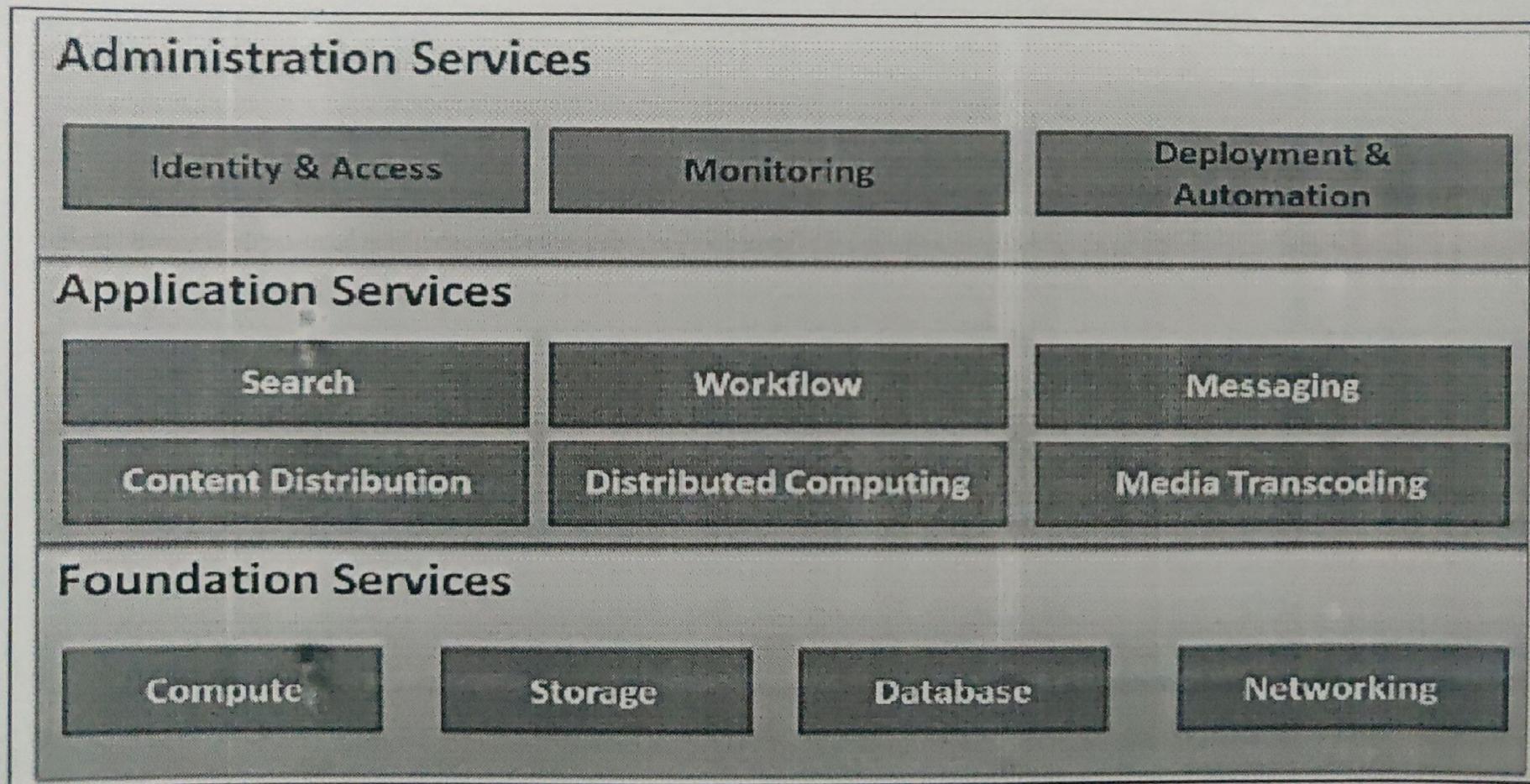
What is a Region & AZ



AWS Platform Overview

- The AWS platform consists of a variety of services that you can use either in isolation or in combination based on your organization's needs.
 - Foundation services
 - Application services
 - Administration services

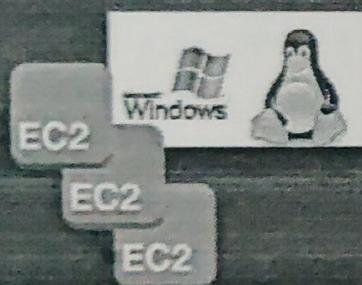
AWS Platform Services



Compute Services

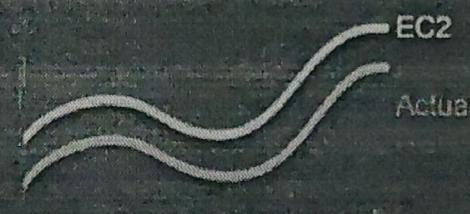
Amazon EC2

Elastic Virtual servers
in the cloud



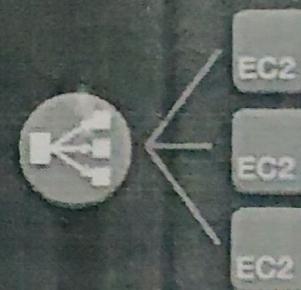
Auto Scaling

Automated scaling
of EC2 capacity



Elastic Load Balancing

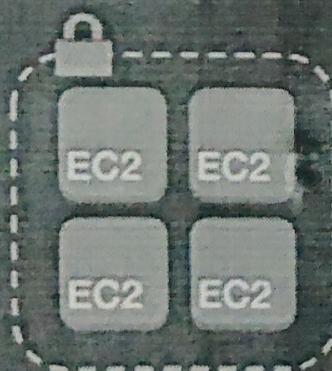
Dynamic traffic
distribution



Networking Services

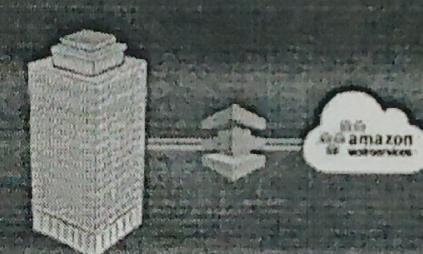
Amazon VPC

Private, isolated
section of the AWS
Cloud



AWS DirectConnect

Private connectivity
between AWS and your
datacenter



Amazon Route 53

Domain Name System
(DNS) web service.



