



## AWS



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

وَأَنْ لَيْسَ لِلْإِنْسَانِ إِلَّا مَا شَعَرَ (39) وَأَنْ سُبْحَانَهُ سُوْفَ يُرَى (40) ثُمَّ يُجْزَاهُ الْجَزَاءُ الْأَوْفَى (41) وَأَنَّ إِلَيْهِ رُجُوكَ الْمُنْتَهَى (42)



## ليه لازم إحنا كمبرجيون نتعلم الـ Cloud Computing؟

الدنيا Cloud بص يا سيدى، زمان كنا بنكتب الكود ونشغله على جهازنا أو على سيرفر في الشركة. بس دلوقتى اتغيرت، وبقينا نقدر نشغل التطبيقات بتاعتنا من أي مكان في العالم عن طريق الـ

مش رفاهية، دي بقت حاجة أساسية لأي مبرمج حابب يشتغل صح. ليه؟  
علشان:

تقدير **تشغل تطبيقك لأي عدد مستخدمين** بسهولة من غير ما تشيل هم السيرفرات.

- **توفر فلوس** لأنك بتدفع بس على اللي بتستخدمه.

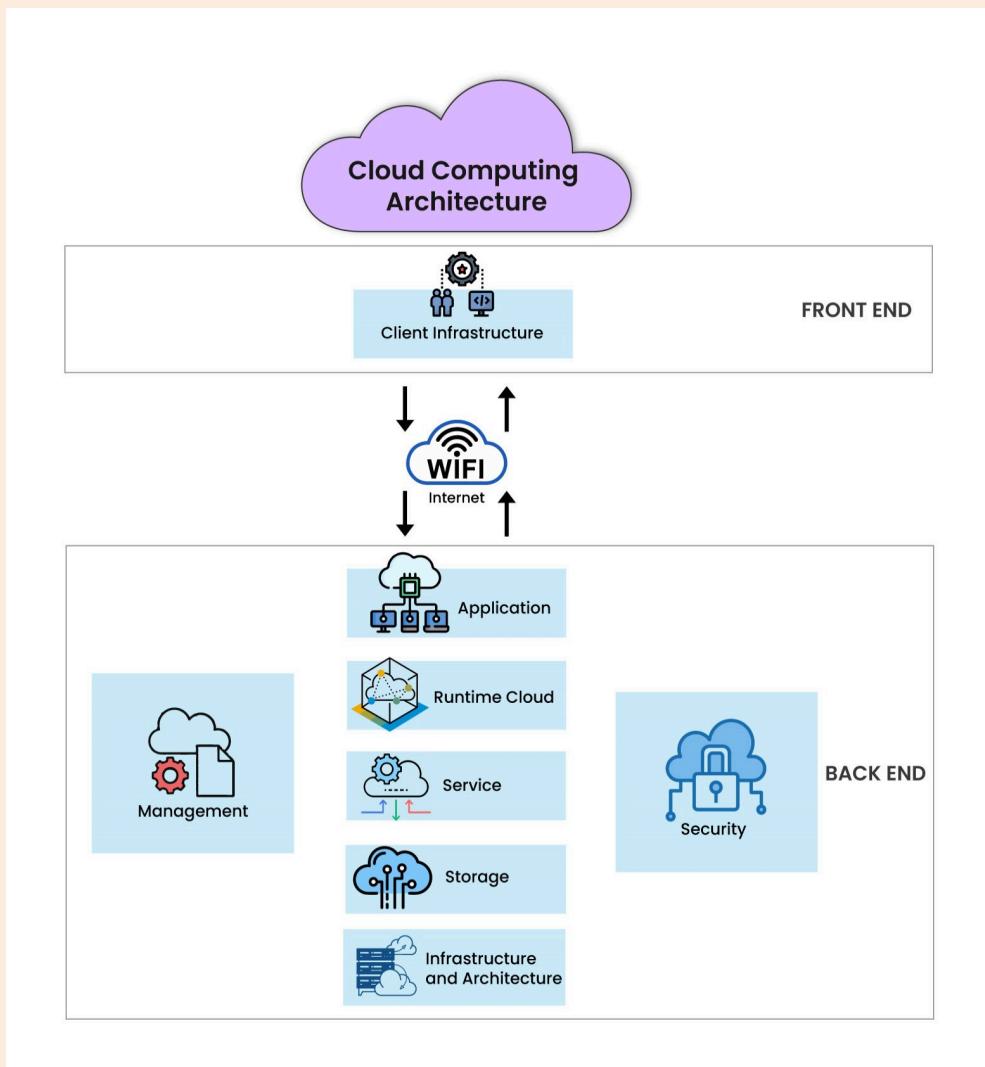
- تقدر **نشر وتحديث تطبيقك بسرعة** من غير خطوات معقدة.

- تلاقي **خدمات جاهزة** تساعدك في حاجات زي قواعد البيانات، التخزين، الحماية... كل ده من غير ما تعمله بيايدك.



## Cloud computing

is the on-demand delivery of compute power, database, storage, applications, and other IT resources via the internet with pay-as-you-go pricing





## Main Benefits of Cloud Computing ?

### 1. Cost Saving

فالطبيعي عشان كنت تعمل للمشروع بتاعك كنت هتحتاج سرفارات وحماية وتربيد مكان وناس تدير المكان وليلة كبيرة جت الـ Cloud قلتلك ريح دماغك من كل دا واجر الي محتاجو ادفع علي قد استهلاكك : and pay as you go

### 2. Scalability & Flexibility

works on-demand :

دلوقتني بقا عندك القدرة تزود الـ Resources زي مانت عاوز وقت مانت عاوز بطريقتين:

1. Horizontal: هتنزود مثلا سيرفر كمان علي الي عندك;

2. Vertical: هتنزور قدرة السيرفر الي عندك مثلا زودلو مساحة او CPU

### 3. Accessibility & Mobility

بقيت قادر تتصل بالـ Resources من اي مكان فقط من خلال الانترنت

### 4. Disaster Recovery & Backup

بقا في حماية اكتر للبيانات بتعتك بدل مكان يبقى عندك الداتا سنتر يحصل في مشكلة وتضيع الدنيا دلوقتني هيككون في حماية اكتر عن طريق ان هيبقى في نسخ من الداتا بتعتك في اماكن مختلفة وبالتالي لو وحدة وقعت الباقى يغطي مكانها وهنشرح الموضوع بالتفصيل قدام

### 5. Automatic Update And Maintenance

يقولك بدل متفضل تعذر وتصلح في الـ Infrastructure ومثلا حجة عطلت تغيرها حاجة أدأها

بقا ضعيف تبدلها وليلة كبيرة الـ Cloud قلذلك السرفارات وكل حاجة عندي انا ههتم بيهها وهمعمل الازم

### 6. High Availability

نظرا لان الـ Environment بتعتعتك ه تكون في اكتر من مكان ه تكون متاحة بنسبة اكتر من لما

تكون في مكان واحد



## Shared Responsibility

### Services of Cloud :

تقليديا كنت بتشغل (Private Cloud) On-Premises كل البنية التحتية  
و بالتالي هيبيقا ليك تحكم في كل حاجة  
مع ظهور الـ Cloud قدمتك 3 طرق للخدمة

#### 1. Infrastructure As a Service

هنا هتأجر بنية تحتية كاملة هيبيقا ليك تحكم في

1. Data
2. Access
3. Application
4. OS

غير كدا هيبيقا مسؤولية الـ Cloud Provider

#### 2. Platform as a Service

كل ما عليك تجاري الـ app and data

هيبيقي ليك تحكم في

Data , Access , Application

#### 3. Software as a Service

هيبيقي ليك تحكم فقط في

The diagram illustrates the shared responsibility matrix for cloud services, comparing On-premises, IaaS, PaaS, and SaaS models across various infrastructure layers.

	On-premises	IaaS (Infrastructure-as-a-Service)	PaaS (Platform-as-a-Service)	SaaS (Software-as-a-Service)
User Access/Identity	Customer Responsibility	Customer Responsibility	Customer Responsibility	Customer Responsibility
Data	Customer Responsibility	Customer Responsibility	Customer Responsibility	Customer Responsibility
Application	Customer Responsibility	Customer Responsibility	Customer Responsibility	Customer Responsibility
Guest OS	Customer Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility
Virtualization	Customer Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility
Network	Customer Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility
Infrastructure	Customer Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility
Physical	Customer Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility	Cloud Service Provider Responsibility



## Region

دي عبارة عن منطقة جغرافيا جواها مجموعة مناطق بنسميتها Data Centers الي هنحط عليها ال Apps الي جواها ال تكون منتشرة في العالم وكل Region في مكان مختلف عن الآخر طب هنخترها بناءا علي اي ؟

### 1. Latency

لو بتعمل app لمستخدمين في افريقيا مترجح تختار ال Region في اووبا لأن دا هيسبب عليه وهو بيحصل بالتطبيق او الموضع

### 2. Services

لان منش كل منطقة فيها ال الي محتاجها لازم تتأكد الاول ان الي محتاج موجود فيها

### 3. Cost

تكلفة الخدمة مش ثابتة في كل منطقة كل مكان ولية سعرو

### 4. Compliance or Data Sovereignty

سياسة الحكومات بتختلف من دولة لآخر لان معنون تكون في حكومة تلك مثل بيانات المستخدمين لازم تكون في نفس ال Region





## Availability Zone (AZ)

عبارة عن مدن متفرقة داخل ال Region Data Centers منفصلين عن بعض في أماكن متفرقة داخل ال Region حيث لو حصل مشكلة في مكان الباقي هيقوم ويعوض المشكلة وبالتالي منتأثرش بأي failure يحصل كل ال AZ ليها power, cooling and networking منفصلة.

### ف دة بيضمنك حاجتين من مميزات ال Cloud

#### 1. Disaster recovery

معني ان لو حصل failure في service بقدر تلقائي استرجعها من AZ تانية

#### 2. Fault tolerance

عملية التبديل التي بتحصل لو حصل مشكلة بتتم تلقاءـي من غير فالمستخدم يحس نهائـي



## EC2 Instance (Elastic Compute Cloud)

عبارة عن web service provided by Amazon Instances بتمكن المستخدم او يأجر Servers

طب مشن دا طب مأكيد هحتاج احدهلو Server RAM,CPU,OS and Storage

عشان كدا قسملك الموضوع لمجموعة :Categories

### 1. General Purposes

ودي بتكون فالوسط يعني ال CPU & RAM متناسبين مفيش حاجة اعلي من الباقي

ودي بتستخدمها ما يكون عندك Small or medium Data base or Dev Environment

Examples : T3 ,T4g ,M5 ,M5g

### 2. Optimized Compute

من اسمها هحتاج اعلي ال CPU زي في ال Media Transcode

Examples : C5 ,C6g

### 3. Optimized Memory

وهنا لما تحتاج اداء عالي فالMemory زي في ال in-Memory Caches / Databases H-Performance

Examples : R5,R6g ,X1 , X1e

### 4. Optimized Storage

لما بنحتاج اداء عالي في ال Storage

like in No SQL Databases or Relational DBs

Examples : I3,D2,H1

### 5. Computing Accelerated

ودا بستخدمو لما اكون عاوزH.W عالي لـ machine learning

Examples : P3,P4,G4,F1





## EC2 Payment Plans

### 1. demand-on

هتدفع عند الحاجة لما تعود خدمة ادفع وشغل

### 2. Reserved

يتأجر مثلاً لمدة سنة او اتنين وكل متزود السنين كل مهتاجد تخفيض اكتر

### 3. Instance Spot

هنا انت بتقول والله انا اخري في تأجير ال service دي وليكن \$50

فاللي هيحصل ان اول متغلي عن السعر الي انت محددو كأقصي حد ليك هتخسر ال Instance

### 4. Savings plans

دي بتستخدمنا لما تكون هتغير ال instance اثناء مانت مستخدمها ودا نوعين

- Compute Saving plan

T3 ⇒ R5 وهنا انت بتغير من Family to another بمعنى انك ممكن تغير من

- EC2 instance Saving plan

Foamily like T3 ⇒ T4g هنا هتغير جوه نفس ال

### 5. Dedicated Hosts

فاللي فات انت كنت بتاخذ ال VM على Server وممكن حد تاني يأخذ VM على نفس ال Server ولكن هنا انت بتاخذ ال Server كلو تهويست عليه

### 6. Dedicated Instance

زي الي فات بس مبيكنش ليه تحكم بال W.H



## EBS(Block Storage )

عبارة عن Network Drive بيتصل بالEC2 ذي لاما توصل هارد خارجي في جهازك كذا another EC2 واحد وتحتى لو الـ EC2 اتمسح تقدر تربط بـ EBS instance وبيبقا خاص بـ EC2 instance can has many EBSs

لو عاوز ان لما تدزف الـ EC2 يتحذف الـ EBS يبقى لازم تفعل الـ AZ to another AZ Backup from EBS تنقلها من طب لو عاوز تعمل

EBS Snapshot حاجة اسمها هتسخدم

هناخد الداتا من الـ EBS الي انت عوزها وتعملها Snapshot in EBS snapshot

تقوم بـ restore a Snapshot EBS to another EC2





### EBS Volume Types :

1. GP2/GP3 (SSD) : General Purpose يعني حاجة فالمعتسط
2. IO1,IO2 (SSD) : Low Latency , High Performance , High throughput

3. STL (HHD) low cost ,big data تستخدمو لو عندك

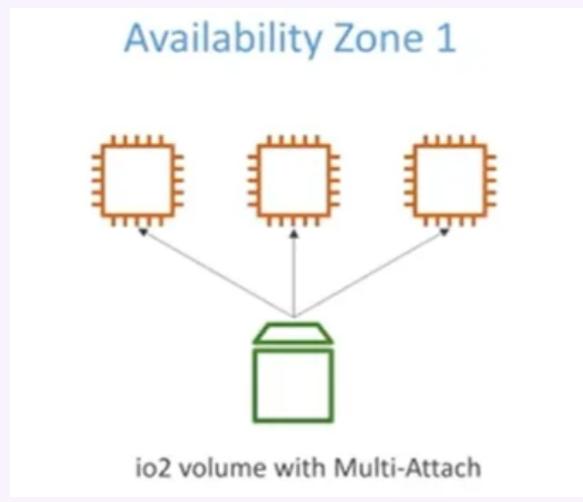
High throughput ومش فارق معاك التأثير بس عاوز

4. SC(HHD) with big data ⇒ low cost

وميفرقش معاك بردو اللتأخير وبت access data مش ديمى يعني كل مدة

**attached to multi EC2 Instances يكون EBS لـ عاوز الـ**

اول حاجة لازم يكون من نوع IO1,IO2 وتفعل الـ





## EFS (File Sharing )

دا عباره عن نظام ملفات هيكون مشترك بين اكتر من EC2  
معكن يكونو في نفس ال AZs او في مختلفة pay per use  
طبع هو اغلي ويتدفع علي قد استخدامك Linux instance فقط  
طبعا دا نظام ملفات يبقى شغال مع Configurations هتببدأ تحطلو ال

### 1. Performance Mode

- General Purpose (Balance) or
- High I/O for Higher throughput

### 2. throughput mode (for Networking Downloading or speed)

- Bursting : High traffic in EFS لو عندك
- Provisioning : هنا انت هتقلو والله انا عاوز سرعة ال traffic تكون قد كدا
- Elastic : Automatic الموضع هنا انت هتديلي

### 3. Storage Tier : هنا انت هتحدد هل الملفات الي عندك هيحمل علية دينما ولی كل فترة

- Standard : for Frequency Access
- Infrequency Access :with low price

### 4. Availability :

- Standard : AZ عوزها متاحة في اكتر من for production
- One Zone : هنا عوزها في واحدة بس لو وقعت خلاص for Dev



## Compare Between EBS & EFS

Feature	EBS (Elastic Block Store)	EFS (Elastic File System)
Type of Storage	Block storage	File storage
Use Case	Attached to <b>one EC2 instance</b> (like a hard disk)	Shared between <b>multiple EC2 instances</b> (like a network drive)
Performance	High performance for single instance	High concurrency, good for parallel access
Mounting	Mounted to <b>one EC2 instance</b> at a time (except multi-attach for some types)	Can be mounted by <b>many EC2 instances at once</b>
Access Protocol	Uses <b>block-level access</b>	Uses <b>NFS protocol</b>
Data Persistence	Persistent, survives instance termination	Also persistent and durable
Scalability	You must choose size in advance	Automatically scales as you add/remove files
Pricing Model	Pay per GB provisioned + IOPS	Pay per GB used
Backup	Supports snapshots (manually or automated)	Automatically backed up by AWS
Latency	Lower latency (good for databases, OS)	Higher latency than EBS (better for shared access)
Best For	Databases, boot volumes, apps needing fast I/O	Web servers, shared file storage, big data analytics



## S3

هي خدمة تخزين بتقدر تخزن الملفات في شكل Objects بحيث

1. توسع غير محدود عن طريق انك تقدر تحط ملفات زي مانت عاوز بدون حد اقصي
2. تتضمن توافر عالي للبيانات وقت مانت عوزها
3. بتقدم سرعة وصول للبيانات في اي وقت

وكمان مفید في تخزين ال Static Files like (HTML,CSS,JS)

BucketS3 بببقي جواه مجلد افتراضي رئيسي بتخزن في ملفاتك وهو ال

بحيث ان اسمو لازم يكون فربد من نوعو مش متشابه مع غيرهو وميحتويش غير علي حروف صغيرة

وعشان توصل لكل Object لازم يكون ليه path start with S3://(S3 name ) /File

كمان لكل object ببغيت تقدر تشوف اي تعديلات سابقة هتعملها



## S3 Security

عندك مسارين اما يعمل على Security users

1. Use user base policy: IAM من خلال اال

2. Resource based policy : External users لما يكون عندك

becket Police **فهنهعمل**

يبقى عشان افتح الاتصال بالS3

ندخل على ال becket ونختار PERMISIONS ونلغي تفعيل ال

2. Create packet police use auto generated JSON

وتخددها وتطبّقها في مكانها وتعمل حفظ

## Example

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor    **JSON**    Import managed policy

```
1  {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "Stmt1671091081730",
6       "Action": "s3:*",
7       "Effect": "Allow",
8       "Resource": [
9         "arn:aws:s3:::honsondemobucket",
10        "arn:aws:s3:::honsondemobucket/*"
11      ]
12    }
13  ]
```

Security: 0   Errors: 0   Warnings: 0   Suggestions: 0

Character count: 356 of 2,048  
The current character count includes character for all inline policies in the user: hanson

Cancel    **Review policy**



## How can create a static web in S3?

بعد معرفة الملف هتغير الخصائص بتعتو وتفعلو ك static website

ثم هتحدد ال Index.html & Error.html

وتحفظو



## Classes Storage

من خلالهما هتحدد ال Availability of files

هنا انت هتاخذ اعلي ائحة للملفات باسرع اداء : 1. Standard(Default)

اقل سرعة من الي قبلها وبتستخدمها لو مش هتعوز الملفات كثير : 2. Infrequent Access

وطبعاً اقل تكلفة

نفس الي قبلها بس شغاله علي One AZ : 3. One Zone availability Access

4. Glacier ( archiving , Retrieval & Pay for Storage ) :

دا ارخص نوع وبتستخدموا لو مش هتعمل Access كثير وليه نوعين

الملف هيكون موجود عالاقل 90 يوم : 1. Instant Retrieval

اقل سرعة وبيوفر انواع مختلفة زي : 2. Flexible Retrieval

- Expedited 1 Minute

- Standard 3-5 Hours

- Bulk 5-12 Hours

دا ابطأ نوع والملفات بتبقى موجودة عالقل 180 يوم : 3. Deep Archive

- Standard 12 Hours

- Bulk 48 Hours

اقل تكلفة من ال 3. (RRS) Reduced Redundancy Storage : Infrequent Access

بس بيبقا فيه مخاطر اكبر

4. Intelligent Tiering : هنا يتم التحكم بشكل تلقائي من AWS



## S3 Versions

دي يتمكنك ترجع للنسخ القديمة الي عملت عليها تعديل او انك تسترجع ملفات حذفتها لأن  
لما بتحذف فعليلا لا يتم الحذف ولكن ينتقل الملف الي ال Delete Marker  
فلو جيت حذفت الملف من ال Delete Marker هتسترجع الملف تاني  
قبل متفعل ال Versioning بيبقى ب null ID