Storage Services

Amazon EBS

Block storage for use with Amazon EC2



Amazon S3

Internet scale storage via API



Images
Videos
Files
Binaries
Snapshots

Amazon Glacier

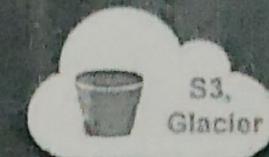
Storage for archiving and backup



Images
Videos
Files
Binaries
Snapshots

AWS Storage Gateway

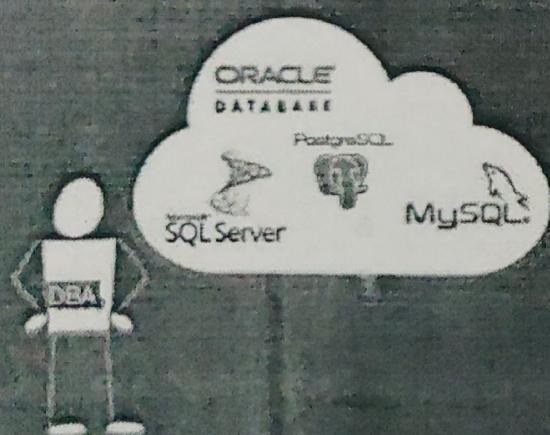
Integrates on-premises IT and AWS storage



Database Services

Amazon RDS

Managed relational database service



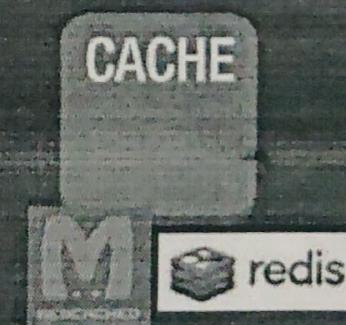
Amazon DynamoDB

Managed NoSQL database service



Amazon ElastiCache

In-Memory Caching Service



Administration

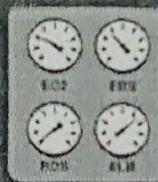
AWS IAM (Identity & Access Mgmt)

Manage users,
groups &
permissions



Amazon CloudWatch

Monitor resources



AWS CloudTrail

AWS API call logging for
governance &
compliance



Application Services

Amazon CloudFront

Distribute content
globally



Amazon CloudSearch

Managed search
service



Amazon Elastic Transcoder

Video transcoding
in the cloud



Deployment

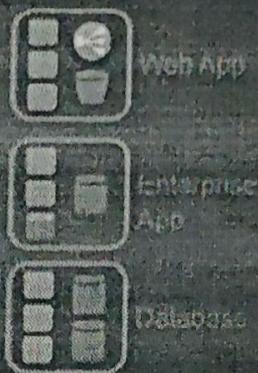
AWS OpsWorks

Dev-Ops framework
for application lifecycle
management



AWS CloudFormation

Templates to deploy
& manage



AWS Elastic Beanstalk

Automate resource
management



AWS Sign Up!!!



Getting started with AWS

- A Free Tier
- No charge for a period of 12 months from the date of the actual signup
- For a complete insight into the free tier usage, check
 - <http://aws.amazon.com/free/>

AWS Free Tier Account

AWS Product	What's free?
Amazon EC2	750 hours per month of Linux and Windows micro instance usage
Amazon S3	5 GB of standard storage 20,000 get requests 2,000 put requests
Amazon RDS	750 Hours of RDS Single-AZ micro instance 20 GB of DB Storage: any combination of general purpose (SSD) or magnetic 20 GB for backups 10,000,000 I/Os
Amazon ELB	750 hours per month 15 GB of data processing

Sign Up for AWS

- <http://aws.amazon.com/>

The screenshot shows the official AWS website homepage. At the top, there's a dark navigation bar with the AWS logo, a search bar containing "nu", and various menu items: Contact Sales, Products, Solutions, Pricing, Getting Started, More, English, My Account, and a prominent "Sign In to the Console" button.

The main content area features a large banner with a dark background. On the left, the text "Introducing EC2 P3 Instances" is displayed above a subtext: "Accelerate machine learning and HPC workloads with the most powerful GPU compute instances". Below this text is a "Learn more »" link. To the right of the text is a graphic showing several server racks connected by a network of dashed lines, with one specific rack labeled "GPU".

On the far right of the banner, there's a call-to-action box with a white background and a thin gray border. It contains the text "Manage Your Resources" above a "Sign In to the Console" button. Below this is another section with the text "AWS Console Mobile App" followed by "View your resources on iOS and Android devices", and a "Download the Mobile App »" link at the bottom.

Sign Up for AWS



- Provide a valid email address and choose a password
 - Provide your contact information and set your preferences
-
- Provide information about your payment method
-
- Provide a phone number you can be reached at in the next few minutes
 - Type the on-screen phone number into your phone's keypad
-
- Choose one of the available AWS Support plans

Sign Up for AWS

In the Login page, Create a new AWS account

Name, E-mail address

Password, Confirm Password

Click on Create account when done.

The next screen is the Contact Information page. Provide your Full Name, Company Name, Country, Address, City, Postal Code, and Phone Number as requested. Check the Amazon Internet Service Pvt. Ltd. Customer Agreement checkbox and select the Create Account and continue options.

Rise 'n' Shine Technologies

Sign in 

Email address of your AWS account

To sign in as an IAM user, enter your [account ID](#) or [account alias](#) instead.

Next

New to AWS?

[Create a new AWS account](#)

Sign Up for AWS

- Enter a suitable Cardholder's Name and your Credit/Debit Card Number in the Payment Information page as shown:

Payment Information

Please enter your payment information below. You will be able to try a broad set of AWS products for free via the Free Usage Tier. We will only bill your credit card for usage that is not covered by our Free Usage Tier.

AWS Free Usage Tier free for 1 year	Compute Amazon EC2 750hrs/month*	Storage Amazon S3 5GB	Database Amazon RDS 750hrs/month*
--	--	-----------------------------	---

*View full offer details »

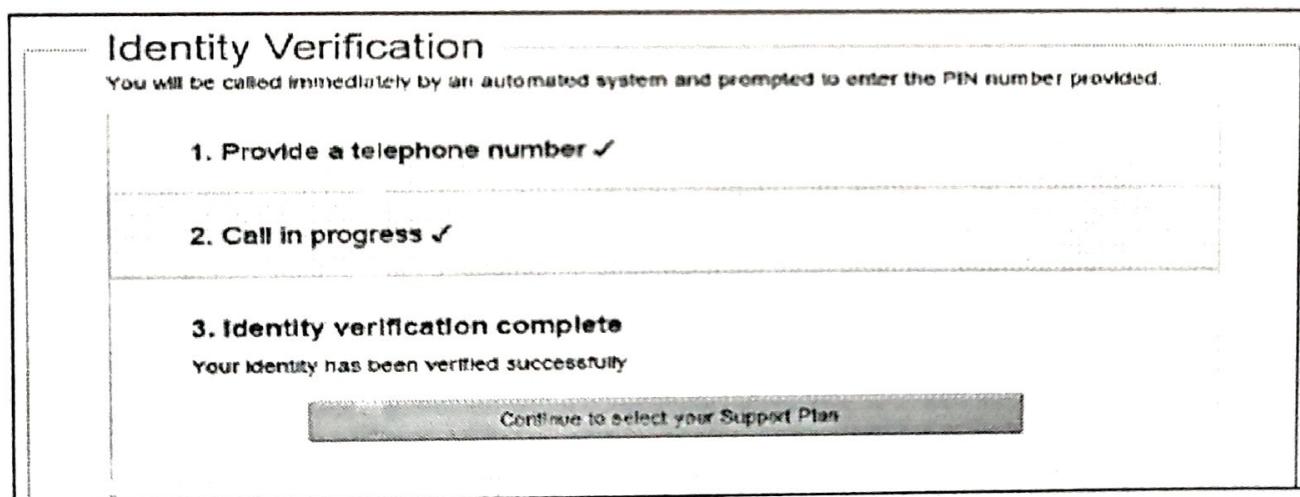
Cardholder's Name

Credit/Debit Card Number
   

Expiration Date

Sign Up for AWS

- The last part of the signup process is the **Identity Verification process**



Introducing the AWS Management Console

AWS Management Console

AWS services

Find services You can enter names, identifiers or keywords.

Search:

All services

Build a solution Get started with simple, ready-to-use AWS solutions.

Launch a virtual machine With EC2 ~2-5 minutes 	Build a web app With Elastic Beanstalk ~5 minutes 	Build using virtual servers With Lightsail ~1-2 minutes 	Connect an IoT device With AWS IoT ~5 minutes 
Start a development project With CodeStar ~5 minutes 	Register a domain With Route 53 ~5 minutes 	Deploy a serverless microservice With Lambda, API Gateway ~2 minutes 	Create a backend for your mobile app With Mobile Hub ~5 minutes 

Access resources on the go

Access the Management Console using the AWS Console Mobile App. [Learn more](#)

Explore AWS

Amazon RDS Set up, operate, and scale your relational database in the cloud. [Learn more](#)

Run Serverless Containers with AWS Fargate AWS Fargate runs and scales your containers without having to manage servers or clusters. [Learn more](#)

Amazon Redshift Fast, simple, cost-effective data warehouse that can extend queries to your data lake. [Learn more](#)

Amazon SageMaker Build, train, and deploy machine learning models. [Learn more](#)

Introducing the AWS Management Console

- To the right-hand side
- The first your ‘**name**’ as an end user:
 - It will help you with configuring your account details, security credentials, and billing management.
- ‘**Region**’ from where you will currently be operating.
 - The US East (North Virginia) region is the cheapest region in AWS as it was one of the first regions to get set up and started.
- The final tab is the ‘**Support**’ tab:
 - Support Center, AWS Forums, and view the latest set of AWS Documentation

Introducing the AWS Management Console

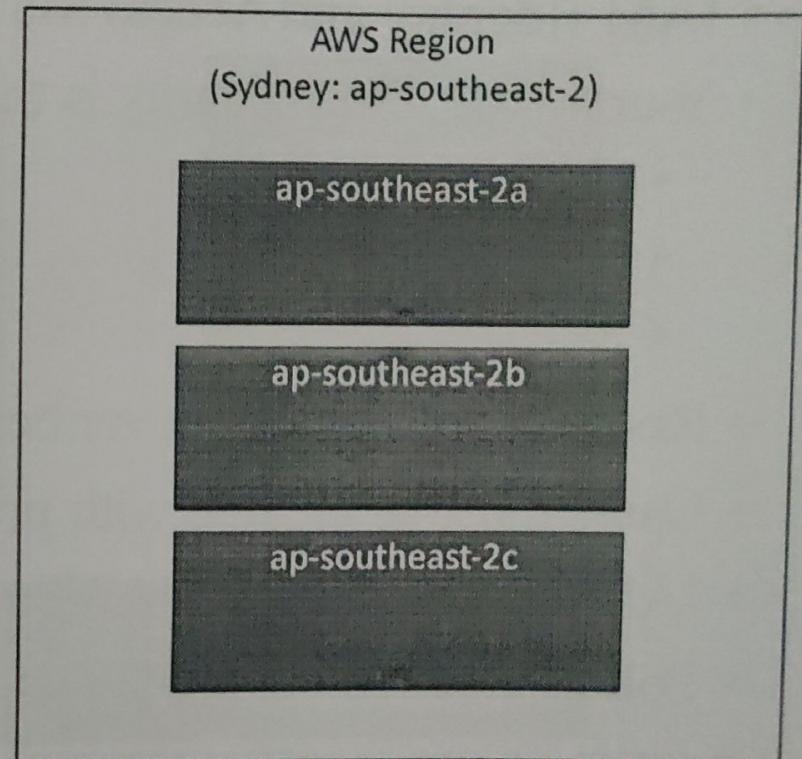
- To the left-hand side
- **Home Screen**
 - when clicked on will bring you to the AWS dashboard screen
- **Services**
 - which lists the AWS services according to their class.
 - It also has a history option to list recently used AWS services.
- **Resource Groups**
 - These are a collection of AWS resources that can be organized and viewed as per your requirements
- **Edit tab**
 - customize your toolbar by filling it with those AWS services that you use frequently