## S3 Replication

تقدير تعمل نسخ لـ Bucket بطريقتين

1. Cross-Region-Replication : to another Region

2. Same-Region-Replication : Region في نفس ال

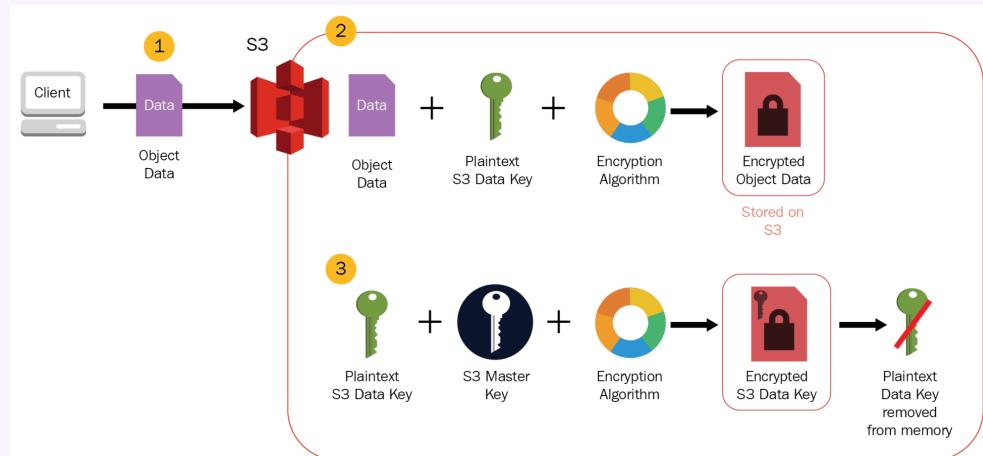
Management of Becket ودا بتتعددو من ال



## Encryption

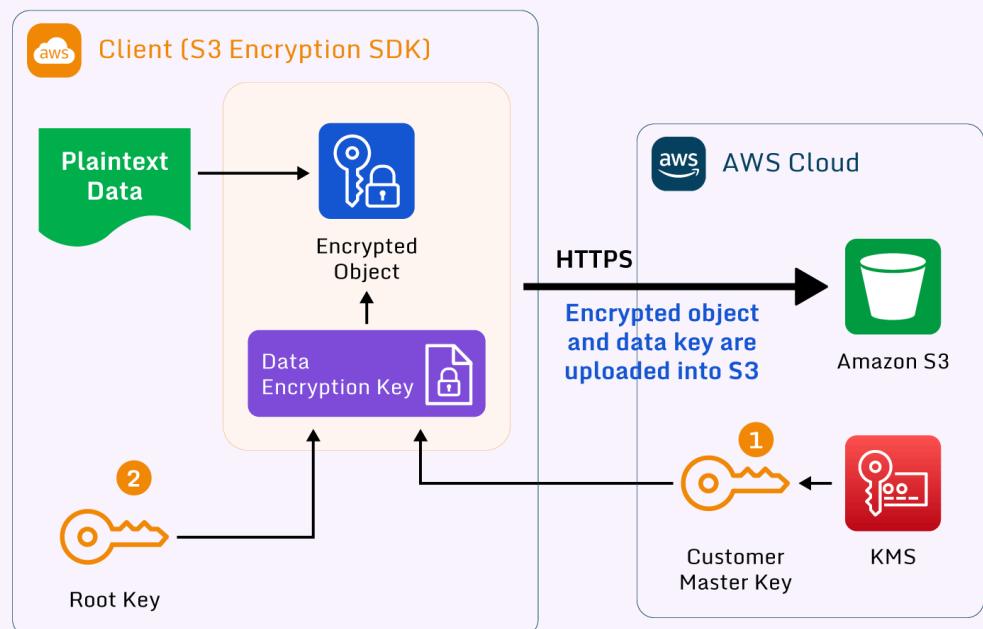
: عندك طريقتين

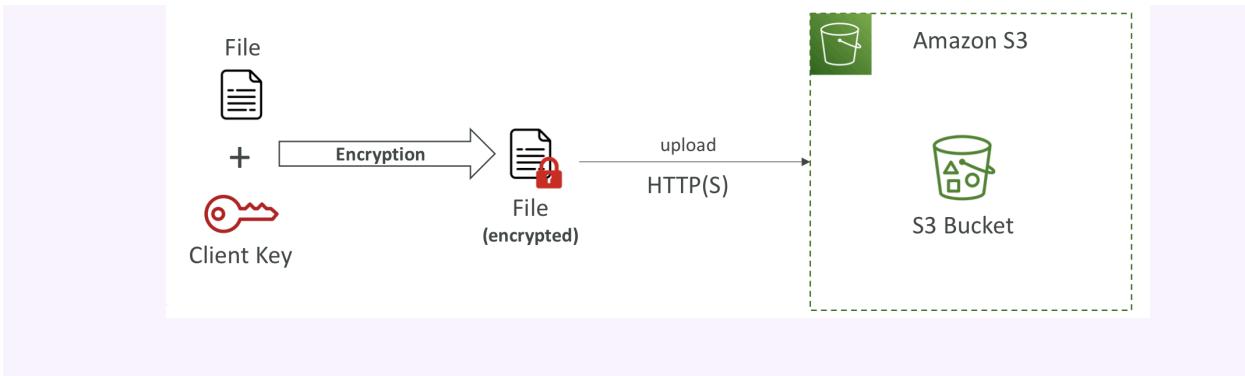
1. Server Side Encryption : يتم التشفير عند السرفر بـ S3 Encryption Key



1. Client Side Encryption : يتم التشفير قبل رفع الملف من خلال ان ال

S3 Encryption Key هيبيقا عدو ميرفع على Client







## IAM(Identity Access Management)

من خلالها تقدر ت Create Users or Groups

بحيث كل واحد بيقي عارف يقدر يعمل اي و ميقدرش يعمل اي

عن طريق انك بتعمل Policy اما تديها لمستخدم او تطبقها علي جروب فكل الي فيه يخدوها

User can be in one Group or Multi Groups and inherit All Policies

Or User Can be without Group

### Example of Policy

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor    [JSON](#)    Import managed policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "Stmt1671091081730",
6       "Action": "s3:*",
7       "Effect": "Allow",
8       "Resource": [
9         "arn:aws:s3:::honsondemobucket",
10        "arn:aws:s3:::honsondemobucket/*"
11      ]
12    }
13  ]
14 }
```

Security: 0   Errors: 0   Warnings: 0   Suggestions: 0

Character count: 356 of 2,048.  
The current character count includes character for all inline policies in the user: hanson.

[Cancel](#)   [Review policy](#)



## Multi Factor Authentication (MFA)

### يعني إيه؟

بدل ما تدخل كلمة السر فقط، بتضيف خطوة تانية (أو أكثر) علشان تتأكد إنك فعلاً صاحب الحساب.

### المكونات الأساسية:

بيعتمد على 3 أنواع من العوامل، وغالباً يتطلب اتنين منهם

1.  **حاجة تعرفها**: زي كلمة السر أو رقم سري.
2. **حاجة تمتلكها**: زي موبايلك، تطبيق Google Authenticator، أو رسالة SMS.
3. **حاجة تمثلك**: زي بصمة صباعك، بصمة الوجه، أو صوتك.

### مثال بسيط:

لما تدخل على حسابك:

1. تكتب **كلمة السر**
2. **يوصلك كود على الموبايل** أو تستخدم **بصمة صباعك**.

### لية نستخدم MFA؟

- لأن لو حد سرق الباسورد بتاعتكم، مش هيقدر يدخل من غير العامل الثاني.
- بيقلل فرص الاختراق بنسبة كبيرة جداً.



## How can Access AWS ?

عندك 3 طرق لدخول:

1.  **AWS Console**: تدخل عليه بحسابك (واجهة رسومية) موقع AWS.
  2. **AWS CLI**: تكتب أوامر في التيرمنال/الكوماند ليني.
  3. **AWS SDKs**: للتعامل مع Python أو Java زى AWS.
- كل طريقة ليها استخدام حسب احتياجك.



## IAM Roles

من خلالها هتقدر تتحكم في اي الي الـ Service تقدر تعملو

تخيل إن عندك:

- EC2 Instance
- S3 وعايز تخليه يدخل على

(وده خطير) بدل ما تديه **username g password**,

بتقوله:

خد ال Role دي فيها اذن انك تدخل وتقل تعمل كذا وكذا وهتنتهي بعد وقت كذا



## عشان تعمل On Premise Database بتحتاج يكون عندك

- Physical Storage
- Operating System
- Storage Capacity
- High Availability
- Backup

جت AWS قالت هريحك وهوفرلك اللي انا هديرو:

- Operating System
- High Availability
- Auto Storage Scaling
- Automatic Backup

Deference Databases هنتعرف على

1

## Amazon RDS

For Relational Database Service

Can query in it with SQL

**Can be:**

- MySQL
- PostgreSQL
- SQL Server
- Oracle
- MariaDB
- Aurora

**طب انا هستفاد اي لما استخدمنا بدلا من ال EC2 ؟**

1. Can Do Vertical or Horizontal Sailing
2. تقدر تحططها في اكتر من AZ
3. Can do Backup & Restore at Specific Time  
يعني ممكن تقلو عاوز استرجع الداتا من الساعة كذا لكانها
4. Monitoring Database in Dashboard

**بس الي متقدرش تعملو انك تعمل SSH**

لان دا مش فعلي زي EC2 هو خدمة مدارة من AWS

يعني AWS بتدبر السيرفر اللي فيه قاعدة البيانات



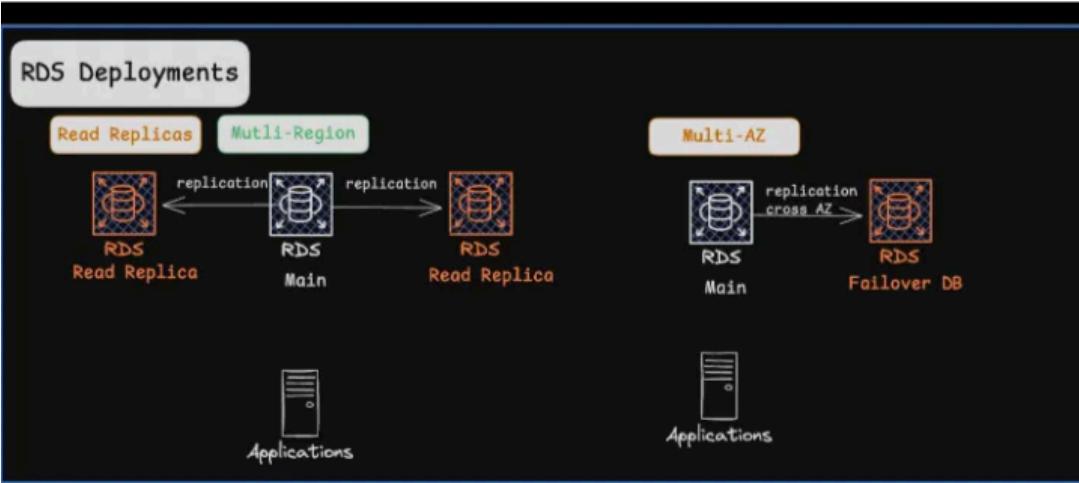
## RDS Deployment

### 1. Read Replica

عن طريق انك بتأخذ نسخة من الاداتا الي في ال Main RDS وتحط في Read Replica ويعمل عليها عمليات القراءة فقط

### 2. Multi AZ

عن طريق انك هتعمل Backup in Another AZ بحيث ان لو الاساسية وقعت تقدر تشغيل ال Backup



2

## Aurora

Supported for Postgrad & MySQL Only (Not Open Source)

هي زي ال RDS بس اسرع بكثير منها

MYSQL IN Aurora

اسرع 5 مرات من ال RDS

Postgrad IN Aurora

اسرع 3 مرات منو في ال RDS

Storage Scaling up to 128 TB

طبعا اغلبي من ال RDS بـ 20%

في نوع منها اسمها Aurora Server Less مميزاته

Auto-Scaling based on usage and you pay per sec

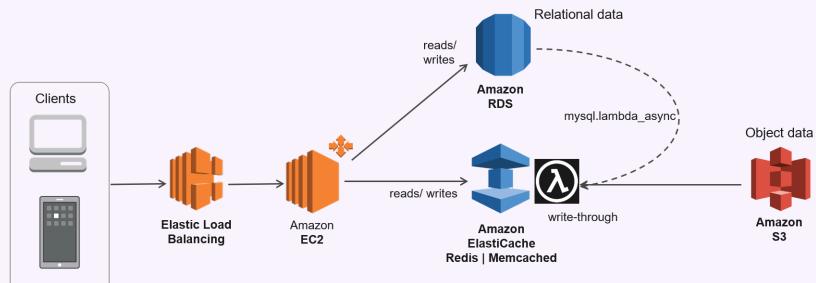
3

## Elastic Cache

Redis or In memory DB Cache ىگىچ Cache ئىچ ئۆلىك

AWS manage OS, Monitoring , Backup

### Real-Time Apps and Caching



4

## DynamoDB

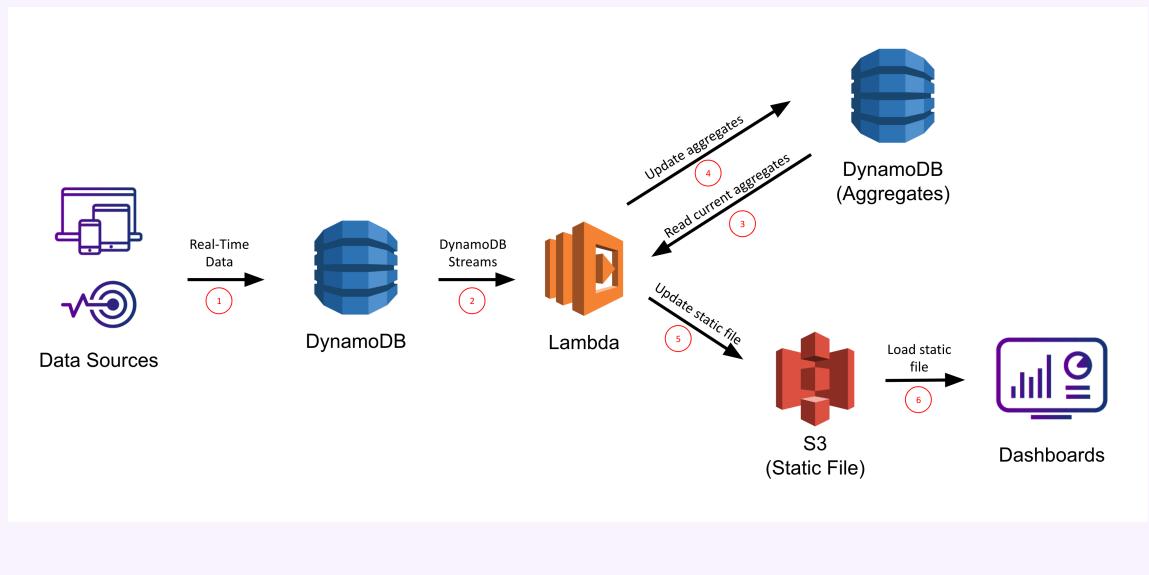
NO SQL DB based on Key & Value دی جا صة ب

High Available With 3 Replicas in 3 AZ

works Serverless : Configurations من غير متصل اي

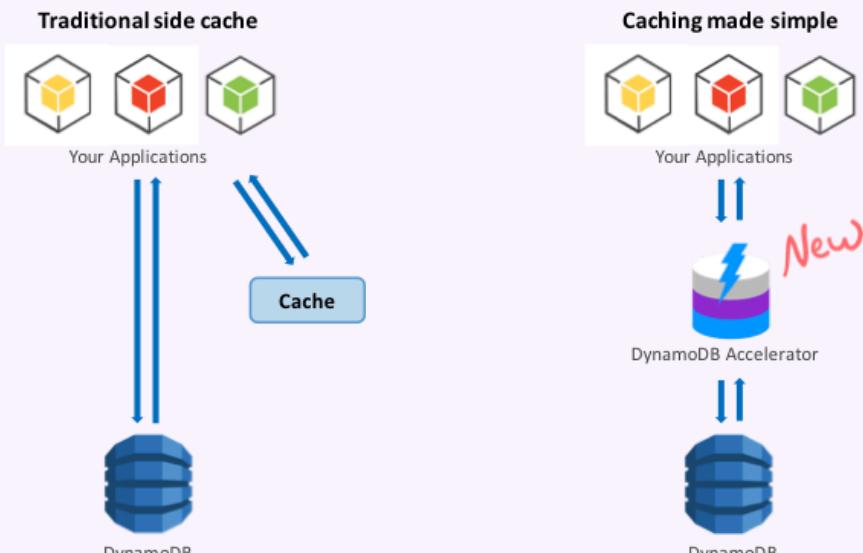
that can handle millions request per sec

and low Latency



## 5 DynamoDB Accelerator(DAX)

ودي الCache الي بتشتغل مع ال DynamoDB  
بتحسن الاداء اكتر بـ 10 مرات



## 6 Documentation DB

Like Arora but it works with MongoDB only

NO SQL وشغالة

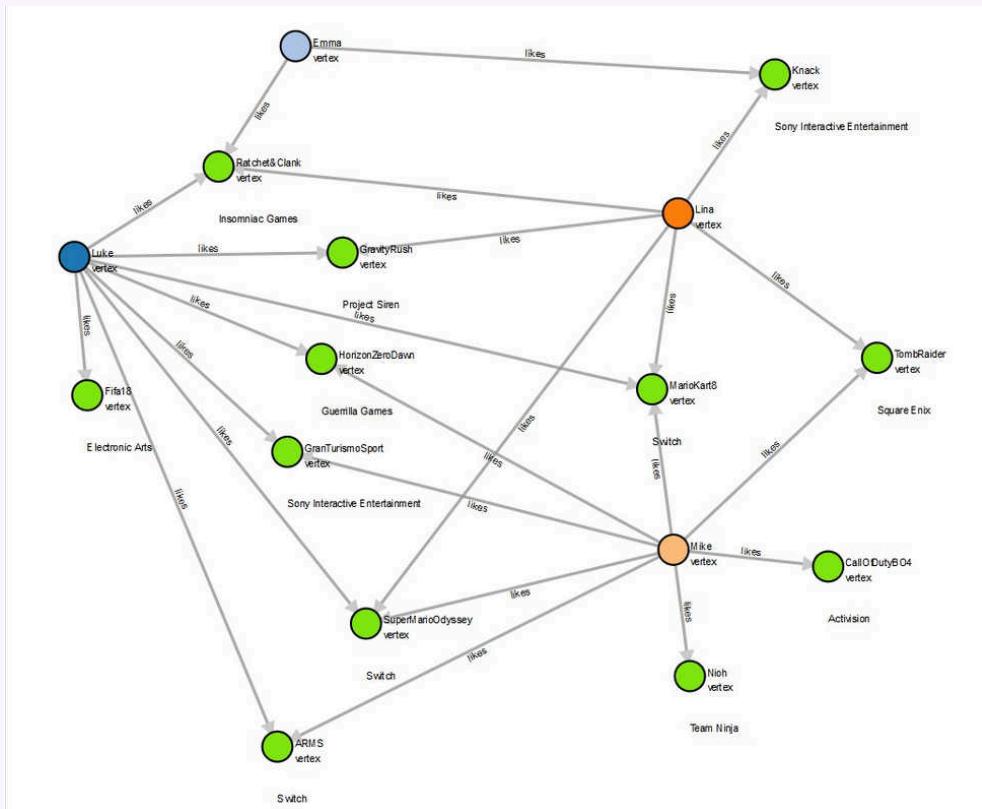
High Available With 3 Replicas in 3 AZ

7

## Neptune

## Graphs بال خاص

High Available With 3 Replicas in 3 AZ with 15 Read Replicas

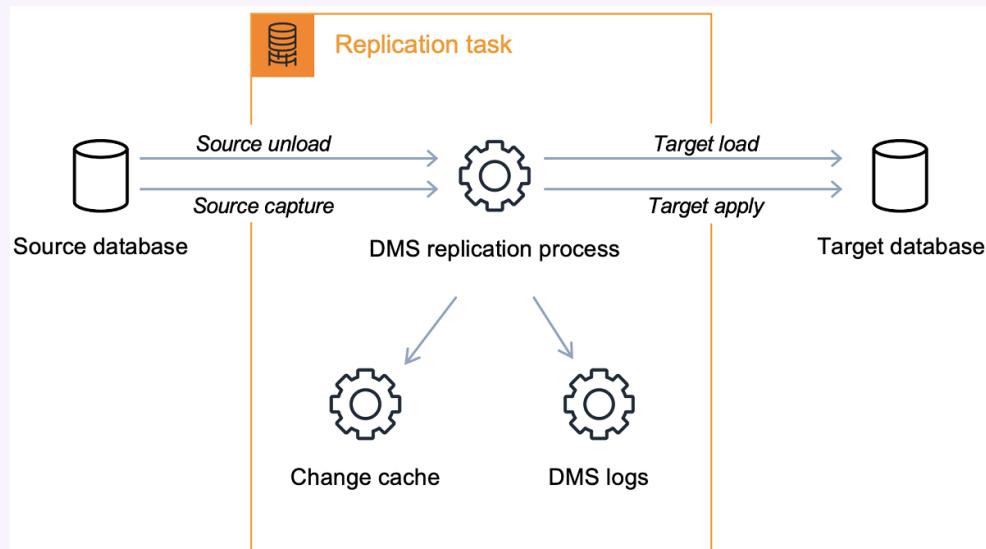


8

## DMS

دی بستگی کو لاما تيجي تعامل

Extract data from DB to DB يعني ت



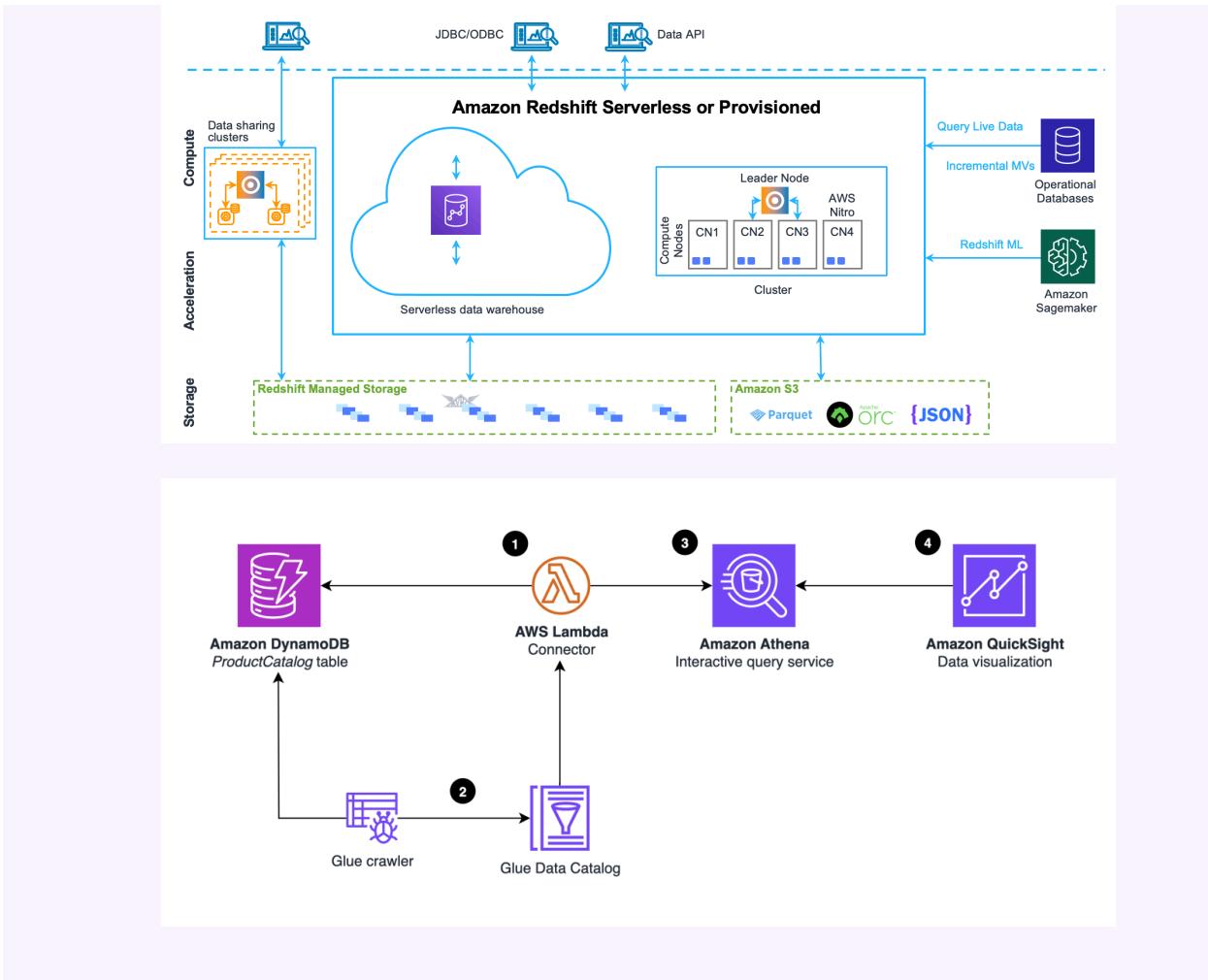


## Another Databases

الخدمة	الوظيفة الأساسية	الاستخدامات الشائعة	الميزة الرئيسية	متى تستخدمها
<b>Redshift</b>	تحليل البيانات (OLAP) مبني على (الخدمة PostgreSQL)	تقارير - لوحات تحكم - التحليلات الفورية	أداء عالٍ لتحليل البيانات - تكاملٌ - Serverless مع أدوات التحليل	عندما تحتاج تحليل كميات كبيرة من البيانات بسرعة وكفاءة خاصة بالتقارير أو BI
<b>EMR</b>	Hadoop إنشاء عنقود لتحليل ومعالجة البيانات	المعالجة والتحليل الضم - تعلم الآلة	قابلية التوسع - إدارة مرنة للعناقيد	إذا كنت تحتاج إلى بيئة تحليل بيانات ضخمة مثل Hadoop أو Spark
<b>Athena</b>	خدمة تحليل بيانات Serverless مباشرة من S3 ملفات CSV, JSON, إلخ. على S3	تحليل بيانات من S3	لا حاجة لبنية تحتية - SQL يدعم	عندما يكون لديك بيانات وتحتاج إلى S3 على تحليلها بدون إعداد خوادم