AWS Fundamentals – Part I

Regions, IAM & EC2

AWS Regions

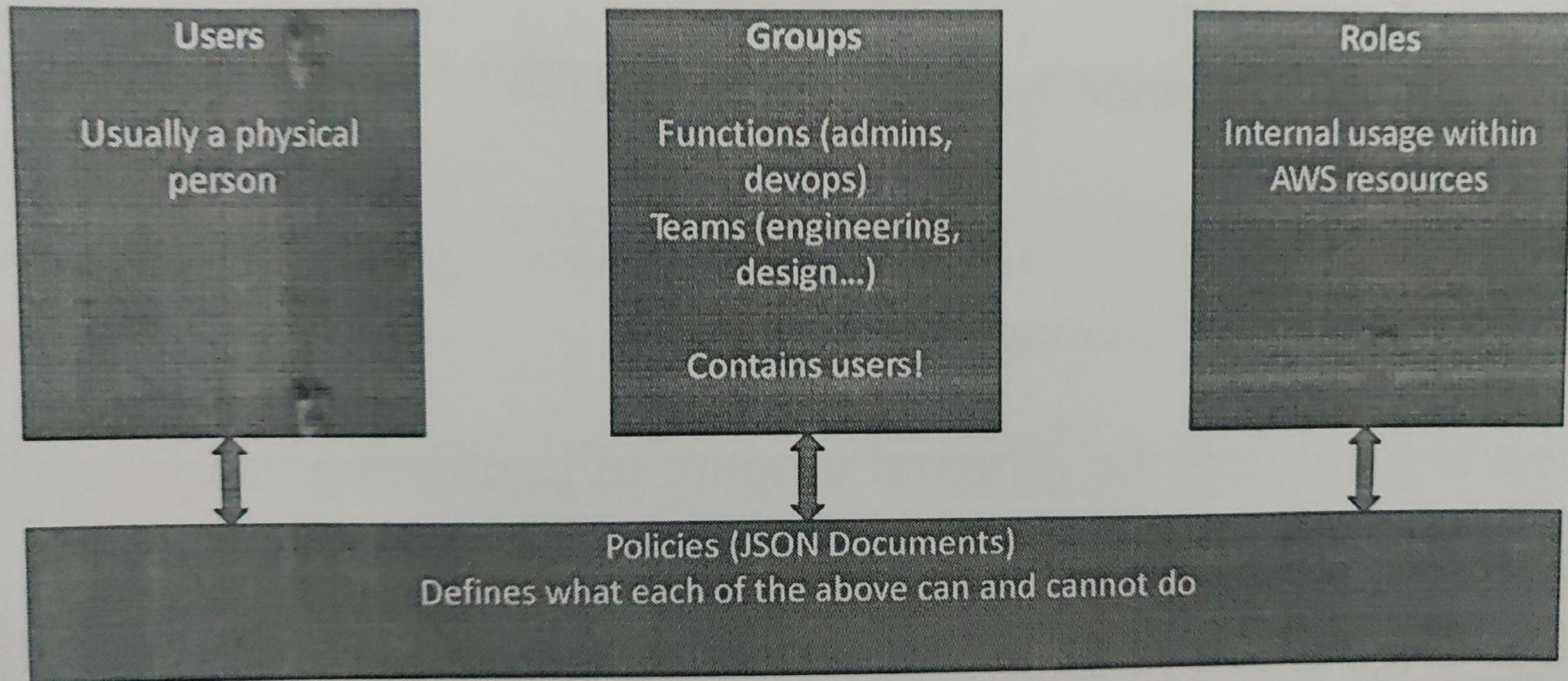
- AWS has regions all around the world (us-east-1)
- Each region has availability zones (us-east-1a, us-east-1b...)
- Each availability zone is a physical data center in the region, but separate from the other ones (so that they're isolated from disasters)
- AWS Consoles are region scoped (except IAM, S3 & Route53)



IAM Introduction

- IAM (Identity and Access Management)
- Your whole AWS security is there:
 - Users
 - Groups
 - Roles
- Root account should never be used (and shared)
- Users must be created with proper permissions
- IAM is at the center of AWS
- Policies are written in JSON (JavaScript Object Notation)

IAM Introduction



IAM Introduction

- IAM has a global view
- Permissions are governed by Policies (JSON)
- MFA (Multi Factor Authentication) can be setup
- IAM has predefined “managed policies”
- It’s best to give users the minimal amount of permissions they need to perform their job (least privilege principles)

IAM Federation

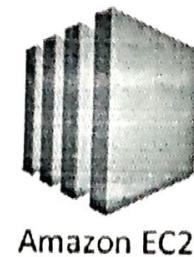
- Big enterprises usually integrate their own repository of users with IAM
- This way, one can login into AWS using their company credentials
- Identity Federation uses the SAML standard (Active Directory)

IAM Important Points:

- One IAM User per physical person
- One IAM Role per Application
- IAM credentials should NEVER BE SHARED
- Never, ever, ever, ever, write IAM credentials in code. EVER.
- And even less, NEVER EVER EVER COMMIT YOUR IAM credentials
- Never use the ROOT account except for initial setup.
- Never use ROOT IAM Credentials

What is EC2?

- EC2 is one of most popular of AWS offering
- It mainly consists in the capability of :
 - Renting virtual machines (EC2)
 - Storing data on virtual drives (EBS)
 - Distributing load across machines (ELB)
 - Scaling the services using an auto-scaling group (ASG)
- Knowing EC2 is fundamental to understand how the Cloud works

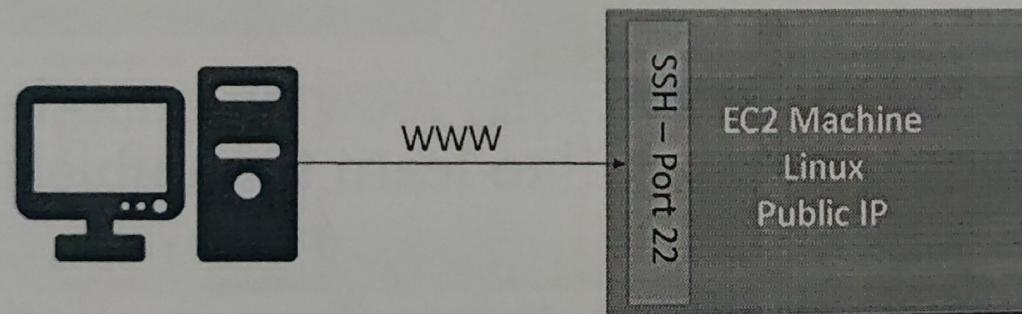


Hands-On: Launching an EC2 Instance running Linux

- We'll be launching our first virtual server using the AWS Console
- We'll get a first high level approach to the various parameters
- We'll learn how to start / stop / terminate our instance.

How to SSH into your EC2 Instance Linux

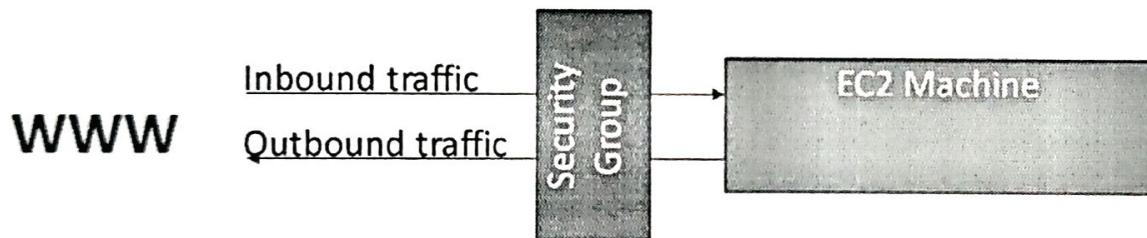
- We'll learn how to SSH into your EC2 instance using Linux
- SSH is one of the most important function. It allows you to control a remote machine, all using the command line.