<b>QuickSight</b>	إنشاء لوحات تحكم تفاعلية وتحليلات بصرية للبيانات	تحليل أعمال - تقارير بيانية متقدمة	تكامل مباشر مع Redshift, Athena, RDS - واجهة رسومية سهلة	عند الحاجة إلى عرض مرئي وتحليلات تفاعلية للبيانات
<b>Glue</b>	استخراج، ELT خدمة (تحميل، تحويل)	تجهيز البيانات للتحليل - تنظيف البيانات	ذكي - - يربط بين مصادر البيانات المختلفة	عندما تكون البيانات غير منظمة وتحتاج تجهيزها أو تحويلها قبل تحليلها
<b>Kinesis</b>	معالجة بيانات بث مباشر (Streaming)	التحليلات الفورية - تتبع الأحداث أو السجلات لحظياً	ingestion سرعة في - كبير دعم الحجم	عندما تحتاج إلى تحليل بيانات لحظية مثل Clickstream أو Log Streams

### خلاصة سريعة:

- للتحليلات المعقدة على قواعد بيانات كبيرة **Redshift** استخدم.
- إذا كنت ترغب مرتبة كاملة في بيئة تحليل بيانات مثل **EMR** أو Hadoop أو Spark.
- بدون إعدادات معقدة S3 إذا كنت تحتاج تحليل بيانات مخزنة على **Athena** استخدم.
- وتحليلات بصرية **QuickSight** لعمل Dashboards.
- لتحويل وتجهيز البيانات قبل التحليل **Glue** استخدم.
- للتحليل اللحظي للبيانات المتداقة **Kinesis** استخدم.





## Load Balancer

هو جهاز أو برنامج يقوم بتوزيع حركة المزور (الطلبات) على مجموعة من الخوادم (السييرفرات) على الشأن.

- يمنع الضغط على سيرفر واحد.
- يحتسب الأداء والسرعة.
- يضمن استمرارية الخدمة حتى لو حصل عطل في سيرفر معين.

**Types of Load Balancer :**

1

## Application Load Balancer

1. works is layer 7 (Application layer) http بالذاتية
2. load balanced to multi Http Applications Called Target Groups

كل APPS or (Servers) يحتوي على مجموعة Target Group

3. Routing to Difference Target Groups based on

- URI like `example.com /target _Group_Name`
- Host Name like one.example.com

يبيعت يتطممن علي السرفرات الي عندو هل هي شغاله ام لا فببیعut

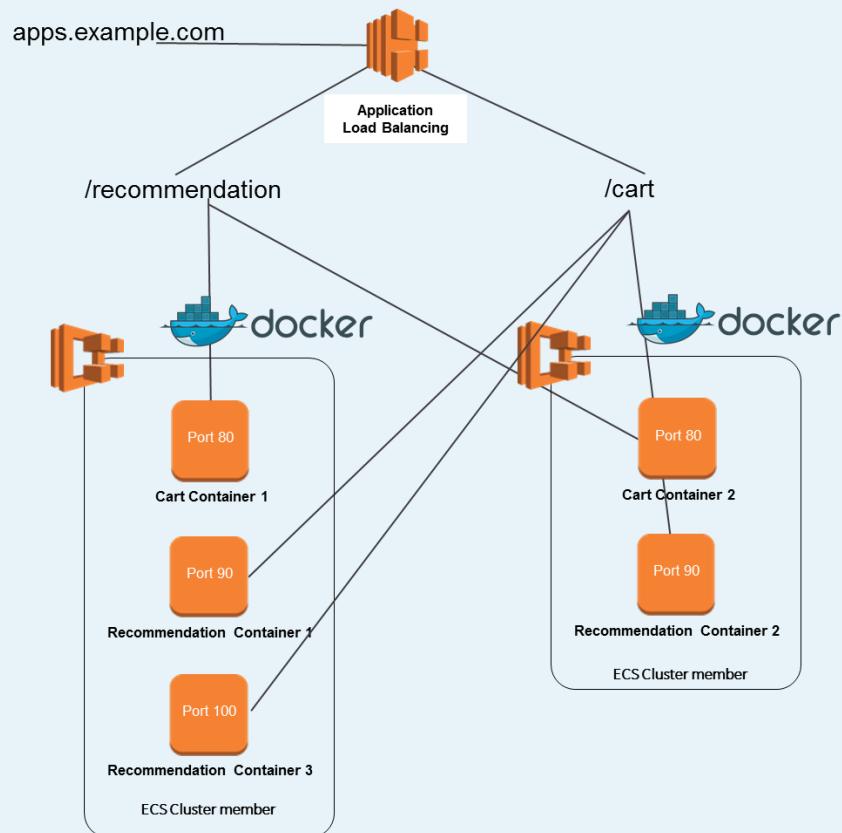
وبالتالي لو لقي واحد عطلان يطلعون من الـ Target Group

4. Health Check

هو الي هبيقي متضاف عند السرفر الي رايحلو

طلب هتجيب الـ IP of User منين لما تحتاجو

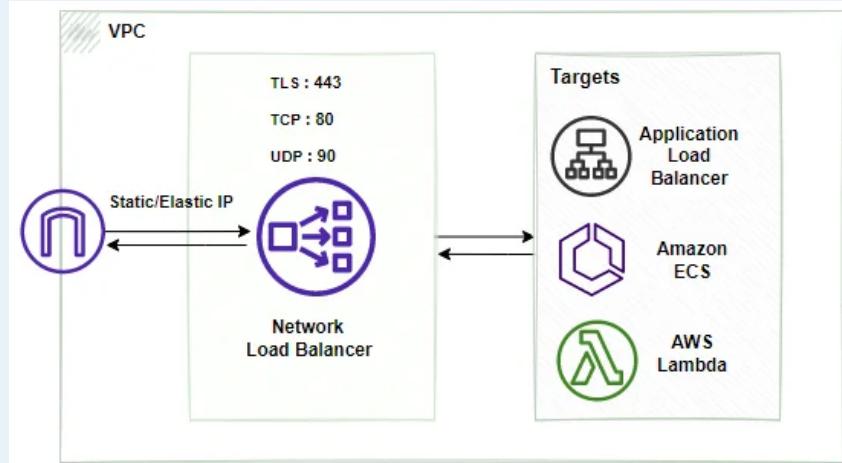
هتبطو في جزء من الـ 3 HTTP IN X-Forwarding in Layer 3



## 2

## Network Load Balancer

يبتغل في ال Application Load Balancer من ال Latency ميزتو او اقل بمقدار 300ms  
IP of user will be static per AZ  
ودا هيفيدك في اعداد ال Fire Wall



3

## Gateway Load Balancer

IDS / IPS و Fire Walls إلزامي Virtual Appliances

works at layer 3

تحليل إن عندك شركة ويستخدم:

- Palo Alto من نوع Firewall
- وعالي كل الترافيك اللي داخل من الإنترن特 يعدي على الفايروول الأول.

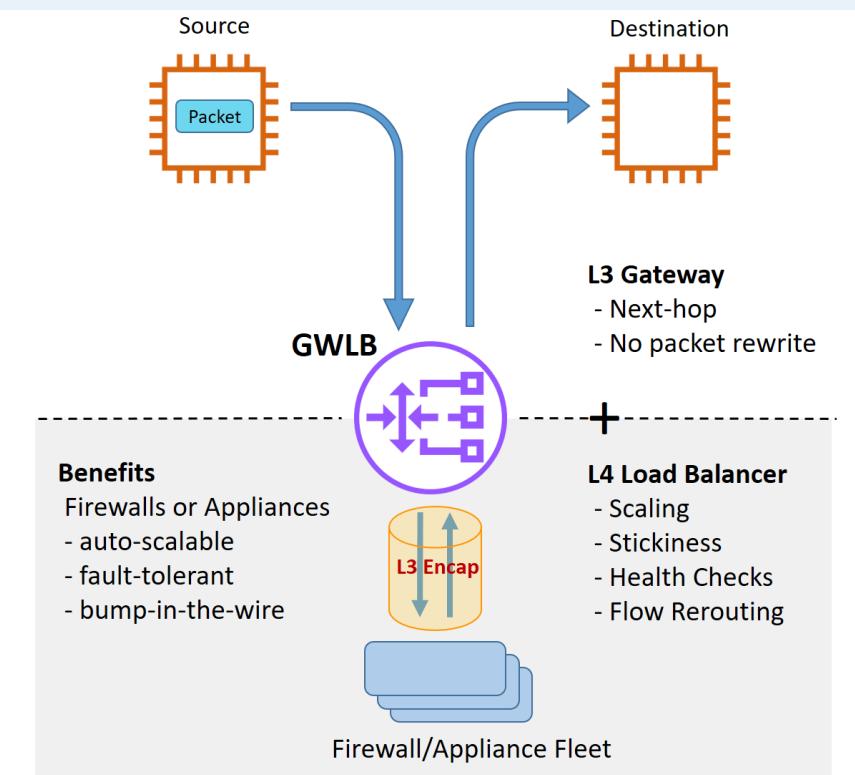
يستخدم GWLB:

1. الترافيك يصل لـ GWLB.
2. يبعثه للفايروول.
3. الفايروول يراجع الحزمة ويرجعها.
4. لو تعام، تروح للسيفر.

### الفوائد الحقيقة:

بقدر توسيع عدد أجهزة الفايروول زي ما يتسع السيرفرات.

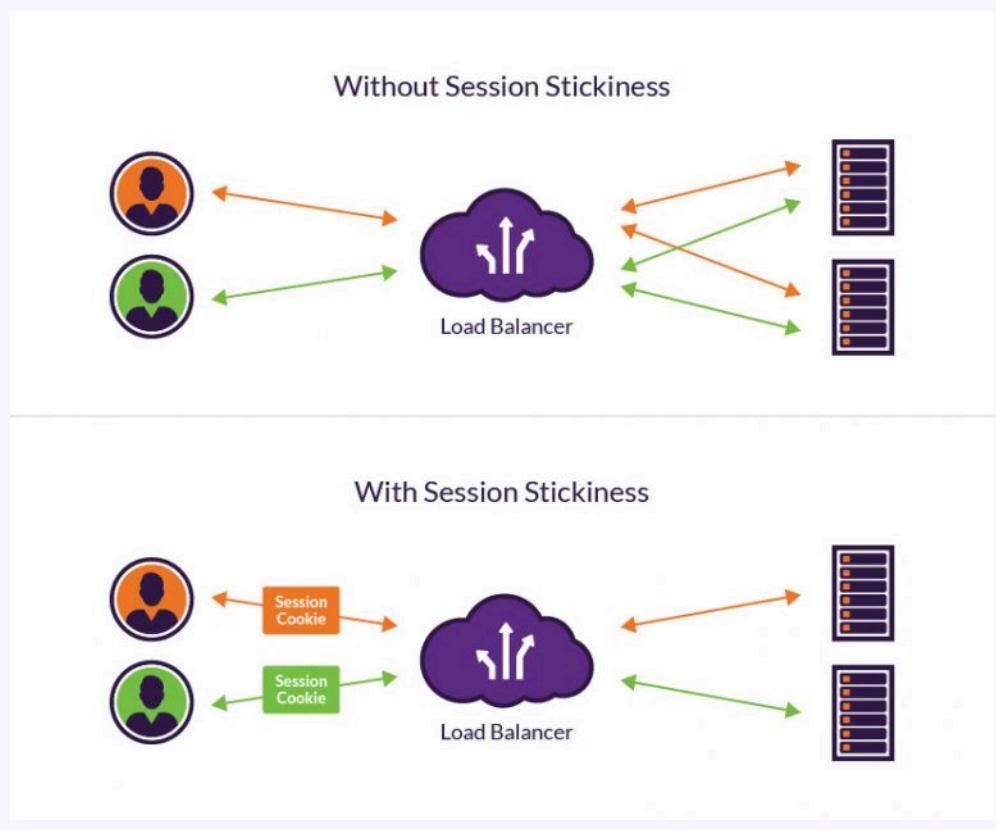
- Highly available.
- Static IP.
- كتير ومشحتاج تغيير routing tables.





## Sticky Session

دلوقي في اكتر من Load Balancer وكل واحد فيه اكتر من Target group request بطبعتو بيوزع الحمل عشان ميصلش ضغط علي حاجة طب افرض انت بعت لسرفر معين وعاوزو اون يفضل فكرك وتكلمو باقى طلباتك؟ هنا بتفعل الـ Sticky Session فمساعتها طول مانت معاك الـ Cookie هو هيفضل فاكرك وتبعنلو هو





## SSL Certificates

فكرتها انها بتشفير البيانات بين ال Client and Load Balancer

Certificate : Public key and Private key عبارة عن

بتدي ال Public key to client وتقولو لما تيجي تبعتلي حاجة شفرها بال key دا

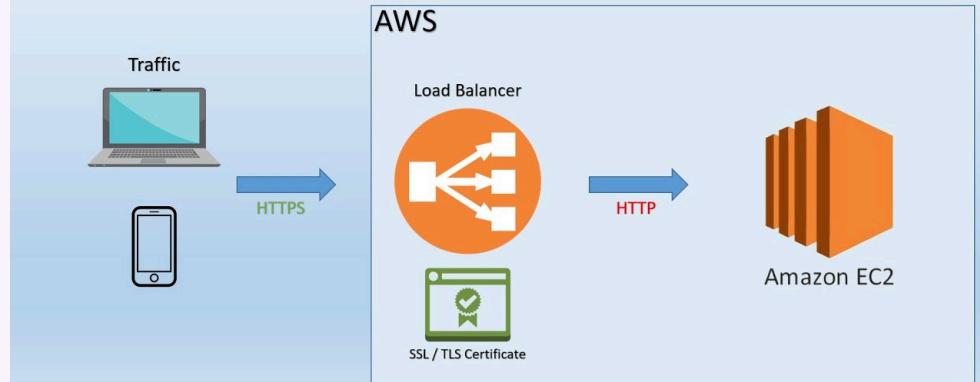
وال Private key هيبيقي عندو Load Balancer هيفك بيته التشفير

دي خدمة في امازون مهمتها انها تدير الحوار دا : ACM

هيخللي السرفر يقدر يصل على ال Certificate المناسب بناء على ال Host Name

يعني ادين ال Host Name وانا هرجعلك ال Certificate المناسب ليك الي هتشفر بيها

## AWS Certificate Manager Tutorial (Get Free SSL/TLS Certificates)





## Auto Scaling Group

: يحل مشكلة زيادة الحمل او قلة الحمل تلائي بمعنى

Scale out : add new EC2 Instance , لما الضغط يزيد,

و لو لقي الضغط قليل وممكن نقل ونوفر , Scale in : Remove EC2 Instance,

ودا اعتمادا على ال alarms

زي مثلا ان لو استهلاك ال CPU وصل لحد كذا زود ولو قل قلل

الموضوع دا مجاني بس انت بتدفع علي الي بيزيدي

طب الموضوع دا بيتم بناء علي اي؟

بتبدأ تحدد شوية اعدادات زي

1. Min : هنا بتحدد مثلا اقل مساحة تكون 50 جيجا

2. Desir : هنا بتحدد المتوسط مثلا يكون 70

3. Max : هنا بتحدد اقصي حاجة ممكن توصلها ومتزدش عن كذا مثلا 100

## Dynamic Scaling Policy

انت هتعمل scale in ,scale out على بناء

1. Target tracking : EC2 Instances لكل ال CPU 85% (Min ,max ,Desir)

لو حصل تغير ابدا بقى ارجع لل اعدادات الي حطتها وشوف المناسب

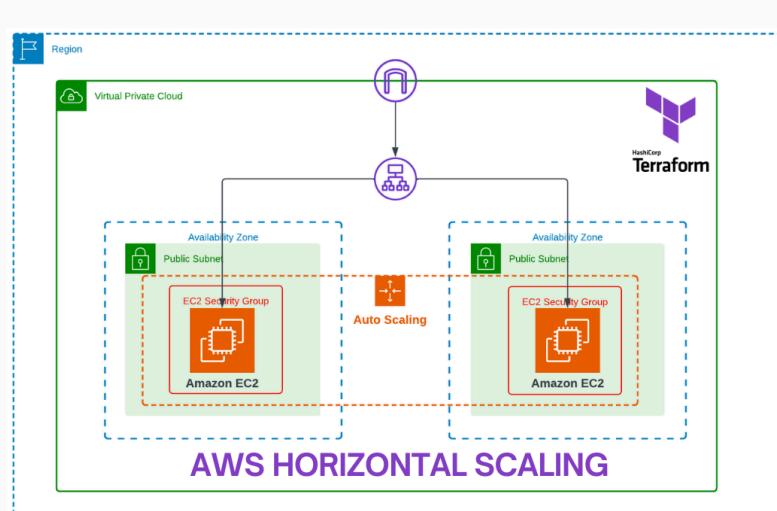
2. Simple /Step : هتقلو ان لما يجي alarm زودلي مثلا 2

وهنا انت بتتحددلو بنفسك هترزود قد اي مش بتسيبو يزود برحتو

3. Scheduled : ودي حاجة مهمه لانك هنا بتحدد مثلا ان في يوم كذا او تاريخ كذا عاوز ازود :

عدد قد كذا خلينا نقول مثلا لو موقع نتيجة الثانوية العامة فهتقول الضغط هيزيد يوم كذا زودلي

400 مثلا EC2







## Networking

### 1 VPC (Virtual Private Cloud)

عبارة عن شبكة خاصة بـ AWS داخلها تقدر تحط ال Services المختلفة  
Partitioning Range of IPs ودي بنستخدمه Subnets عشان نعمل  
of VPC  
Every Subnet in deference AZ  
وكمان الـ VPC يحتوي على Routing Tables and Network Gateways  
Internet Gateway الي هيحصل ان عشان Instance اتطلع على الانترنت بتروح علي

### 2 Routing Table

كل VPC بيبيقا فيها Main Routing Table اللي كل الـ subnets بيتم توجها من خلاله  
وتقدر انت كمان تعمل Costume Routing Table وتحدد فيه الـ Subnets اللي عوزويتوجة  
ودا بيتم اما تعمل

1. Routing Table for Public Subnet

Public IP to Internet Gateway الـ هتوجة وسعتها