- `$ ssh -i keypair.pem ec2-user@Public_IP`

Introduction to Security Groups

- Security Groups are the fundamental of network security in AWS
- They control how traffic is allowed into or out of our EC2 Machines.



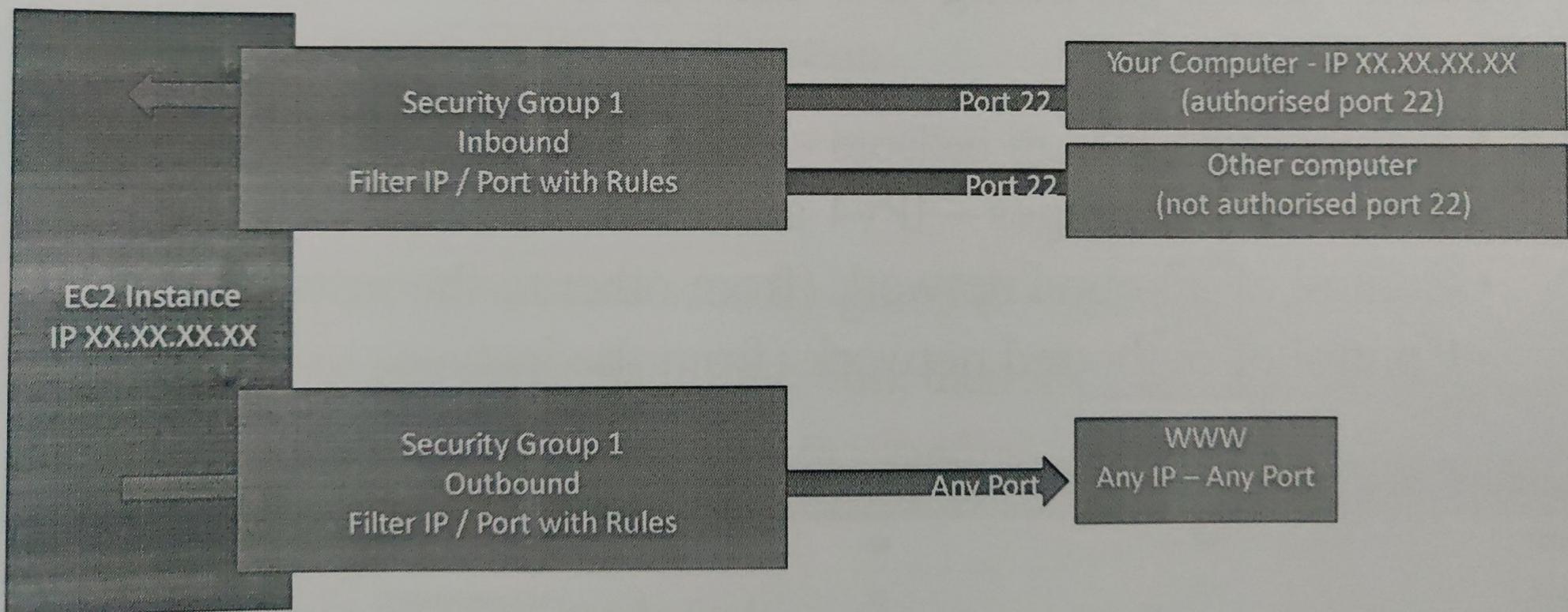
- It is the most fundamental skill to learn to troubleshoot networking issues
- In this lecture, we'll learn how to use them to allow, inbound and outbound ports

Security Groups Deeper Dive

- Security groups are acting as a “firewall” on EC2 instances
- They regulate:
 - Access to Ports
 - Authorised IP ranges – IPv4 and IPv6
 - Control of inbound network (from other to the instance)
 - Control of outbound network (from the instance to other)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	test http page
SSH	TCP	22	122.149.196.85/32	
Custom TCP Rule	TCP	4567	0.0.0.0/0	java app