2. Routing Table for Private Subnet

Private IP to Nat Gateway الـ هتوجة وسعتها

### 3 Types of Subnets

1. Public Subnet

ودي بتطلع انتernet بال IP Public بتعها عن طريق الـ IGW

2. Private Subnet

ودي متقدرش تطلع انتernet بال IP Private عشان كدا لازم تروح لـ IGW ومنو تجيب الـ IP او توجة NAT GW لـ

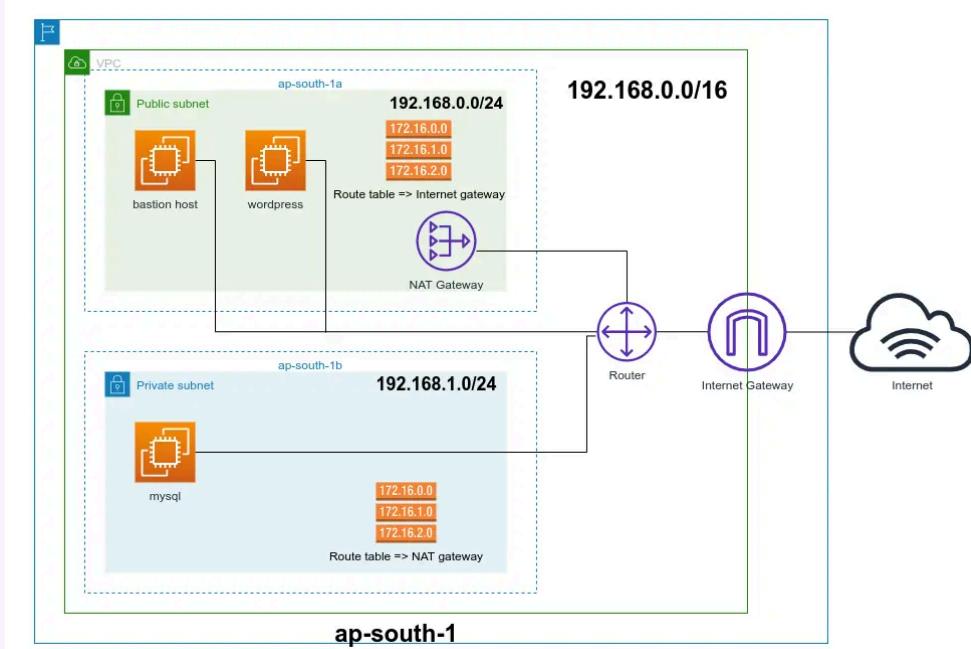
### 4 Internet Gateway (IGW)

من خلاله تقد الـ VPCInstance of VPC الـ تطلع انتernet

5

## Nat Gateway (NAT GW)

من خلاله بنحو الـ Private Subnet نديها IP addresses في الـ Public IP منو تقدر تروح لـ Internet يطلعها انترنت





## Simple Queue Services (SQS)

is a Server less queue to decouple applications

يعني بتفصل الـ apps عن بعضها بحيث ان هجيب زي صندوق بريد احاط فيه الرسائل الي هتتبع  
فهيبيقا عندى

Producers : send message

Consumer : Receive message

يقدر يستقبل رسائل بعدد كبير في الثانية الواحدة :

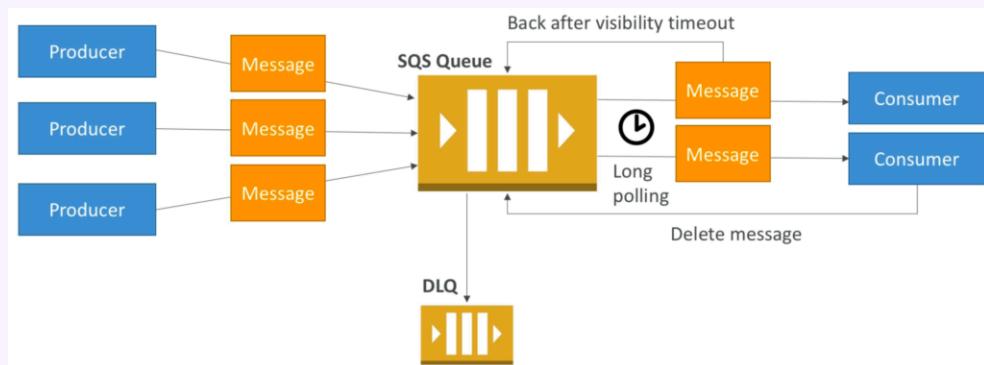
messages in SQS by default still 4 days and maximum 14 day

اول مال Consumer يقرأ الرسالة بتبدأ تدخل في Visibility Time out يعني بتختفي

عن طريق انه بتبقى محدد مثلا عاوز انها تختفي 30 ثانية من بعد متشفها

وال Consumer بيبقى ليه القدرة برد او يحذفها لو هي تخضو بعد ميشفها

لان الـ SQS مش هيحذفها لوحده من غير مال Consumer يحذفها هو





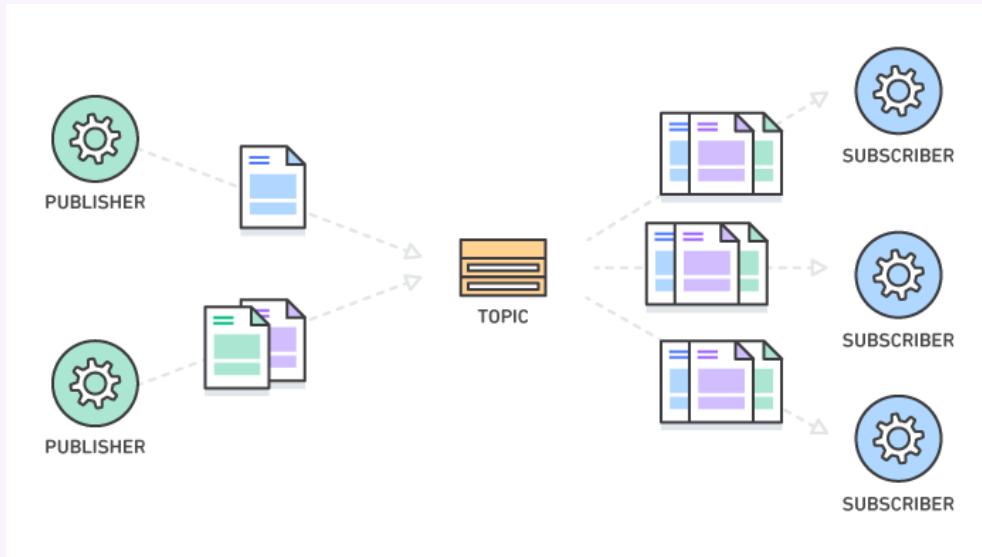
## Simple Notification Service (SNS) (Pub /Sub)

فكرتها انك وانت تتبع رسالة تتبعي Publisher

وبتتبعتها في Topic اول متصل فيه تلقائي تتبع لكل ال Subscribers

مثلاً ممكن تخلي ال S3 وتحتفظ Configurations Publisher as a S3 في فيها ان اول مرتبط فيه ملفات

يتبع على Topic ومنه تتبع رسالة لكل الذي عملين Subscribe

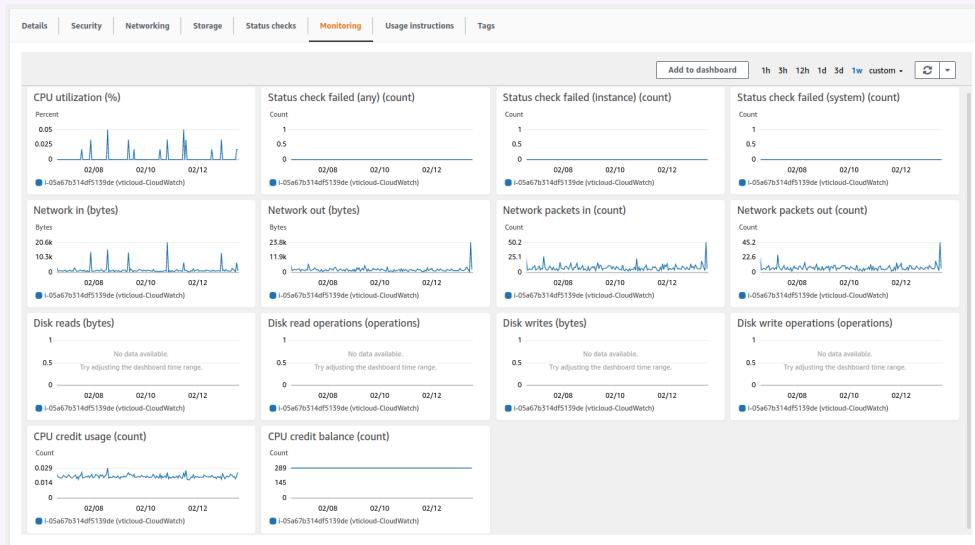




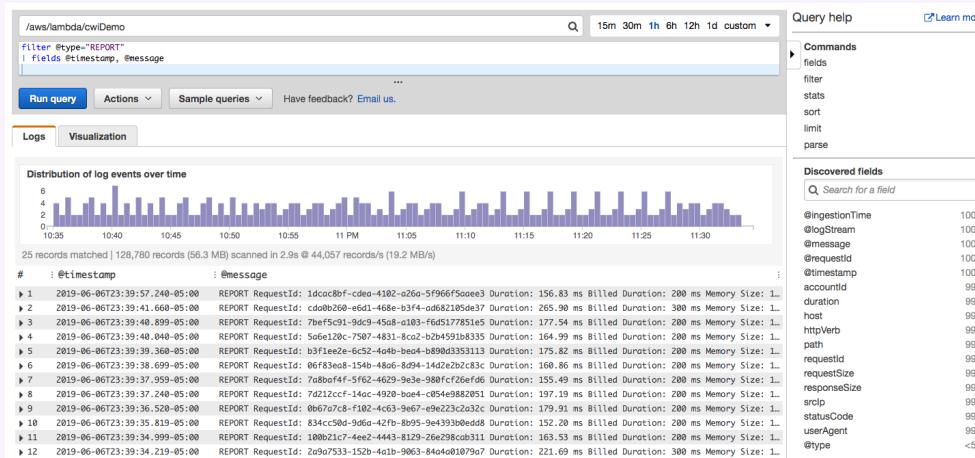
## Cloud Watch

من خاللها بتقدر تجمع لك الLogs اللي حصلت عنك على ال Services  
بتجمع ال Matrices من خاللها تقدر تتبع ال Performance

### Matrices like



### Logs Like



### Alarm

ودا من خاللها بتقدر تقول مثلا هيدحصل حاجه معينه

Based on Matrices

مثلا ممكن تقول لو ال CPU زاد عن 90%

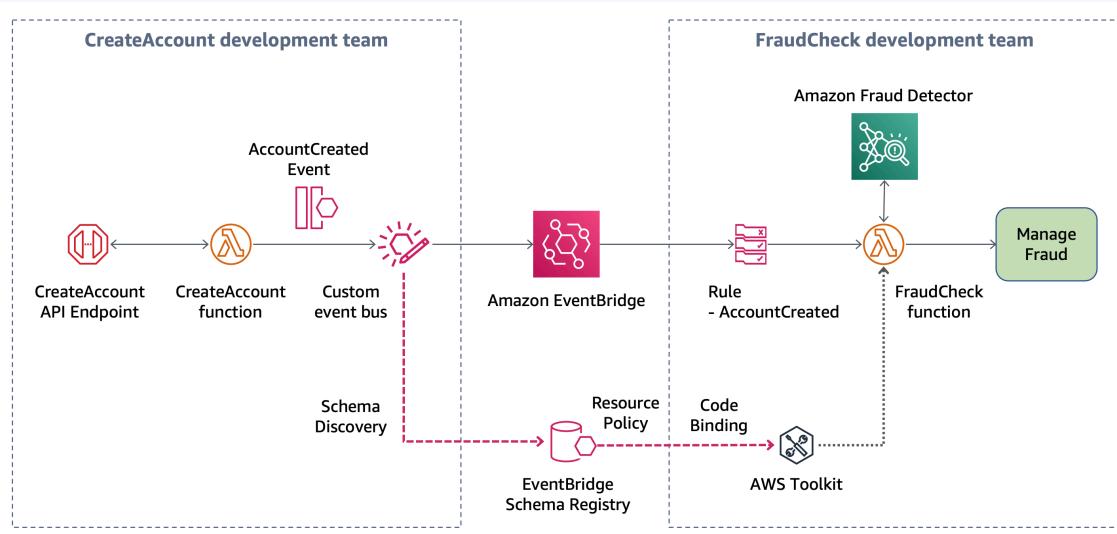
بناء على ال Matrices اللي هيجي سعتها ابعتلي Email معين

او اعملطي لحاجة معينه برتلك بقا



## Event Brage

من خلالها تقدر تعمل مهام محوولة بحيث مثلاً تقول في يوم كذا اعمل حاجة معينه او انك تستنبي ان حدث معين يحصل على ال Service وبعدها تعمل حاجه معينه زي مثلاً تستنبي ان ال S3 يتوقف وبعدها تبعه مثلاً ايميل او مثلاً انك تربطها بال IAM بحيث مثلاً تعوز يتبعه ايميل لما حد يعمل sign in





## Cloud Trail

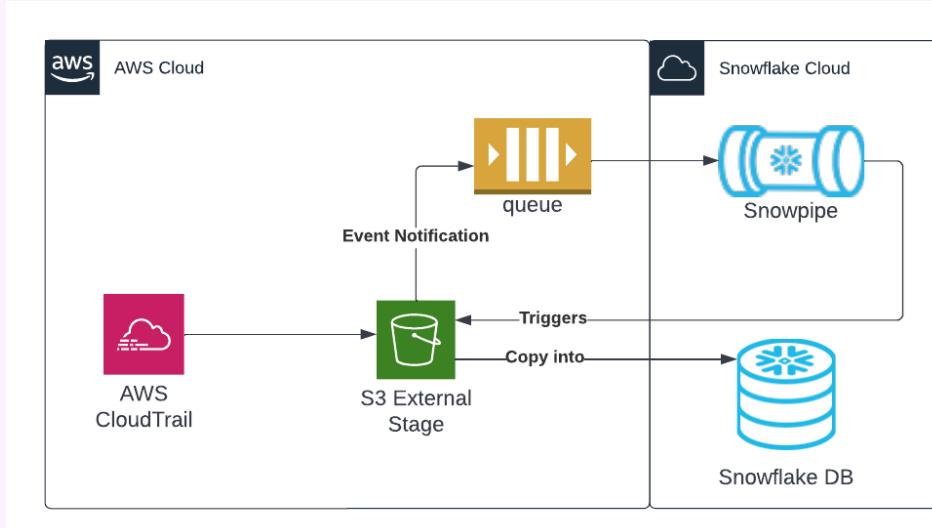
مهمتها إنها تجمع الـ Loges التي حصلت على الـ Services

بحيث ان لو حصل مشكلة في اي منهم تقدر تعرف مين الي عمل المشكلة وبتفضل لمدة 90 يوم

لو عوزها تفضل اكتر من كدا بترميها مثلًا في S3

دا هييفيدك في ان لو حصل وحد حذف طبيق عارف هو مين

زي سلة المهملات كدا



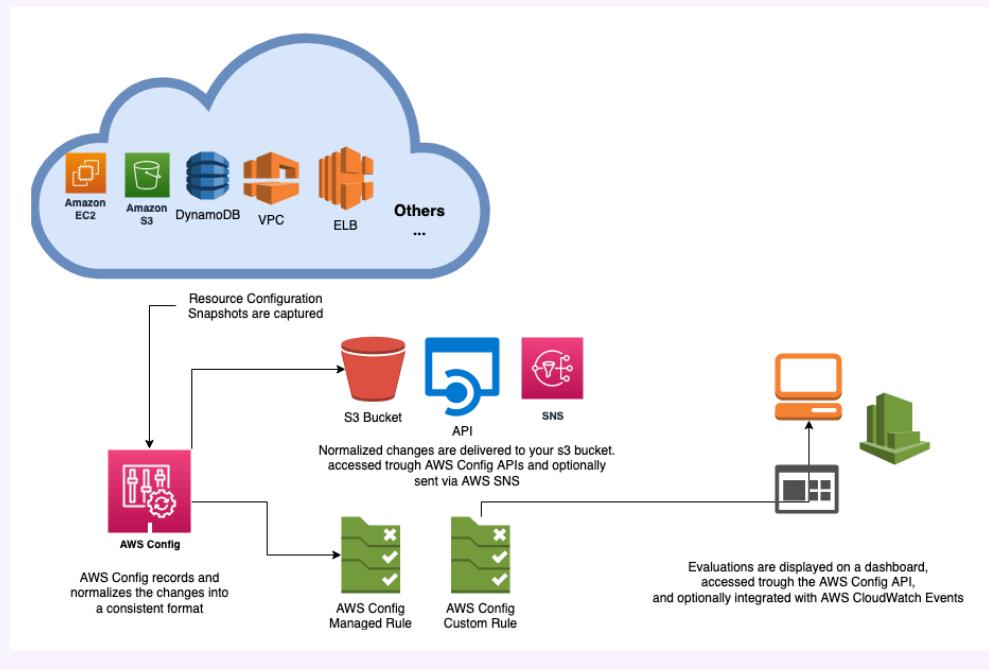


## AWS Config

من خلاله تقدر تعمل Monitoring على الـ Services  
بحيث انك تبحث فيها وتقول مثلاً تقول في اي وقت لو لقيتني عملت

S3 as a Public access

نفذه /alert معينه





## Lambda

it is a service allow you to create a function with many programming language like C++ , Java ,python ,...

it is server less

Executed 15 minute

scaled automatic

بديث تستعمل عدد كبير من ال requests

وبتدفع ازاي بناءا علي اي ؟

بتدفع علي حسب ال requests and compute time

integrated with many services like s3 ,DynamoDB, Cloud watch ,SNS , ...



## API GW

it is a service allow you to create ,Publish APIs automatically

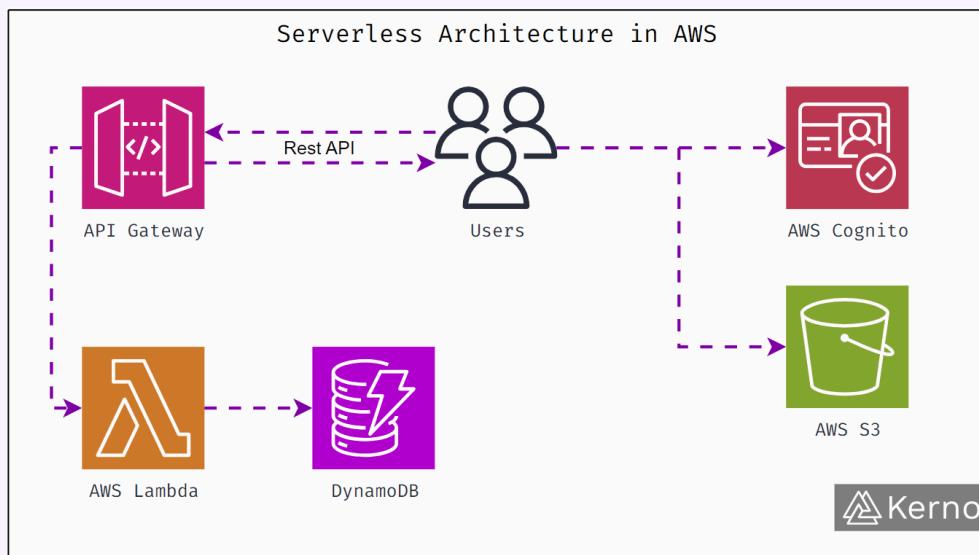
like HTTP ,REST , WebSocket

Server Less and Run OnDemand

Integrated with lambda , s3 ,DynamoDB ,....

يسعدك انك تقدر تعمل

من خلال انك مثلا تدي لنك للمستخدمين يحصلو منو علي بعض البيانات من الـDB





## ECR (Elastic Container Register )

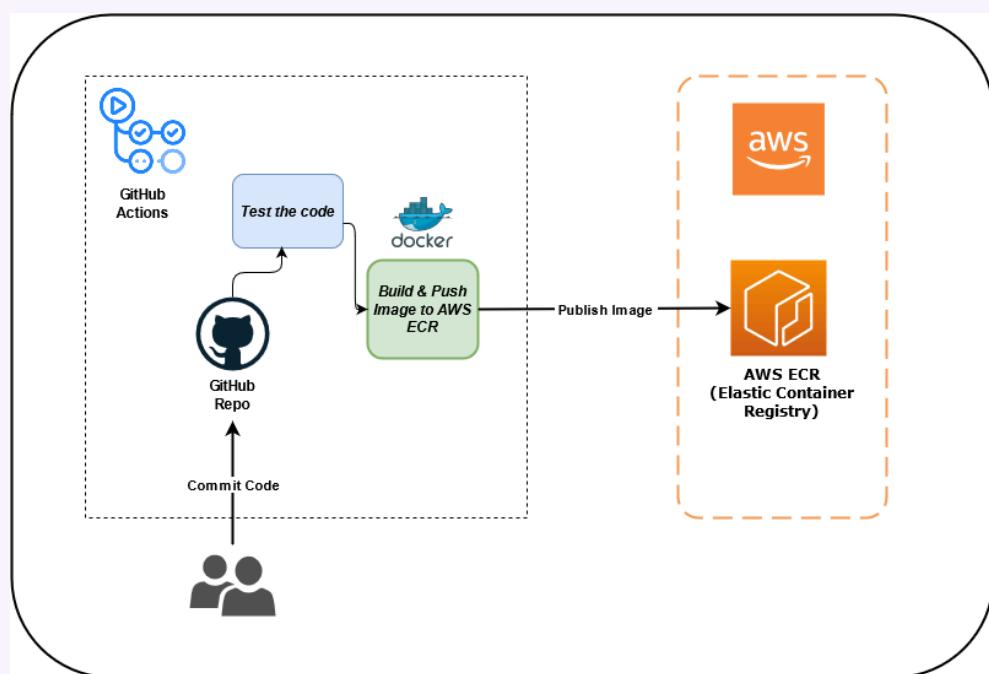
this service used to store and mange docker image

can be Private / Public Repository

**Docker Image :**

هي قالب جاهز يحتوي على كل حاجة محتاجها البرنامج علشان يشتغل  
الكود + المكتبات + الأدوات + الإعدادات.

يعني زي صورة طبق الأصل من البرنامج والبيئة اللي بيشتغل فيها.



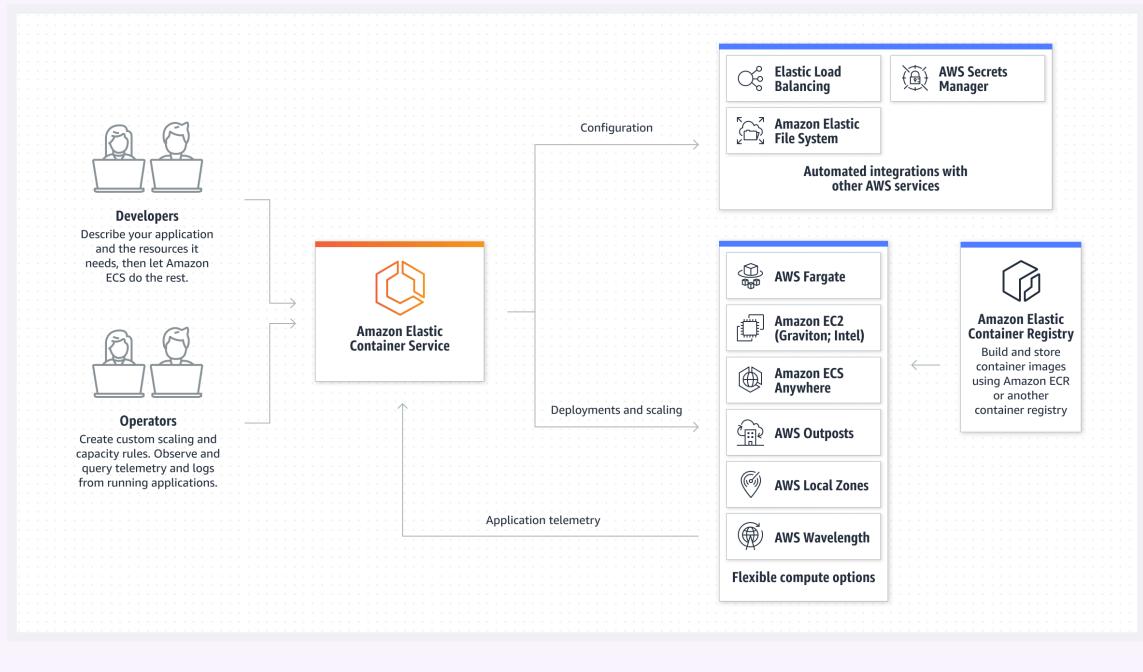


## ECS (Elastic Container Service )

use to launch containers in AWS

container will run in EC2 Instance

Each EC2 run in ECS Cluster

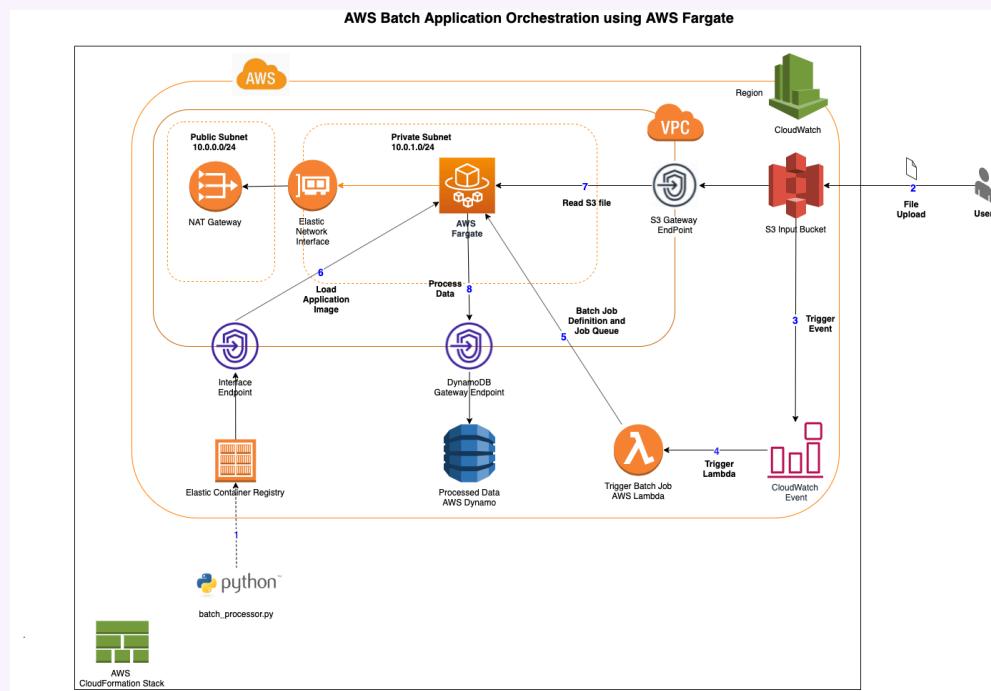




## Fargate

نفس الـ ECS

but works server less based on CPU & Ram you need

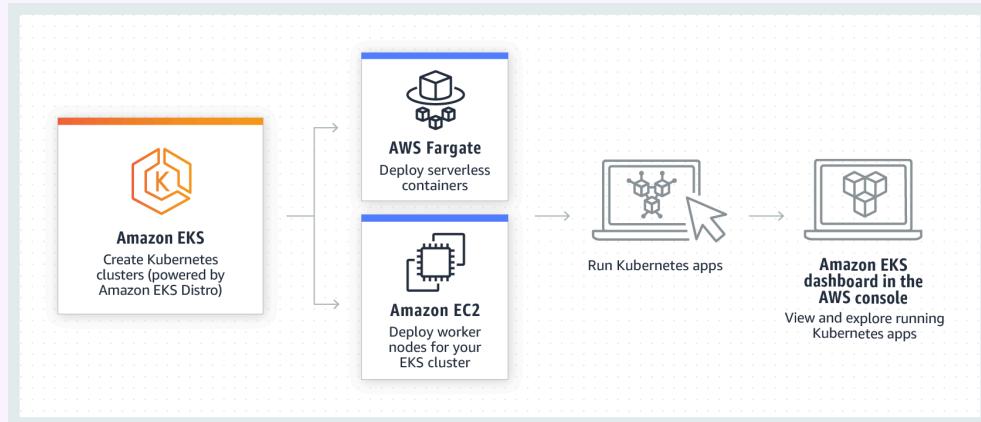




## EKS (Elastic Kubernetes Services)

نفس فكرة ال Docker ولديه 3 انواع

1. Mange Nodes :AWS : EC2 هي التي بتعمل وتدبر ال Nodes
2. Self Management Nodes : وهي تدرها من خلالها انت تعمل الNodes وتسلمهما لل EKS
3. AWS Forgate : بحيث تكسب مميواتها لو انت مش عارف البنية التحتيه تستحمل قد اي : عاوز الموضوع يتدار لوحده



## Ai Services

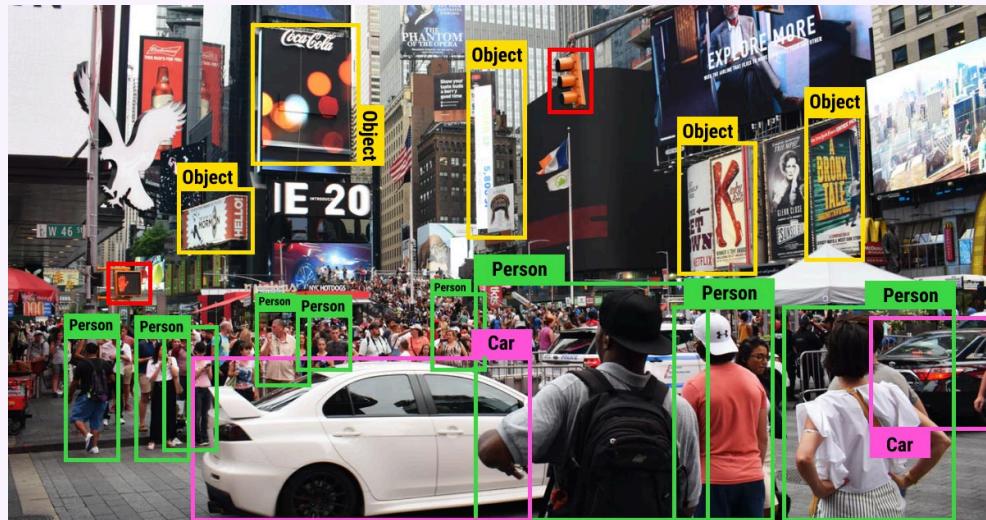
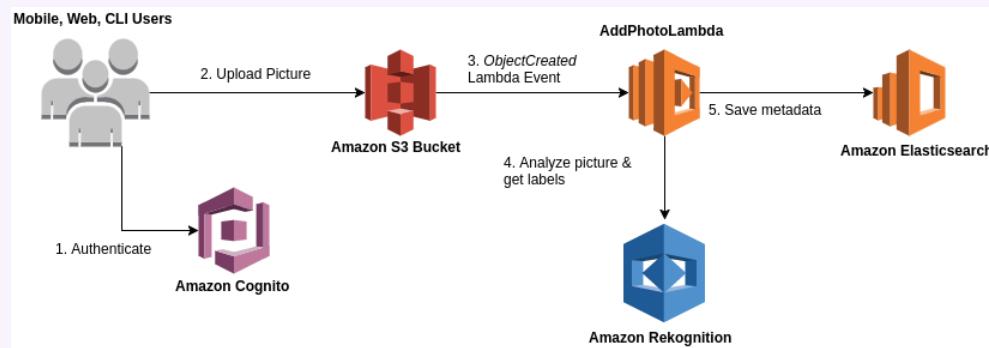


## Recognition

Use for Analysis the Images and Videos

مثلاً تحلل صورة او فيديو تدور فيها على حاجه معينه

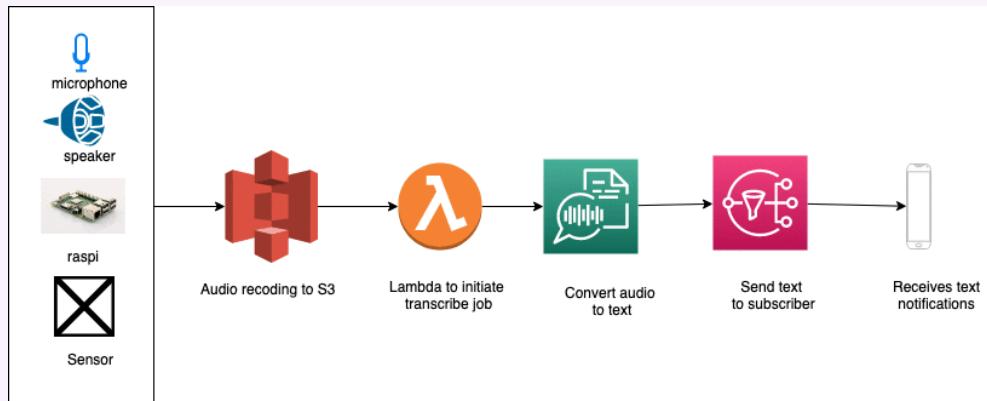
مثلاً هديك صورة لشخص تحللها وتقللي عنده كام سنه او مثلاً هديك صورة تطاعلي منها النصوص  
وعبرو من شغل ال Machine Learning





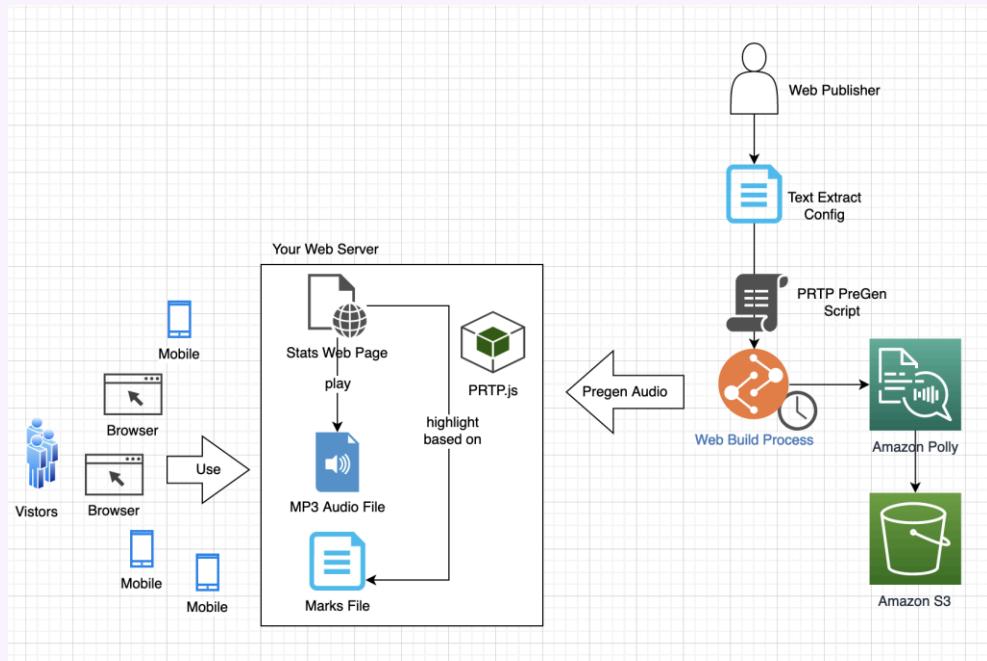
## Transcribe

Convert Audio To Text



## Polly

Convery Text To Audio

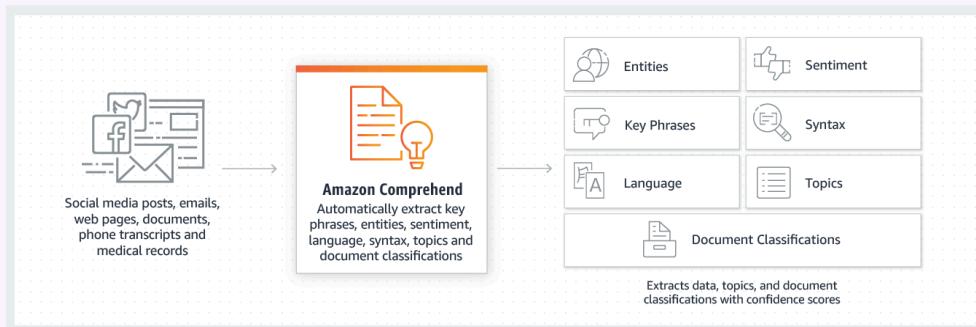




## Comprehend

Used to Analysis Text

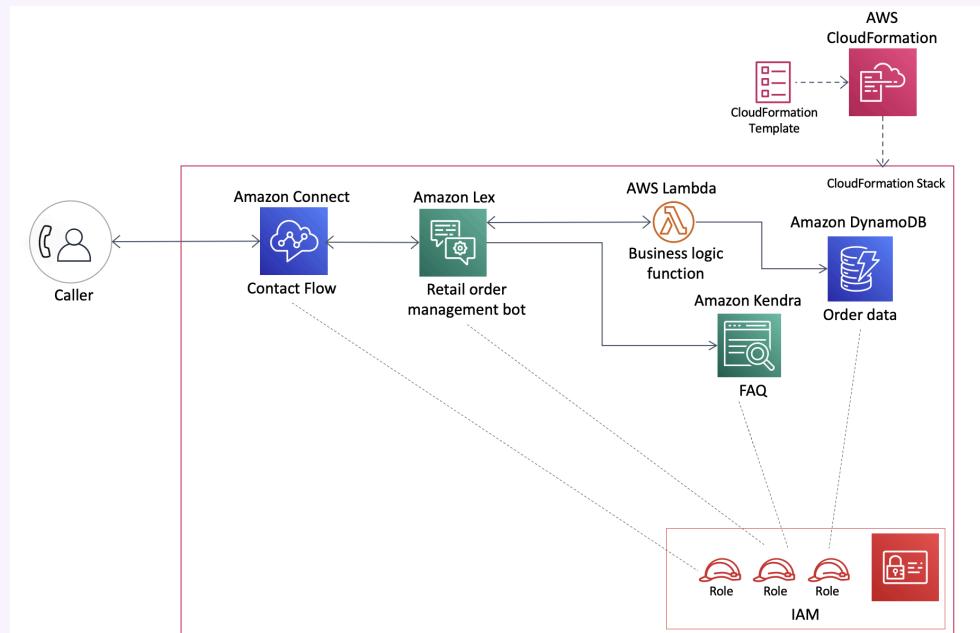
زي مثل انك تديلوالك وبيبدأ يحللوك هل هو ايجابي او سلبي يحلل محتواه وهكذا





## Lex

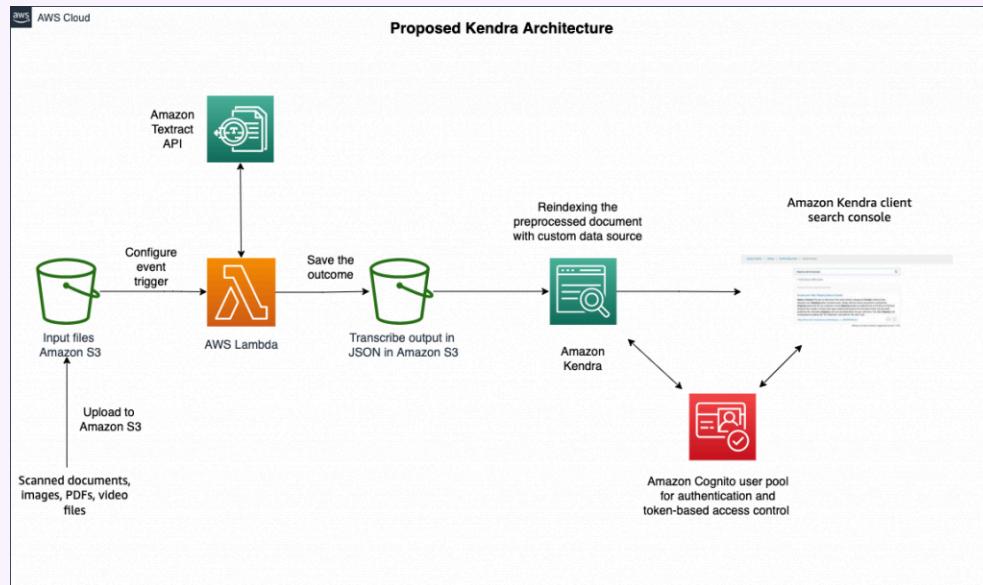
For Build Interactive Chatbot





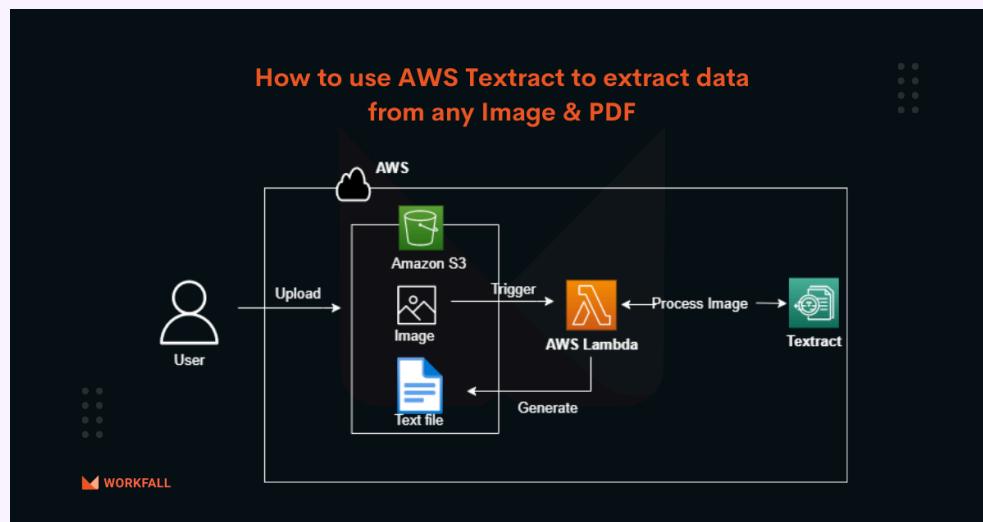
## Kendra

Allow you to search in Document you upload and get answers



## Textract

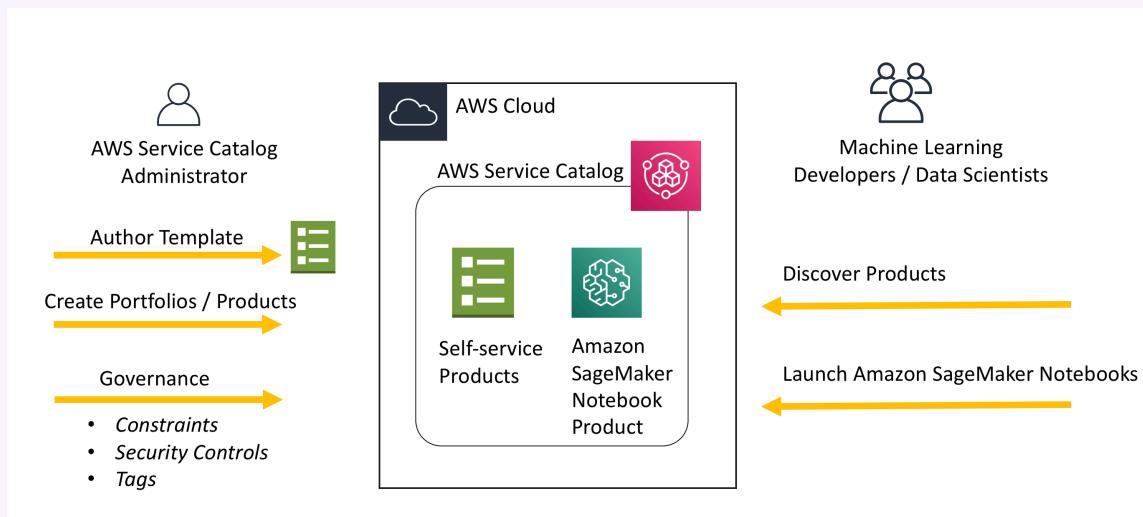
To Extract Text From Document to tables data





## Sagemaker

use to build or train machine lean Models



## Security Services In AWS



## WAF (Web Application Firewall)

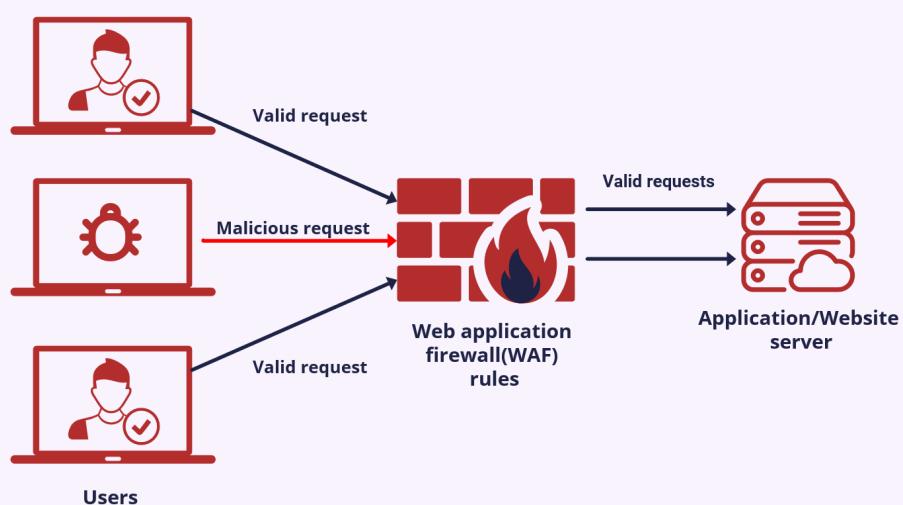
Monitor Attacks in HTTP Layer like SQL Injections , ...

Application load balancer or Cloud Front ماي بيشتغل على ايه  
بيشتغل ازاي ؟

عن طريق او بيبقى عندو قائمة من القواعد لي بيعطيها عليةا عشان يحمي ال Services بتاعتك

AWS ACL ودي اسمها

ممكنا تستخدم ال ACL او Predefine ACL or can create your custom ACL





## Shield

has two types

### 1. Simple :

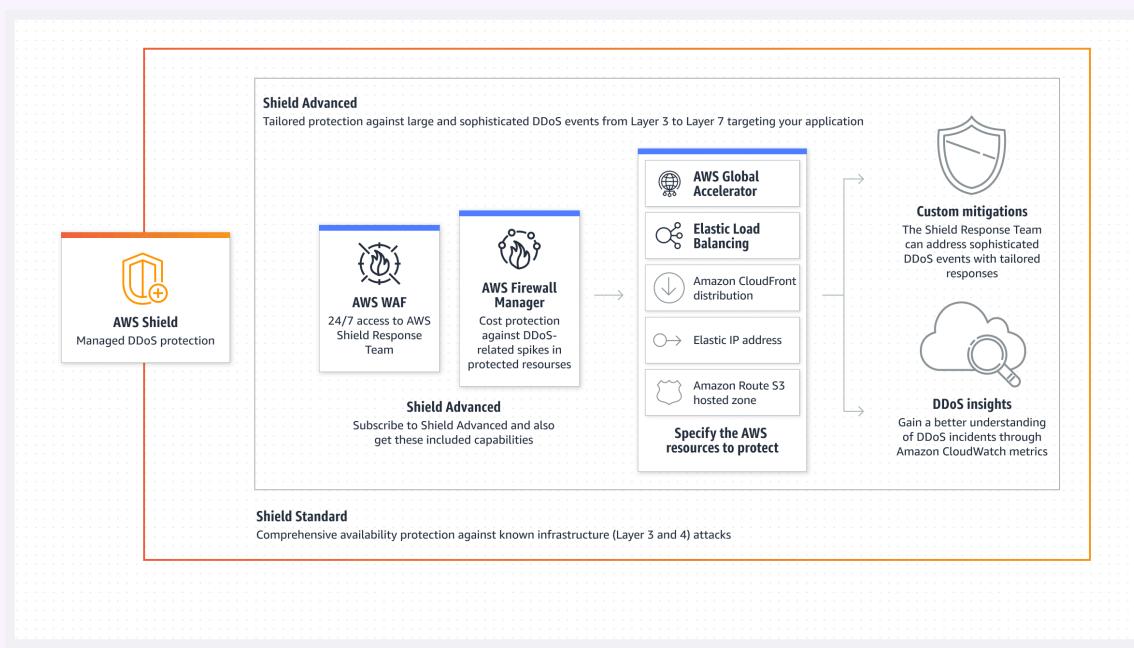
Protects against DDOS attacks in Layer 3 / layer 4

DDOS : دا يبيطل لما مستخدم يفتح اكتر من session بتعنك فمعنكم يوقعها

### 2. Advanced

Protect against Layer 7

ومن خلاله هتقدر تتصل بفريق هو اللي يحلل المشكله





## Data Encryption

: عشان تشفّر البيانات عندك نوعين

### 1. Encryption Transit

ودا بتعملو لما تعاوز تشفّر البيانات وهي بتدرك يعني ال Traffic Move

### 2. Encryption at Rest

وهنا انت هيبقى عندك مثلاً ملفات وعاوز تشفّرها وهي ساكته يعني مفيش

Traffics moved

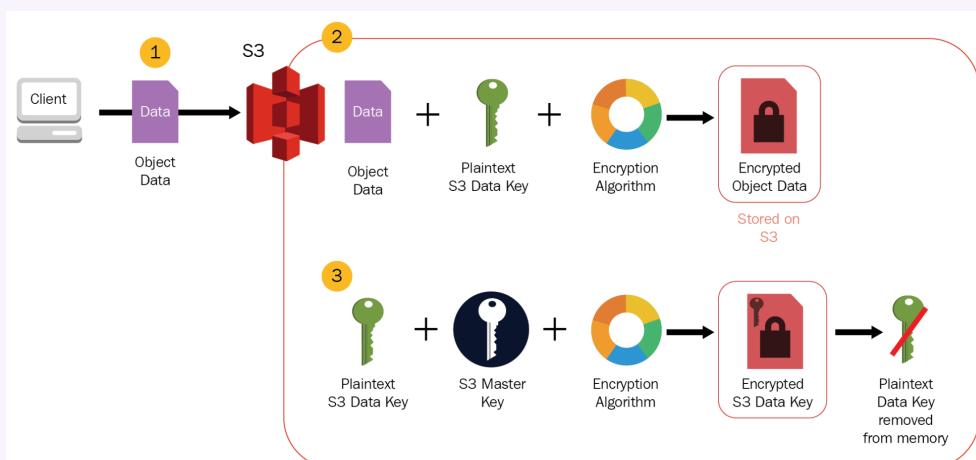
## Certificate Manager

خدناها قبل كدا لو تفتك في ال Application Load Balancer

وقلنا انها عبارة عن Public Key & Private Key

وهنا هنستخدمها في ال Encryption Transit

بحيث ان اللي هيعتلي هقلو شفرلي البيانات بال Public Key ولما تجيلى هفك تشفّرها بال Private Key





## KMS (Key Management Service )

دا مهمتو انو يدير ال Encryption Keys الي هنستخدمها عشان نشفر اي بيانات  
ليه نوعين :

انت الي بتحط الصلاحيات من خلال ال IAM

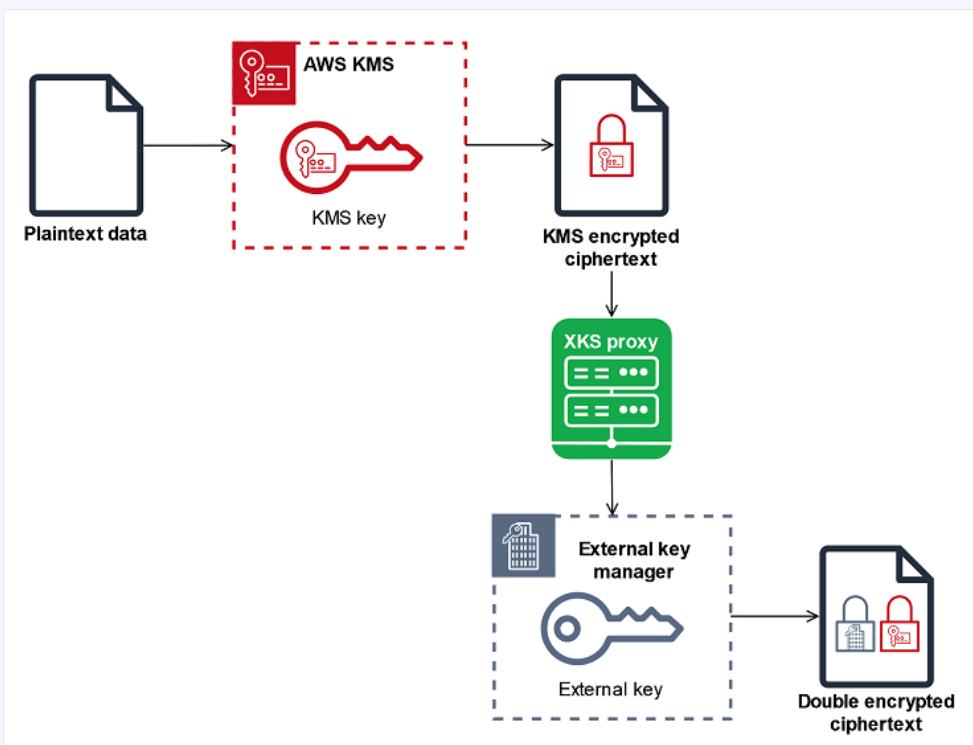
AWS Key : Key by AWS

طب انت لما تشفر بيانات مثلا علي S3 مين هيقدر يشوفها

لازم يكون معاه

1. Access to S3

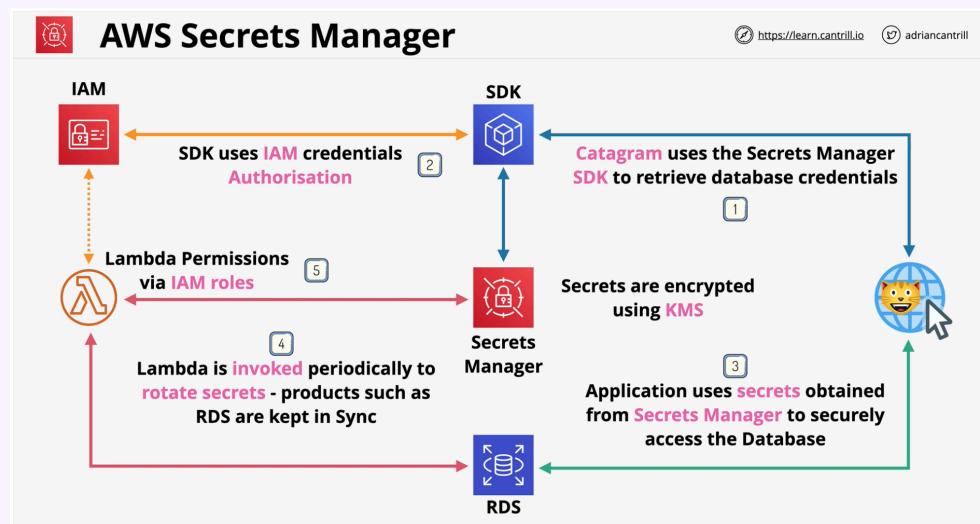
2. Access To Apply Decryption the Data





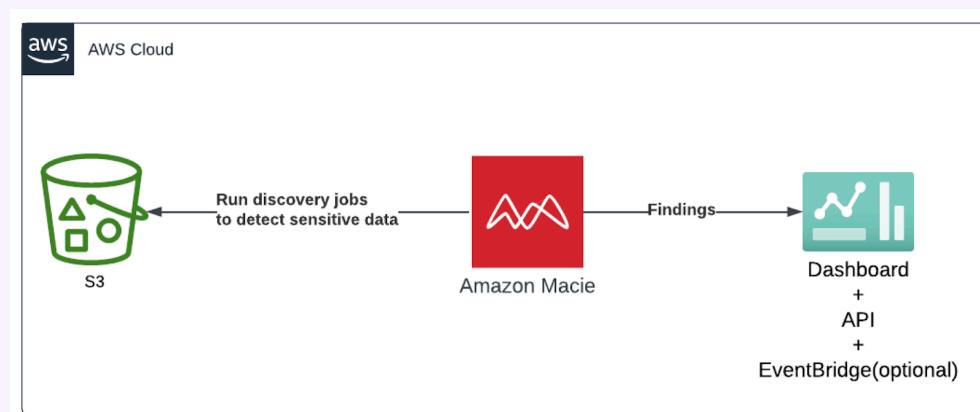
## Secret Manager

عشنان تحمي بيانات المستخد لو بيتصل بسرفر معين هتخزن الـ Credentials , UserName , Password in Secret Manger



## Macie

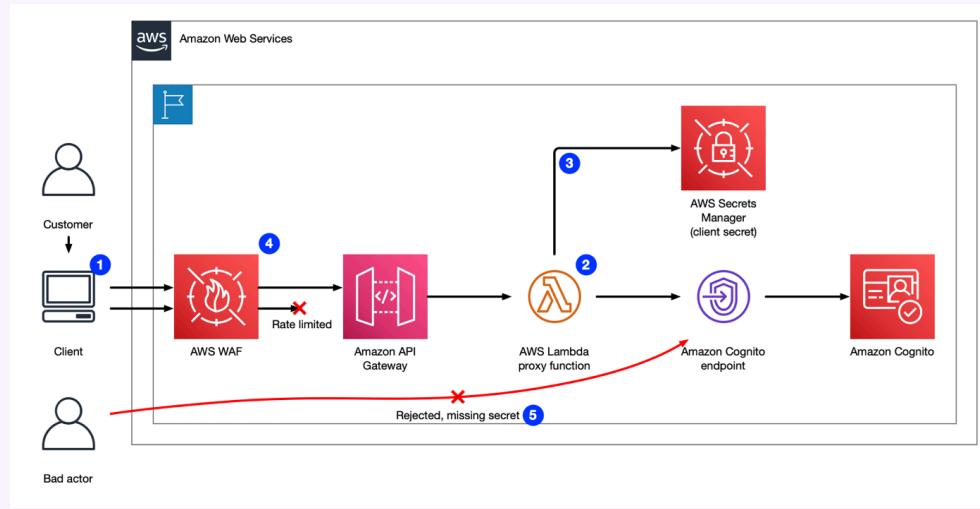
بتسخدم الـ PII in S3 عشنان تحمي الـ Machine Learning الي هي البيانات الشخصية يعني





## Cognito

- المستخدم يقدر يعمل حساب جديد أو يسجل دخول بحسابه.
  - تقدر تسمح له يستخدم:
    - الإيميل والباسورد.
    - أو حسابات خارجية زي Google, Facebook, Apple (OAuth)
- يعني بيستخدم ال Token عشان يقدر يدخلك سواء من موقع او موبيل





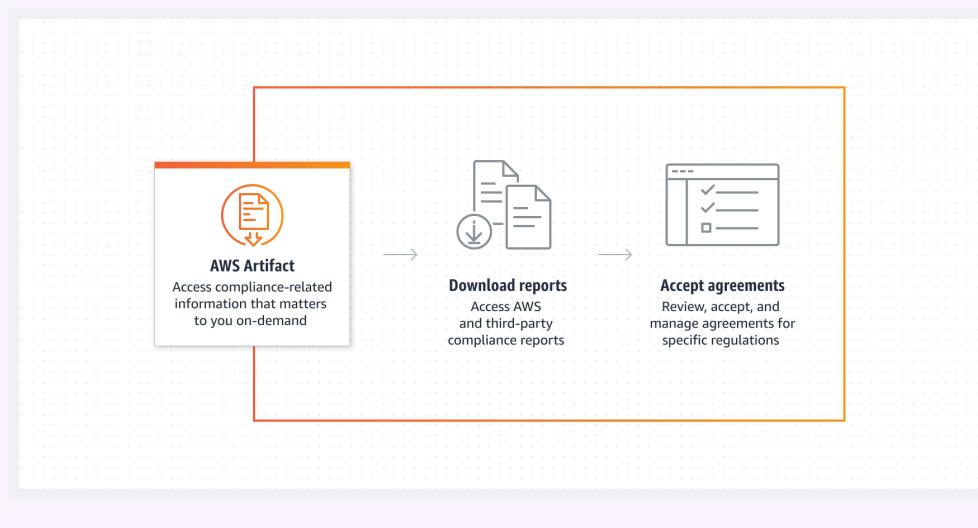
## Artifact

| هي خدمة بتوفرك وثائق وتقارير الامتثال والأمان (Compliance & Security Reports) الخاصة بـ AWS.

يعني لو شركتك محتاجة إثبات إن AWS ملتزمة بقوانين معينة زي

- GDPR
- ISO 27001
- SOC 1, SOC 2
- HIPAA

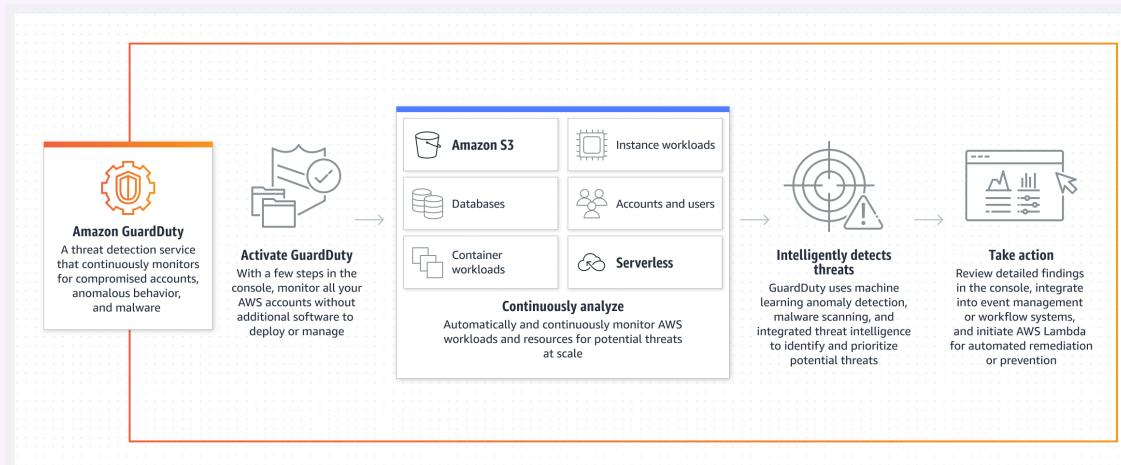
فتقدر تنزل كل التقارير دي من AWS Artifact





## GuardDuty

Analysis Logs in VPC / DNS to show you a unexpected or unauthorize activity

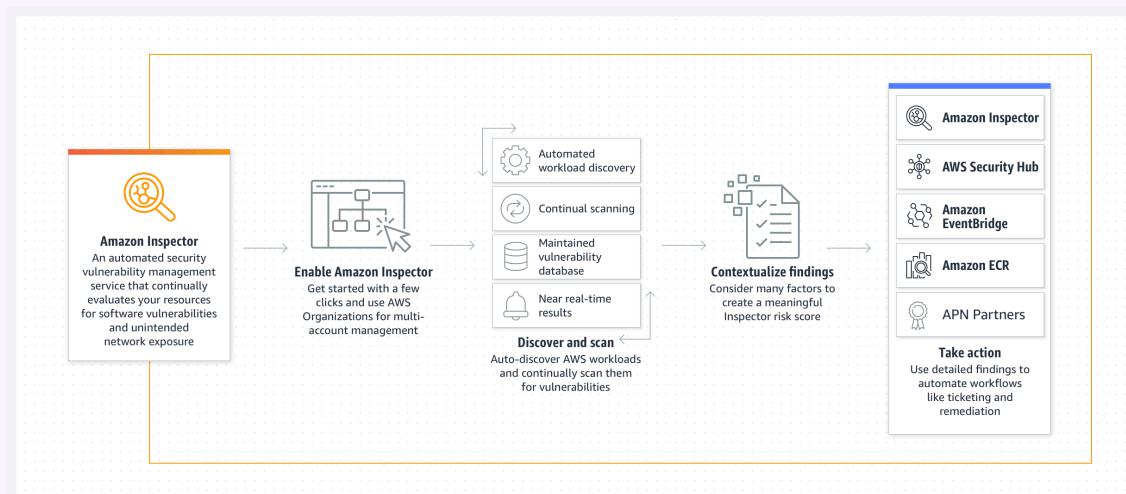




## Inspector

هي خدمة بتعملك فحص أمني تلقائي على السيرفرات والتطبيقات بتاعتك علشان تكتشف:

- الثغرات (Vulnerabilities)
- الأخطاء في الإعدادات (Misconfigurations)
- المخاطر الأمنية



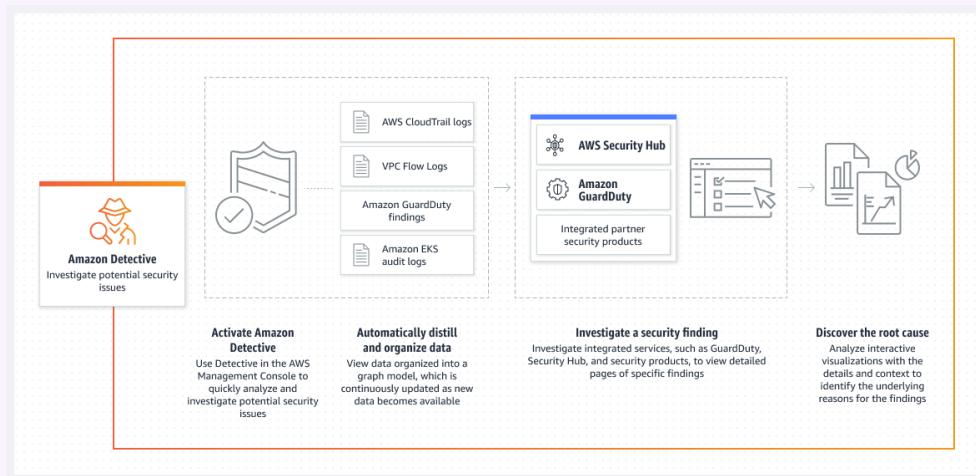


## Detective

هي خدمة بتحليل البيانات الأمنية من حساب بيتك، علشان تساعدك تتحقق في الحوادث الأمنية وتفهم إيه اللي حصل؟ وازاي؟ وليه؟

يعني ببساطة:

"محقق رقمي" بيدور على الأدلة ويحللها لما يحصل تهديد أو نشاط مريب في حسابك.





## Cloud Front

يستخدم في تحسين اداء القراءة للبيانات

by caching data and put it in Edge location , used in Static Content & Dynamic Content

**Edge location:** عبارة عن اماكن (Data Centers) دي اقرب للمستخدم

فبدل مطلب مثلا صورة من سيرفر في أمريكا ويتحقق في تخزين على ميجابايت

ممكن الحصول على طلبو عن طريق ان ال CDN يوجد و على اقرب

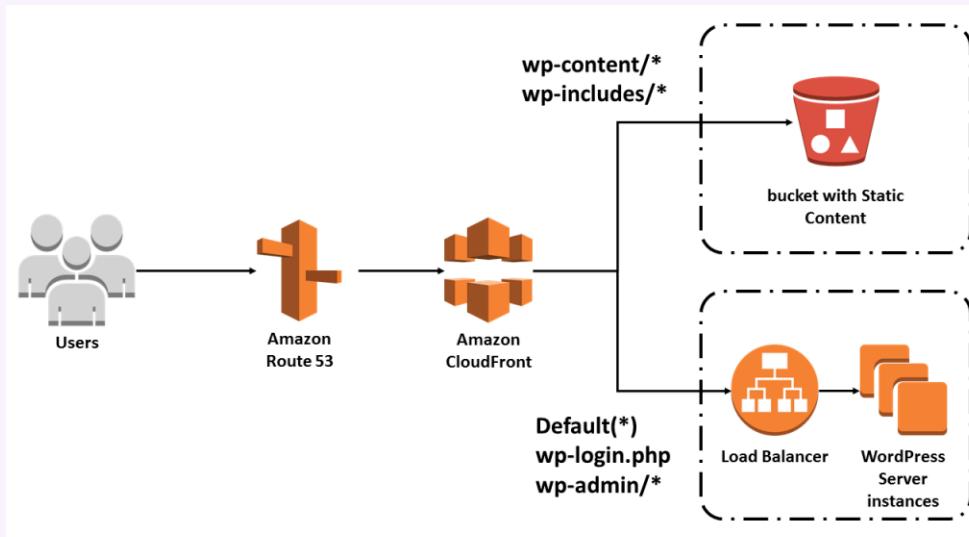
AWS has more than 450 Edge location

## Origin Access Control (OAC)

من خلالها بفتح خدمة

زي مثل لو عندك ملف معين وانت عاوز الـ CloudFront يشتغل عليه

هتدي صلاحيه من الـ OAC وتعمل **Policy** على S3





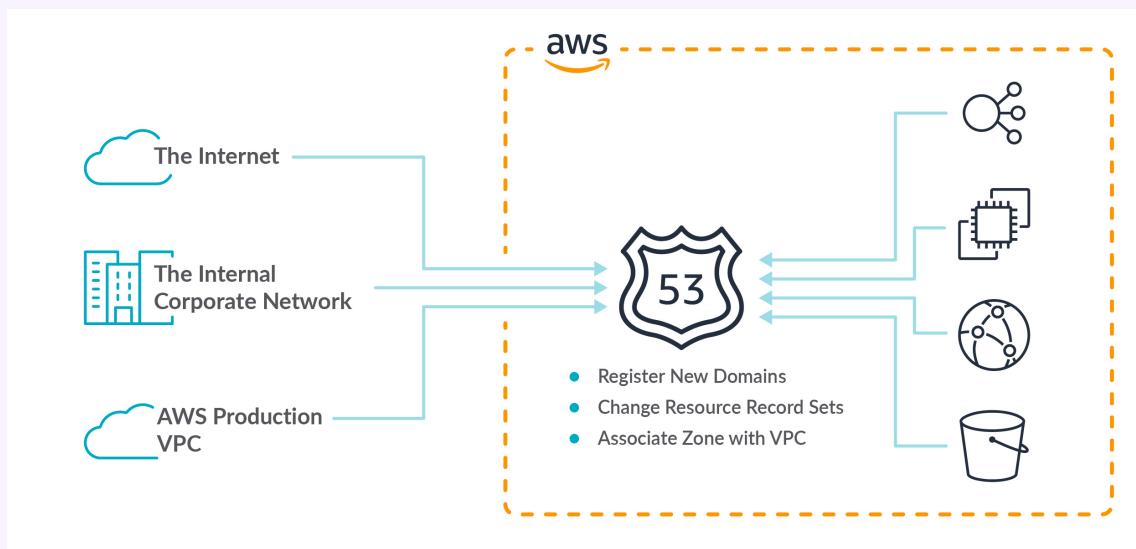
## Route 53 (Amazon DNS)

نفس الـDNS في الشبكة تحول إلى IP Address

Types of Records in DNS :

1. A record ⇒ Domain to IP Address
2. CNAME ⇒ Domain to Domain  
يشير إلى دومين ثاني  
**subdomain** لما تجب تخلية
3. Alias ⇒ Root Domain to AWZ Service

من	إلى
example.com	d1234.cloudfront.net ← CloudFront
example.com	my-loadbalancer.amazonaws.com ← ELB
example.com	mybucket.s3-website-us-east-1.amazonaws.com ← S3 Website



## Routing Policies

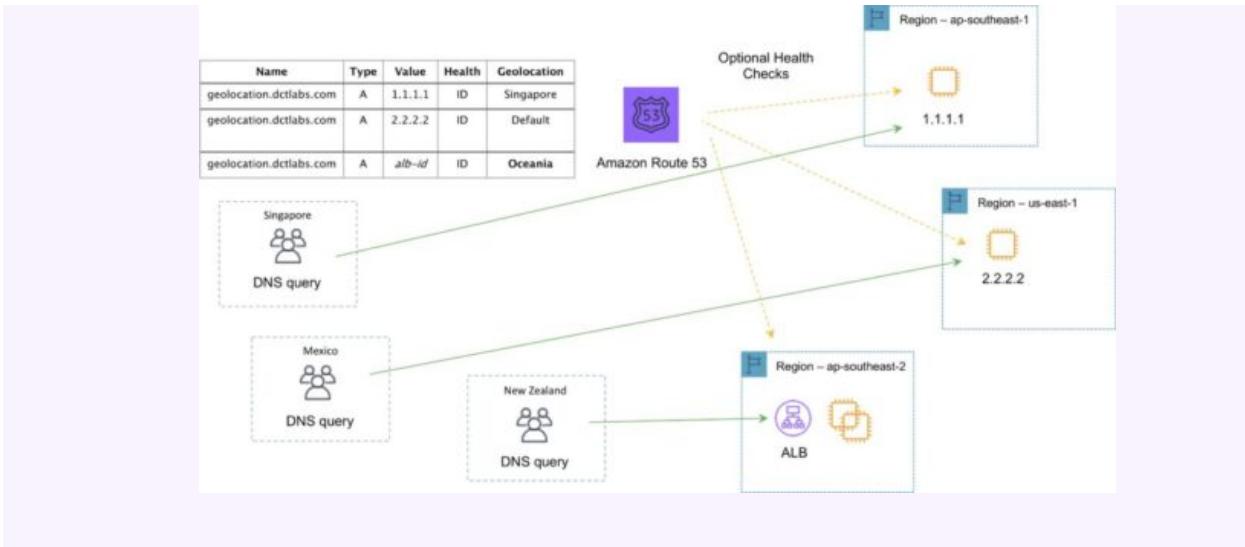
1. **Simple** : ينبع IP من دومين

2. **Weighted** : يمكنك تحكم أكثر بحيث تقول مثلاً

لو جالك حد موقعه الجغرافي كذا اديله الـIP ولو من موقع اخر اديله ثانٍ

او تحددها بالـlatency

وطبعاً على نفس الـDomain



## Outpost

افترض انك نظام بنك فانت عاوز تستفيد من خدمات AWS وفي نفس الوقت خايف على بيانات العملاء

وعوزها قدام عينك يعني ال (On-Prime)

يبقى حلا هو ال Outpost

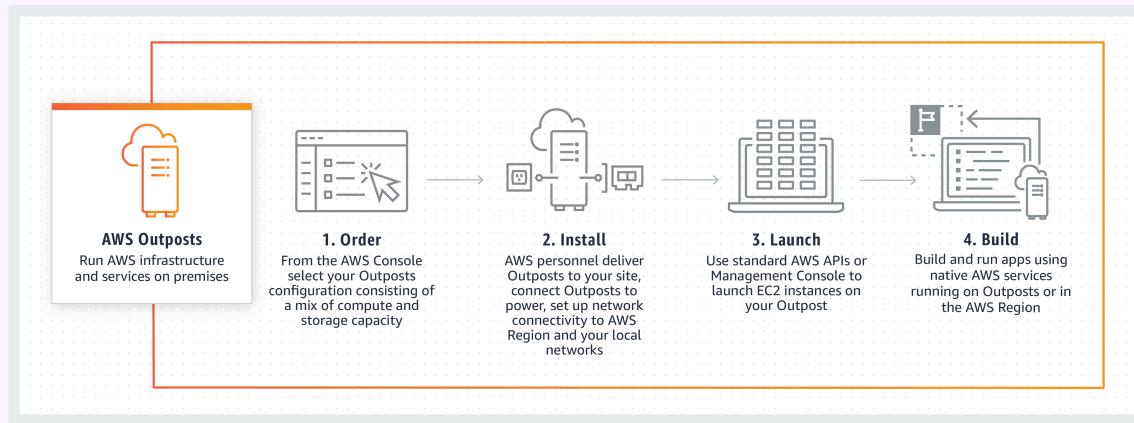
فكرتوا انك هيبقى عند Pre-cables & Pre-configurated Rack(Full mangle)

وتقدر بقا تستفيد من بعض خدمات AWS

بردو ممكن تستفيد من ال او اقرب AZ منك هتسبب High Latency

بس مش كل ال Outpost بتدعم ال anchor point

بس كل ال Regions بتدعم ال Outpost الاول تتأكد انها بتدعم ال





## Wavelength

زي ال بس دا خاص بال Outpost

حيث تربط AWS Services

تخيل إنك بتلعب لعبة أونلاين أو بتسخدم تطبيق VR

أو بتتفرج على فيديو لايف، وكل ده يحتاج يكون سريع جدًا ومفيش تأخير