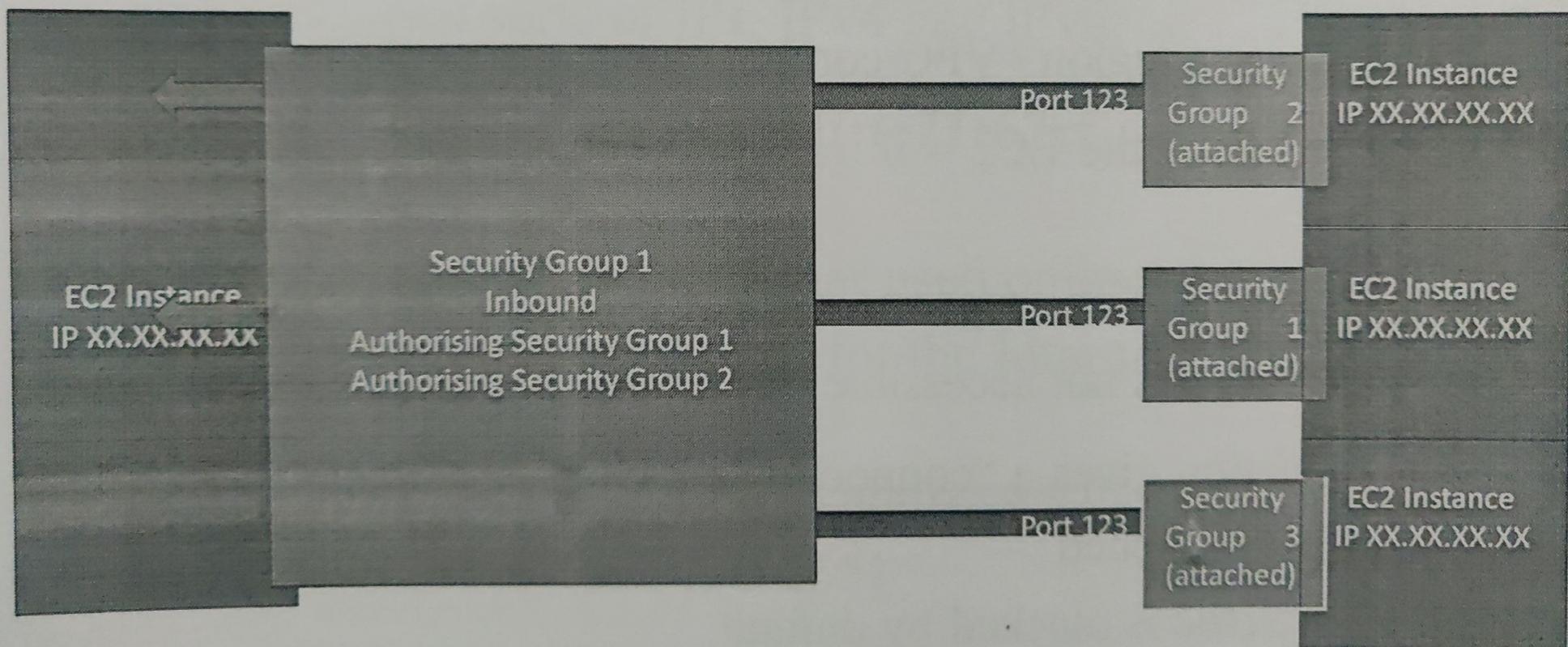
Security Groups Diagram



Security Groups Good to know

- Can be attached to multiple instances
- Locked down to a region / VPC combination
- Does live “outside” the EC2 – if traffic is blocked the EC2 instance won’t see it
- It’s good to maintain one separate security group for SSH access
- If your application is not accessible (time out), then it’s a security group issue
- If your application gives a “connection refused“ error, then it’s an application error or it’s not launched
- All inbound traffic is blocked by default
- All outbound traffic is authorised by default

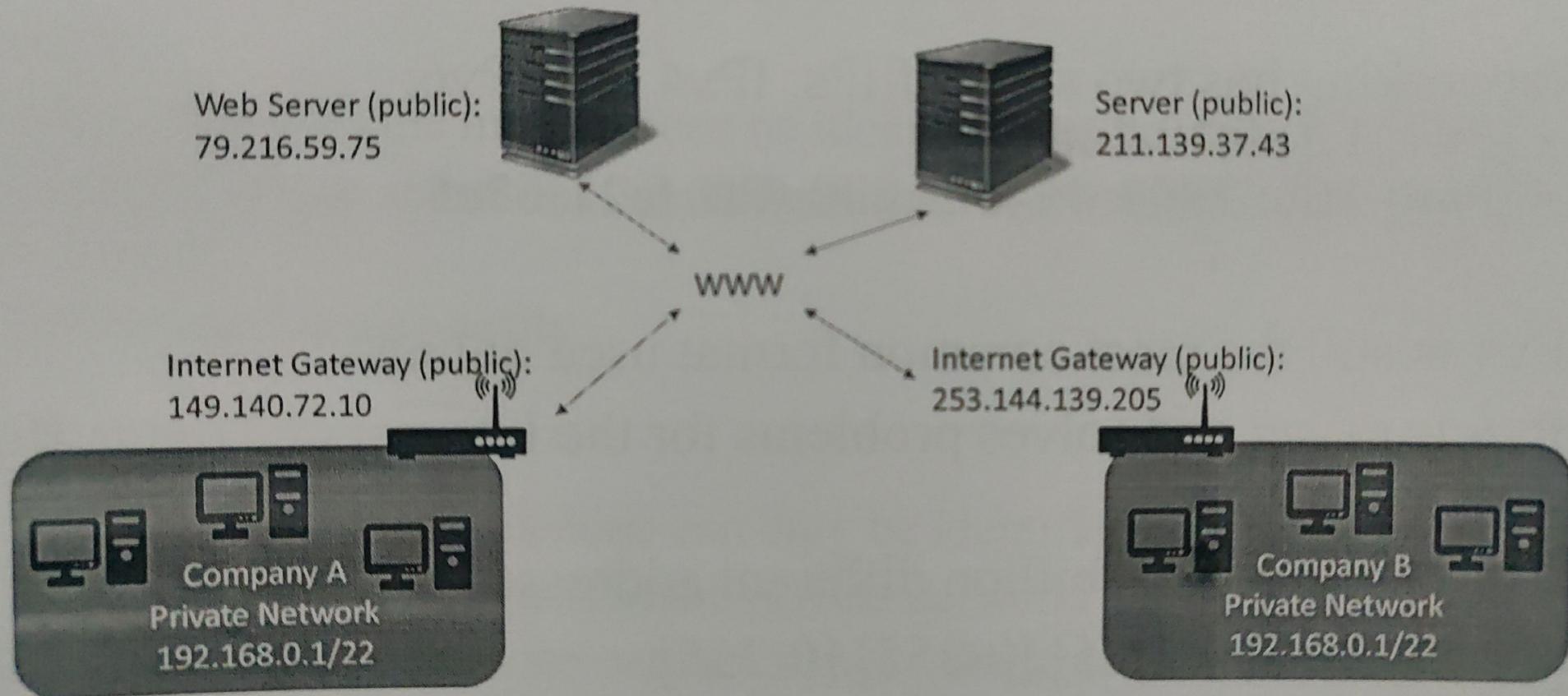
Referencing other security groups Diagram



Private vs Public IP (IPv4)

- Networking has two sorts of IPs. IPv4 and IPv6:
 - IPv4: **1.160.10.240**
 - IPv6: **3ffe:1900:4545:3:200:f8ff:fe21:67cf**
- IPv4 is still the most common format used online.
- IPv6 is newer and solves problems for the Internet of Things (IoT).
- IPv4 allows for 3.7 billion different addresses in the public space
- IPv4: [0-255].[0-255].[0-255].[0-255].

Private vs Public IP (IPv4) Example



Private vs Public IP (IPv4) Fundamental Differences

- Public IP:
 - Public IP means the machine can be identified on the internet (WWW)
 - Must be unique across the whole web (not two machines can have the same public IP).
 - Can be geo-located easily
- Private IP:
 - Private IP means the machine can only be identified on a private network
 - The IP must be unique across the private network
 - BUT two different private networks (two companies) can have the same IPs.
 - Machines connect to WWW using an internet gateway (a proxy)
 - Only a specified range of IPs can be used as private IP

Elastic IPs

- When you stop and then start an EC2 instance, it can change its public IP.
- If you need to have a fixed public IP for your instance, you need an Elastic IP
- An Elastic IP is a public IPv4 IP you own as long as you don't delete it
- You can attach it to one instance at a time

Elastic IP

- With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.
- You can only have 5 Elastic IP in your account (you can ask AWS to increase that).
- Overall, try to avoid using Elastic IP:
 - They often reflect poor architectural decisions
 - Instead, use a random public IP and register a DNS name to it
 - Or, as we'll see later, use a Load Balancer and don't use a public IP

Private vs Public IP (IPv4) In AWS EC2 – Hands On

- By default, your EC2 machine comes with:
 - A private IP for the internal AWS Network
 - A public IP, for the WWW.
- When we are doing SSH into our EC2 machines:
 - We can't use a private IP, because we are not in the same network
 - We can only use the public IP.
- If your machine is stopped and then started, the public IP can change

Launching an Apache Server on EC2

- Let's leverage our EC2 instance
- We'll install an Apache Web Server to display a web page
- We'll create an index.html that shows the hostname of our machine

EC2 User Data

- It is possible to bootstrap our instances using an EC2 User data script.
- bootstrapping means launching commands when a machine starts
- That script is only run once at the instance first start
- EC2 user data is used to automate boot tasks such as:
 - Installing updates
 - Installing software
 - Downloading common files from the internet
 - Anything you can think of
- The EC2 User Data Script runs with the root user

EC2 User Data Hands-On

- We want to make sure that this EC2 instance has an Apache HTTP server installed on it – to display a simple web page
- For it, we are going to write a user-data script.
- This script will be executed at the first boot of the instance.
- Let's get hands on!

EC2 Instance Launch Types

- *On Demand Instances*: short workload, predictable pricing
- *Reserved Instances*: long workloads (≥ 1 year)
- *Convertible Reserved Instances*: long workloads with flexible instances
- *Scheduled Reserved Instances*: launch within time window you reserve
- *Spot Instances*: short workloads, for cheap, can lose instances
- *Dedicated Instances*: no other customers will share your hardware
- *Dedicated Hosts*: book an entire physical server, control instance placement

EC2 On Demand

- Pay for what you use (billing per second, after the first minute)
- Has the highest cost but no upfront payment
- No long term commitment
- Recommended for short-term and un-interrupted workloads, where you can't predict how the application will behave.

EC2 Reserved Instances

- Up to 75% discount compared to On-demand
- Pay upfront for what you use with long term commitment
- Reservation period can be 1 or 3 years and Reserves a specific instance type
- Recommended for steady state usage applications (think database)
- *Convertible Reserved Instance*
 - can change the EC2 instance type
 - Up to 54% discount
- **Scheduled Reserved Instances**
 - launch within time window you reserve
 - When you require a fraction of day / week / month

EC2 Spot Instances

- Can get a discount of up to 90% compared to On-demand
- You bid a price and get the instance as long as its under the price
- Price varies based on offer and demand
- Spot instances are reclaimed with a 2 minute notification warning when the spot price goes above your bid
- Used for batch jobs, Big Data analysis, or workloads that are resilient to failures.
- Not great for critical jobs or databases

EC2 Dedicated Hosts

- Physical dedicated EC2 server for your use
- Full control of EC2 Instance placement
- Visibility into the underlying sockets / physical cores of the hardware
- Allocated for your account for a 3 year period reservation
- More expensive
- Useful for software that have complicated licensing model (BYOL Bring Your Own License)
- Or for companies that have strong regulatory or compliance needs

EC2 Dedicated Instances

- Instances running on hardware that's dedicated to you
- May share hardware with other instances in same account
- No control over instance placement (can move hardware after Stop / Start)

Which host is right for me?

- On demand: coming and staying in resort whenever we like, we pay the full price
- Reserved: like planning ahead and if we plan to stay for a long time, we may get a good discount.
- Spot instances: the hotel allows people to bid for the empty rooms and the highest bidder keeps the rooms. You can get kicked out at any time
- Dedicated Hosts: We book an entire building of the resort

EC2 Instance Types – Main ones

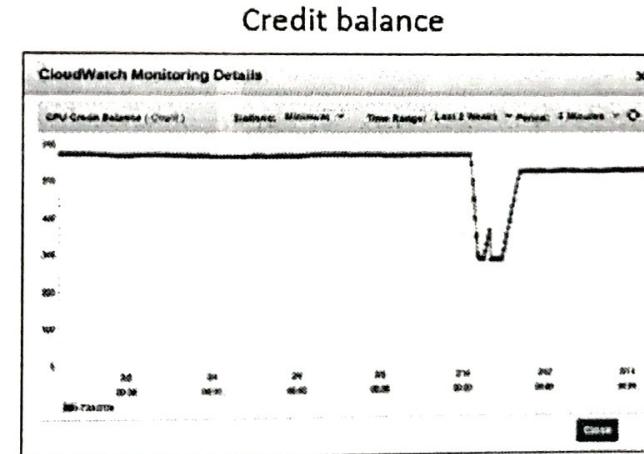
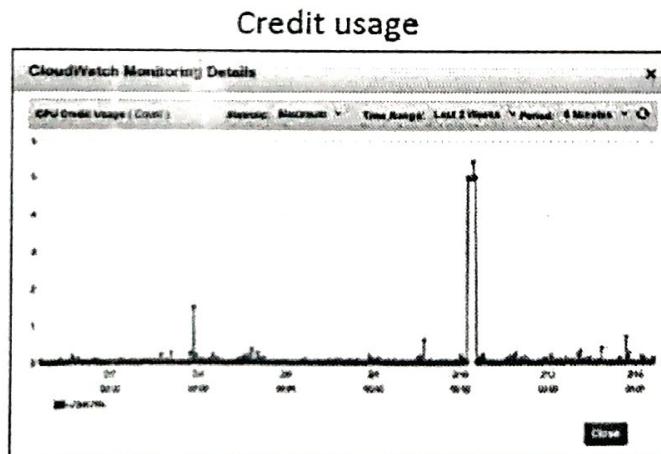
- R: applications that needs a lot of RAM – in-memory caches
- C: applications that needs good CPU – compute / databases
- M: applications that are balanced (think “medium”) – general / web app
- I: applications that need good local I/O (instance storage) – databases
- G: applications that need a GPU – video rendering / machine learning
- T2 / T3: burstable instances (up to a capacity)
- T2 / T3 - unlimited: unlimited burst
- <https://www.ec2instances.info>

Burstable Instances (T2/T3)

- AWS has the concept of burstable instances (T2/T3 machines)
- Burst means that overall, the instance has OK CPU performance.
- When the machine needs to process something unexpected (a spike in load for example), it can burst, and CPU can be VERY good.
- If the machine bursts, it utilizes “burst credits”
- If all the credits are gone, the CPU becomes BAD
- If the machine stops bursting, credits are accumulated over time

Burstable Instances (T2/T3)