المشكلة إن السيرفرات بتاعة AWS

ممكن تكون في بلد تانية زي ألمانيا، فالإشارة بتأخذ وقت تروح وتيجي

**Wavelength** تحل المشكلة دي.

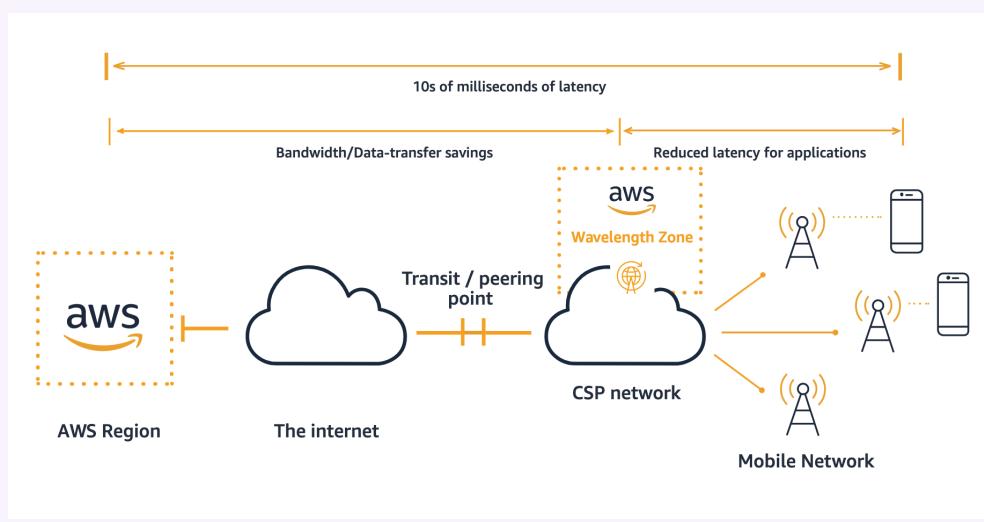
### 💡 بتعمل إيه بالظبط؟

راحت لشركات الموبايل (زي فودافون واتصالات) وقالتلهم:

"إحنا هنحط شوية سيرفرات عندكم جوا شبكتكم"

يعني بدل ما الداتا تروح سيرفر في أوروبا، لأن، هتروج لسيرفر قريب منك، في نفس بلدك أو مدينتك

يبقى قليل جدًا (Latency) وده يخلي التطبيق سريع جدًا... زمن الاستجابة





## Local Zone

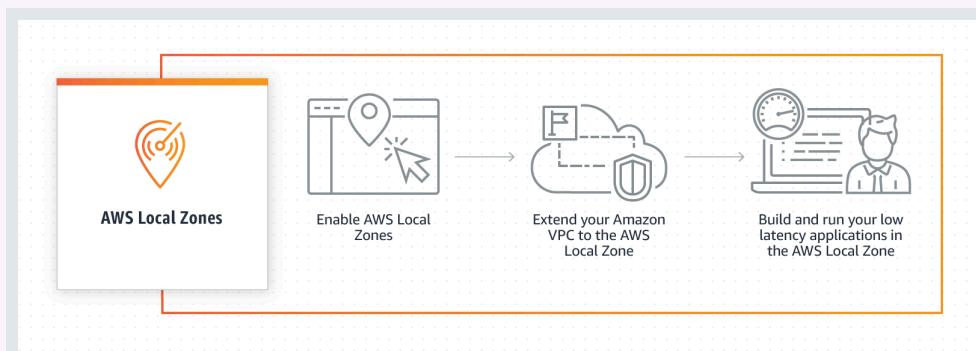
تخيل إن عندها مراكز بيانات كبيرة في دول زي ألمانيا، أمريكا، الهند... الخ.  
المراكز دي بعيدة، فلو التطبيق بتعاك تحتاج سرعة عالية، الداتا بتتأخر شوية.

هنا بيجي Local Zone.

هي ببساطة:

فرع صغير من AWS بيتفتح في مدينة قريبة منك ◆

عشان تقدر تستخدم خدمات AWS الأساسية بسرعة كبيرة وبدون تأخير





## Organization

### يعني إيه

تخيل إنك عندك شركة كبيرة، وفيها أكثر من فريق واحد يشتغل على تطبيق الموبايل، واحد على الويب، واحد على AI ... كل فريق عنده حساب AWS لوحده

هي الطريقة اللي بتخليك تجمع كل الحسابات دي تحت إدارة واحدة **AWS Organizations**، هي الطريقة اللي بتخليك تجمع كل الحسابات دي تحت إدارة واحدة، وتتحكم فيهم بسهولة، وتتوفر في التكلفة، وتطبق سياسات موحدة.

### ليه؟

1.  إدارة كل الحسابات من مكان واحد

2.  تطبيق سياسات موحدة (زي مين يقدر يعمل إيه، أو منع إنشاء خدمات معينة)

3.  يعني كل الحسابات تدفع مع بعض فتاخذ خصم أكبر (Consolidated Billing)

4.  توفير فلوس عن طريق تنظيم أفضل للحسابات حسب الفرق أو المشاريع أو البيئات (development / testing / production)

### مكوناتها ببساطة:

العنصر	معناه ببساطة
<b>Management Account</b>	الحساب الرئيسي اللي بيدير باقي الحسابات
<b>Member Accounts</b>	الحسابات اللي تحت الإدارة (زي فرقك أو مشاربك)
<b>Organizational Units (OU)</b>	تقسيمات تقدر تنظم فيها الحسابات (مثلًا gOU لـ Developers و OU لـ Testing)
<b>Service Control Policies (SCPs)</b>	سياسات تتحكم في الصلاحيات على مستوى الحسابات كلها

### مثال :

- عندك شركة اسمها **My Company**.

- بتعمل:

- تطبيق موبايل (AWS Account 1)
- موقع ويب (AWS Account 2)
- ذكاء صناعي (AWS Account 3)

بدل ما كل واحد يدفع لوحده ويدير خدماته لوحده

➡ **AWS Organization**

➡ بتخلصي حسابك الرئيسي هو الـ **Management Account**

➡ وتنصيف الحسابات الثانية كـ **Member Accounts**

تقدر تطبق سياسة واحدة عليهم، وتتحكم فيهم من مكان واحد

## أين فايداً في Consolidated Billing؟

لو الحساب الأول استهلك 200 دولار

والثاني 300 دولار

والثالث 500 دولار

هتجمع كلها وتدفعها مرة واحدة: **1000 دولار**

ولو فيه خصومات أو عروض ، هتتقاد على المجموع ده كله مش كل واحد لوحده.

أكيد، باختصار شديد:

## SCP (Service Control Policy)

AWS Organization هي سياسة بتدطها على الحسابات جوا

علشان:

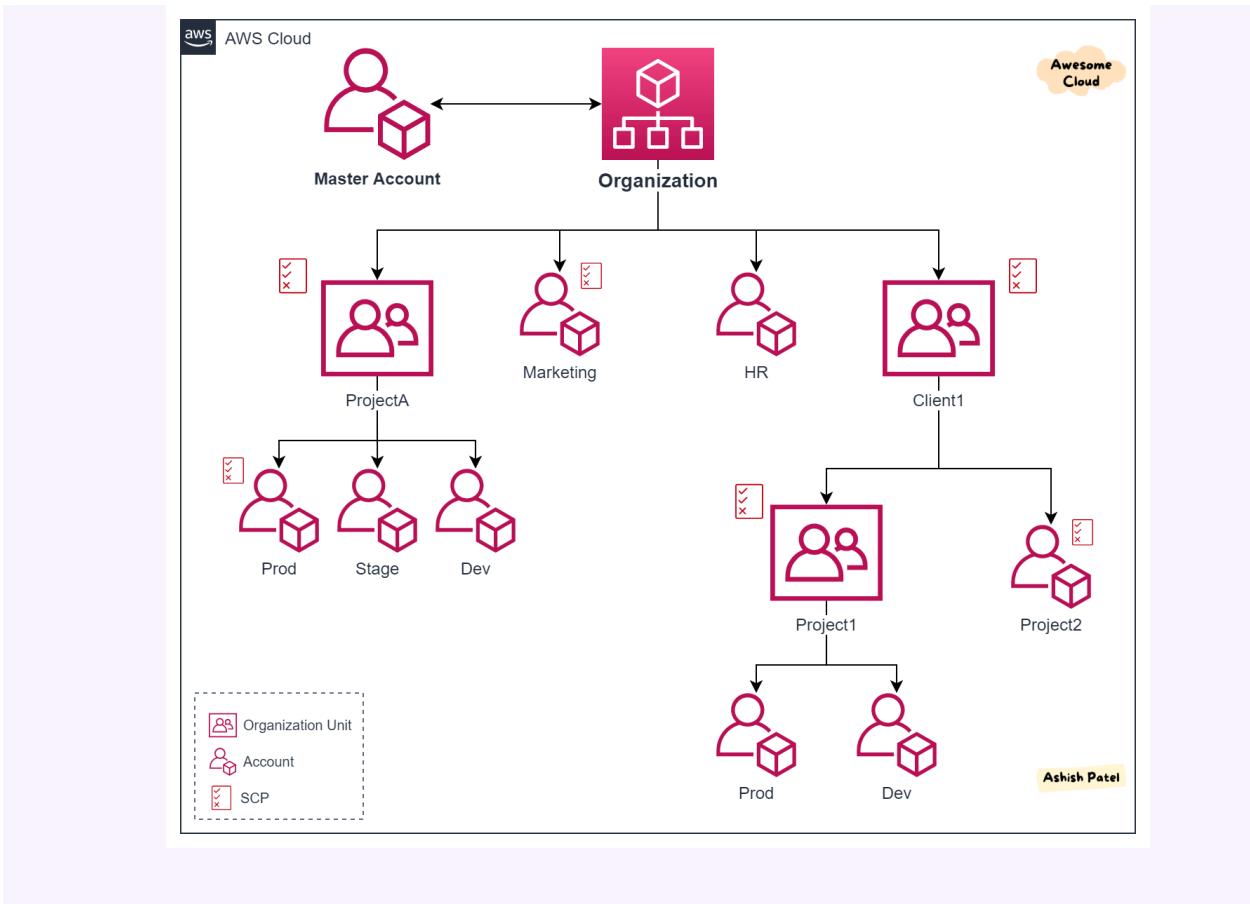
- تمنع أو تسمح باستخدام خدمات معينة.
- بتتحكم في الصلاحيات على مستوى الحساب كله.
- تشتغل مع IAM لكن أقوى منها (يعني تقدر تمنع حاجة حتى لو إلاما يسمح بيها).

## مثال:

عاوز تمنع كل الحسابات من حذف S3 Bucket؟

تحط SCP يقول: Deny s3:DeleteBucket

فتحتى لو حد عنده صلاحية، مش هيقدر يحذف.





## AWS Cloud Formation

IAC Infrastructure as a Code

### يعني إيه

أداة بتدخلك تبني كل البنية التحتية بتاعتك (سيرفرات، قواعد بيانات، شبكات)

بكوود! مش يدوي من ال

### بتفيidak في إيه؟

- يدوي VPC وبعدها EC2، وتعمل AWS بدل ما تقعد تدخل على...
- بتكتب ملف واحد (YAML أو JSON).
- تقراه وتبني كل حاجة لوحدها AWS

### ليه مفيدة؟

الفايدة	الشرح
السرعة	تبني كل حاجة بكوود مرة واحدة
التكرار	(زي) تقدر تطبق نفس البنية في أكثر من بيئه testing / production
التتبع	كل التغييرات موجودة في الكود
التراجع	لو في حاجة باطلت، تقدر تعامل Rollback

### تبني إيه؟

- EC2 (سيرفرات)
- S3 (تخزين)
- RDS (قواعد بيانات)
- VPC
- IAM Roles
- وكل حاجة تقريباً في AWS

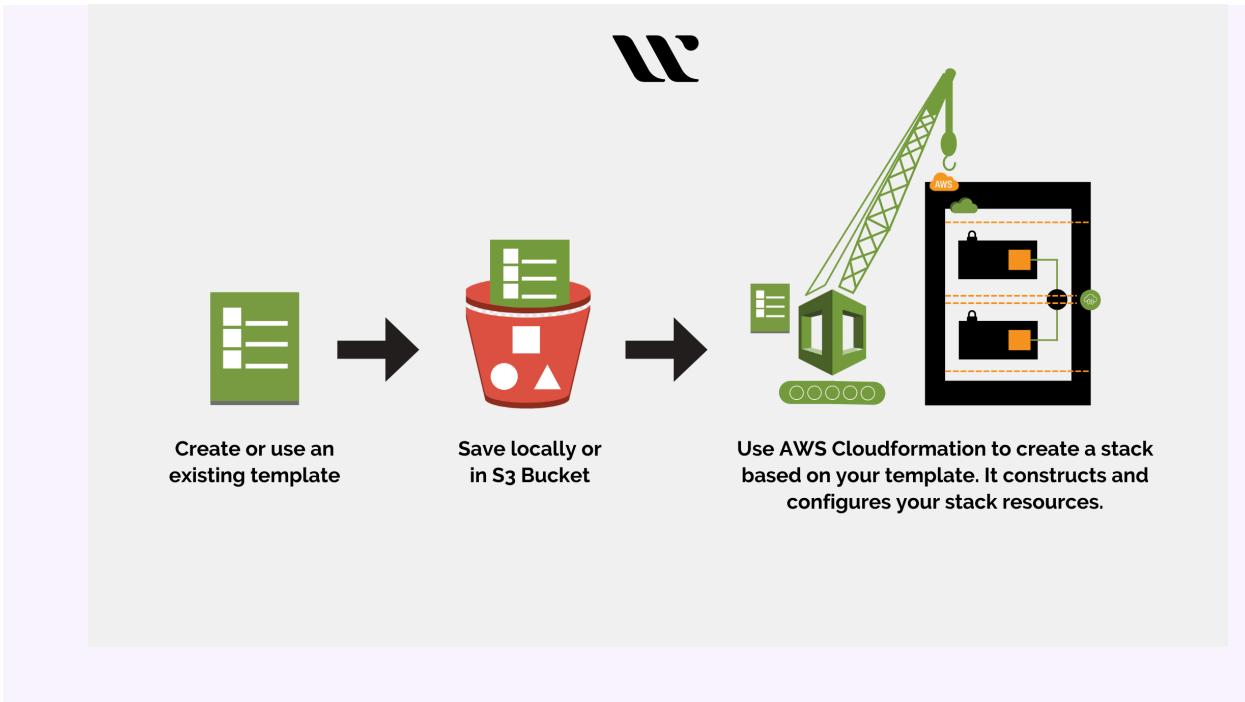
### بيبقى عامل إزاي (Template) شكل الكود:

Resources:

MyBucket:

Type: AWS::S3::Bucket

⬅ AWS: "Bucket MyBucket" في S3 اسمه ده يقول





## Snowball

تخيل انك في مكان مفهوش انترنت وعاوز تستفيد من خدمات AWS طب هترفع البيانات ازاي ؟  
بديل لك Snowball

### يعني إيه

هي "علبة/جهاز مادي" ( حقيقي ، مش خدمة سحابية ) من أمازون AWS  
يتستخدم علشان تنقل كمية كبيرة جدًا من البيانات من مكانك إلى AWS أو العكس

### بتغريك إمتي ؟

لما يكون عندك:

- تيرابايتات أو بيتابايتات من البيانات
- والإنتernet عندك بطيء أو مش عملي
- هيخد أيام أو شهور AWS ورفع البيانات على

لحد مكانك **Snowball** جهاز AWS هنا

تنسخ عليه البيانات

ترجعه لهم

أو أي خدمة تانية جوه S3 وهم يرفعو على AWS

### أنواع Snow Devices:

الجهاز	السعة التقريبية	الاستخدام
Snowball Edge	تيرابايت ~80	نقل بيانات + معالجة محلية
Snowball Edge Compute Optimized	زي اللي فوق لكن أقوى في الـ processing	
Snowmobile	شاحنة عملاقة ! بتتنقل بيتابايتات من البيانات ( زي مراكز بيانات كاملة )	

# AWS Snowball

- Transfer Data Fast & Securely | AWS Cloud

80TB



100TB





## Market Place

### ✓ يعني إيه؟

زى سوق أمازون = AWS Marketplace،  
بس بدل ما تشتري منتجات عادية  
AWS أو تستخدم برامج وخدمات جاهزة تشغّل جوه .

### 🌀 بتسخدمه في إيه؟

لو محتاج حاجة جاهزة بدل ما تبدأ من الصفر زى:

- Firewall من شركات زى) (Palo Alto)
- قاعدة بيانات متخصصة
- برامج حماية
- (Linux / Windows) أنظمة تشغيل
- أدوات ذكاء صناعي أو تحليل بيانات

بتعالك AWS وتقدر تشغّلها على طول في حساب Marketplace هتلaciها في الـ

### 💡 الأنواع اللي تلاقّيها:

النوع	مثال
AMI (سيفر جاهز)	Ubuntu + برماج جاهز عليه
SaaS (خدمة على الإنترن特)	Zy Datadog أو Splunk
أدوات DevOps	Jenkins, GitLab...
برامج حماية	Firewalls, IDS, وغيرها

### 💰 إزاي الدفع بيتم؟

- في خدمات مدفوعة شهرياً أو بالساعة
- وفي أدوات مجانية
- بتعالك AWS الدفع بيتم على نفس فاتورة