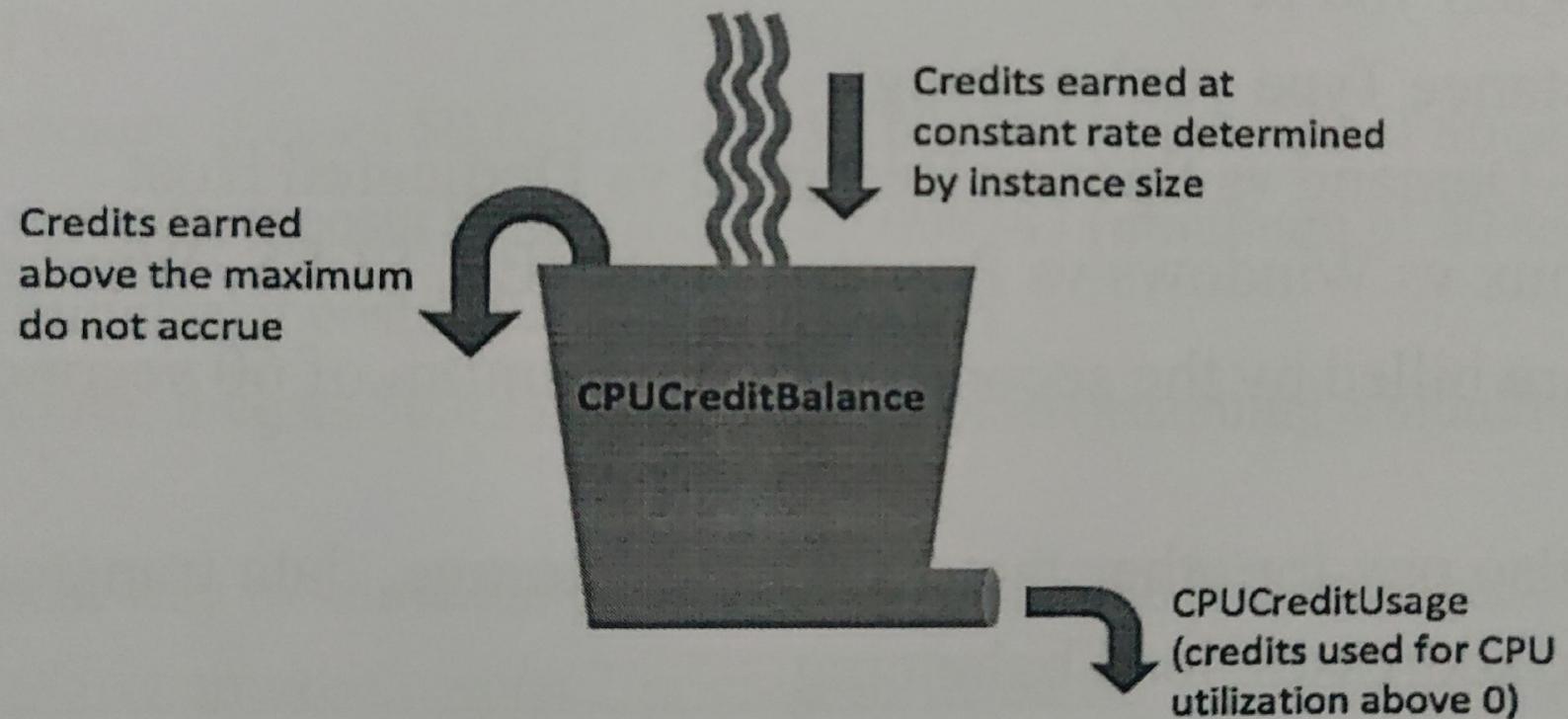
- Burstable instances can be amazing to handle unexpected traffic and getting the insurance that it will be handled correctly
- If your instance consistently runs low on credit, you need to move to a different kind of non-burstable instance



CPU Credits

Instance type	Launch credits	vCPUs	CPU credits earned per hour	Maximum earned CPU credit balance	vCPUs	Baseline performance (% CPU utilization)
t2.nano	30	1	3	72	1	5%
t2.micro	30	1	6	144	1	10%
t2.small	30	1	12	288	1	20%
t2.medium	60	2	24	576	2	40% (of 200% max)*
t2.large	60	2	36	864	2	60% (of 200% max)*
t2.xlarge	120	4	54	1296	4	90% (of 400% max)*
t2.2xlarge	240	8	81	1944	8	135% (of 800% max)*

Burstable Instances (T2/T3)



EC2 Pricing

- EC2 instances prices (per hour) varies based on these parameters:
 - Region you're in
 - Instance Type you're using
 - On-Demand vs Spot vs Reserved vs Dedicated Host
 - Linux vs Windows vs Private OS (RHEL, SLES, Windows SQL)
- You are billed by the second, with a minimum of 60 seconds.
- You also pay for other factors such as storage, data transfer, fixed IP public addresses, load balancing
- You do not pay for the instance if the instance is stopped

EC2 Pricing Example

- t2.small in US-EAST-1 (VIRGINIA), cost \$0.023 per Hour
- If used for:
 - 6 seconds, it costs $\$0.023/60 = \0.000383 (minimum of 60 seconds)
 - 60 seconds, it costs $\$0.023/60 = \0.000383 (minimum of 60 seconds)
 - 30 minutes, it costs $\$0.023/2 = \0.0115
 - 1 month, it costs $\$0.023 * 24 * 30 = \16.56 (assuming a month is 30 days)

The best way to know the pricing is to consult the pricing page:

<https://aws.amazon.com/ec2/pricing/on-demand/>

What's an AMI?

- As we saw, AWS comes with base images such as:
 - Ubuntu
 - Fedora
 - RedHat
 - Windows
 - Etc...
- These images can be customised at runtime using EC2 User data
- But what if we could create our own image, ready to go?
- That's an AMI – an image to use to create our instances
- AMIs can be built for Linux or Windows machines

EC2 Instances Overview

- Instances have 5 distinct characteristics advertised on the website:
 - The RAM (type, amount, generation)
 - The CPU (type, make, frequency, generation, number of cores)
 - The I/O (disk performance, EBS optimisations)
 - The Network (network bandwidth, network latency)
 - The Graphical Processing Unit (GPU)
- It may be daunting to choose the right instance type (there are over 50 of them) - <https://aws.amazon.com/ec2/instance-types/>
- <https://ec2instances.info/> can help with summarizing the types of instances
- R/C/P/G/H/X/I/F/Z/CR are specialised in RAM, CPU, I/O, Network, GPU
- M instance types are balanced

EC2 – Checklist

- Know how to SSH into EC2 (and change .pem file permissions)
- Know how to properly use security groups
- Know the fundamental differences between private vs public vs elastic IP
- Know how to use User Data to customize your instance at boot time
- EC2 instances are billed by the second and can be easily created and thrown away, welcome to the cloud!