## Pricing Calculator

### ✓ يعني إيه

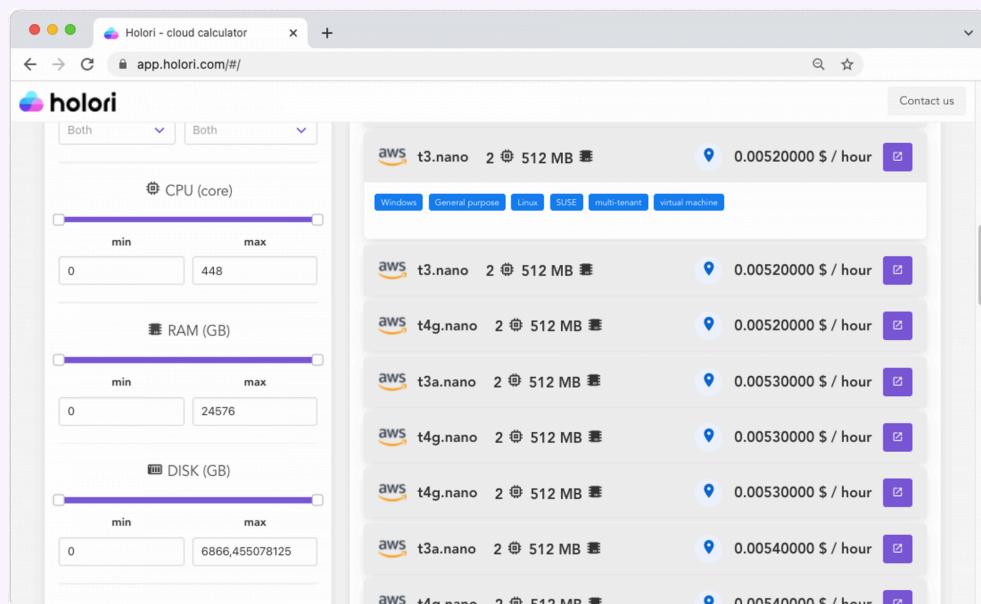
هو أداة من AWS على الإنترنت بستخدمها علشان

تحسب هتدفع كام لو استخدمت خدمات AWS

وغيرها زي EC2, S3, RDS

### 🎯 بتفيذك في إيه؟

- قبل ما تبدأ مشروعك، تقدر تشفف هتحتاج كام فلوس
- مثلاً: سيرفر صغير ولا كبير؟ تقدر تجرب سيناريوهات مختلفة SSD ولا HDD؟
- AWS بتساعدك في التخطيط المالي لأي مشروع على





## Budgets

### ✓ يعني إيه؟

**AWS Budgets** =AWS أداة بـتغليك تحط ميزانية (حد أقصى) لاستهلاكك في وتعتالك تنبيه لو قربت توصل أو تعدد الميزانية دي.

### 🎯 بتنفيذك في إيه؟

- **AWS** تتحكم في المصروفات على
- تعرف أنت صرفت قد إيه لحد دلوقتي
- تبعت لك أو لفريقيك تنبيهات على الإيميل لو استهلاكك زاد
- تقدر تتابع ميزانية خدمات معينة أو الحسابات الفرعية

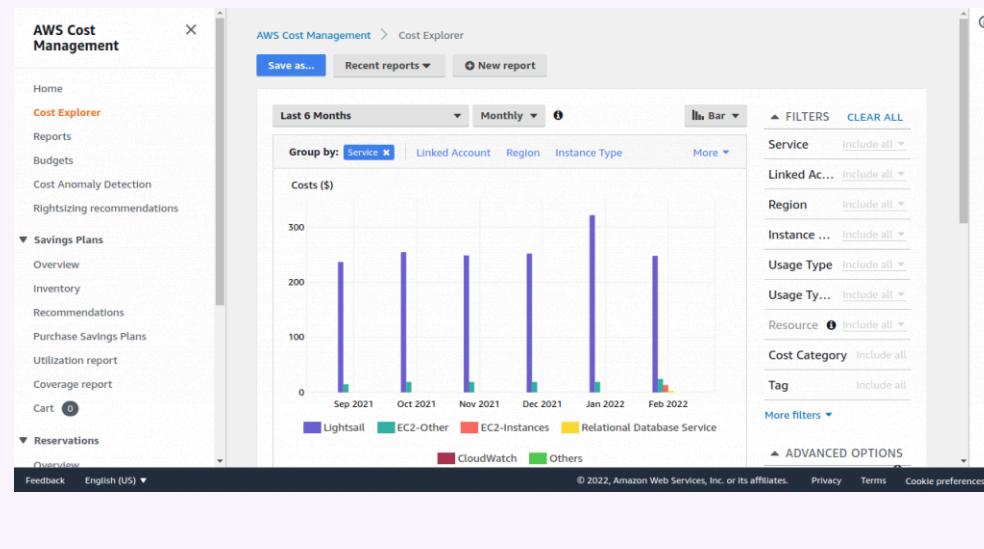
AWS Budgets							
<input type="text"/> Filter by budget name <span style="float: right;">?</span>							
All budgets (9)	Cost budgets (6)	Usage budgets (2)	Reservation budgets (1)	Savings Plans budgets (0)			Create budget
Project Nemo Cost Budget	Cost	\$42.48	\$40.00	\$42.48	<div style="width: 106.2%; background-color: #f08080;"><div style="width: 100%; height: 10px; background-color: #f08080;"></div></div> 106.2%	<div style="width: 106.2%; background-color: #f08080;"><div style="width: 100%; height: 10px; background-color: #f08080;"></div></div> 106.2%	...
S3 Usage Budget	Usage	4,571 Requests	5,500 Requests	7,343.83 Requests	<div style="width: 83.11%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 83.11%	<div style="width: 133.52%; background-color: #f08080;"><div style="width: 100%; height: 10px; background-color: #f08080;"></div></div> 133.52%	...
Eastern US Regional Budget	Cost	\$75.49	\$100.00	\$95.91	<div style="width: 75.49%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 75.49%	<div style="width: 95.91%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 95.91%	...
Total Monthly Cost Budget	Cost	\$121.04	\$200.00	\$169.62	<div style="width: 60.52%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 60.52%	<div style="width: 84.81%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 84.81%	...
Total EC2 Cost Budget	Cost	\$117.47	\$200.00	\$163.86	<div style="width: 58.74%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 58.74%	<div style="width: 81.93%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 81.93%	...
Quarterly Budget	Cost	\$277.09	\$550.00	\$483.04	<div style="width: 50.38%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 50.38%	<div style="width: 87.83%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 87.83%	...
Monthly DataTransfer Usage Budget	Usage	1.82 GB	4 GB	2.83 GB	<div style="width: 45.43%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 45.43%	<div style="width: 70.78%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 70.78%	...
Aggregate Reservation Coverage Budget	RI Coverage	25.49%	100%	-	<div style="width: 25.49%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 25.49%	-	...
SC	Cost	\$0.00	\$10.00	-	<div style="width: 0%; background-color: #4682B4;"><div style="width: 100%; height: 10px; background-color: #4682B4;"></div></div> 0%	-	...



## Cost Explorer

أداة بتحليل ت Shawf فلوكس راحت فين على AWS

بالتفصيل المعلم، برسومات وجداول، لكل خدمة وكل يوم.





## Trusted Advisor

### يعني إيه

AWS Trusted Advisor =

مستشار ذكي من

بيفحص حسابك ويقولك:

"في حالات معكן تصلحها أو تحسنها علشان توفر، أو تبقى أكثر أماناً، أو أسرع"

### ي يعمل إيه

بيفحص حسابك ويديك توصيات في 5 مجالات رئيسية:

المجال	يقولك إيه؟
<input checked="" type="checkbox"/> Cost Optimization	فين تقدر توفر فلوس (مثلاً سيرفر منش شغال وبتدفع عليه)
<input checked="" type="checkbox"/> Performance	إيه اللي معكن يخليلي الخدمات تستغل أسرع
<input checked="" type="checkbox"/> Security	(غلط IAM مفتوح أو مثلاً) هل في مشاكل أمان؟
<input checked="" type="checkbox"/> Fault Tolerance	أو حماية من الأعطال؟ هل في خدمات من غير
<input checked="" type="checkbox"/> Service Limits	هل قررت توصل للحد الأقصى لخدمة معينة؟

### Trusted Advisor Dashboard



#### Cost Optimization



8  0   
0  \$0.00  
Potential monthly savings

#### Performance



10  0   
0

#### Security



14  1   
2

#### Fault Tolerance



14  2   
1

#### Service Limits



50  0   
0

#### Recent Changes

Amazon RDS Multi-AZ

2/11/20

Security Groups - Unrestricted Access

2/11/20

#### What's New

Checks: New Best Practice Checks

Checks: Ten new service limit checks, plus Aurora DB availability, EC2 Windows configuration

Features: Automated actions and custom notifications with CloudWatch Events

Checks: Three new checks for AWS Direct Connect

Features: Filter check results based on resource tags



Done



Course Playlist : [https://youtube.com/playlist?  
list=PLZmPGUyBFvUqo76bXGnXq9EofsaV2d8K5&si=h5Gj4FIPFtWh5b4m](https://youtube.com/playlist?list=PLZmPGUyBFvUqo76bXGnXq9EofsaV2d8K5&si=h5Gj4FIPFtWh5b4m)

## 📌 My personal accounts links

LinkedIn	<a href="https://www.linkedin.com/in/ahmed-hany-899a9a321?utm_source=share&amp;utm_campaign=share_via&amp;utm_content=profile&amp;utm_medium=android_app">https://www.linkedin.com/in/ahmed-hany-899a9a321? utm_source=share&amp;utm_campaign=share_via&amp;utm_content=profile&amp;utm_medium=android_app</a>
WhatsApp	<a href="https://wa.me/qr/7KNUQ7ZI3KO2N1">https://wa.me/qr/7KNUQ7ZI3KO2N1</a>
Facebook	<a href="https://www.facebook.com/share/1NFM1PfSjc/">https://www.facebook.com/share/1NFM1PfSjc/</a>