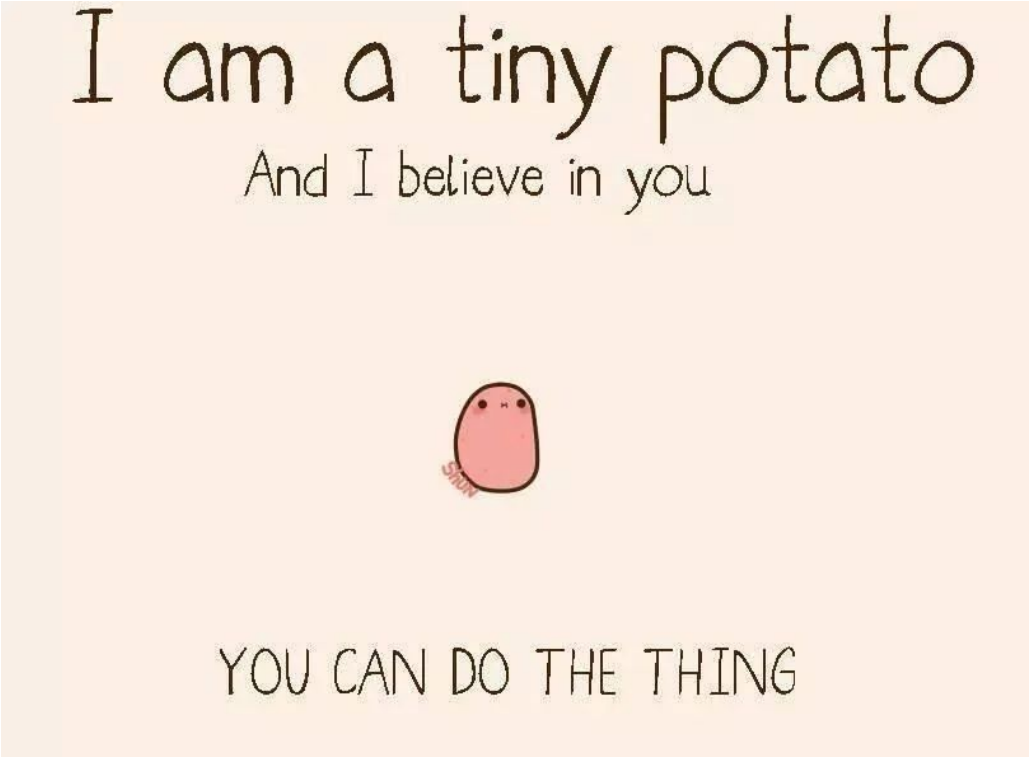
Amazon Primer

Things I wish they'd have told me before I got started

Kisha Delain
&
Levi McCormick

Outline

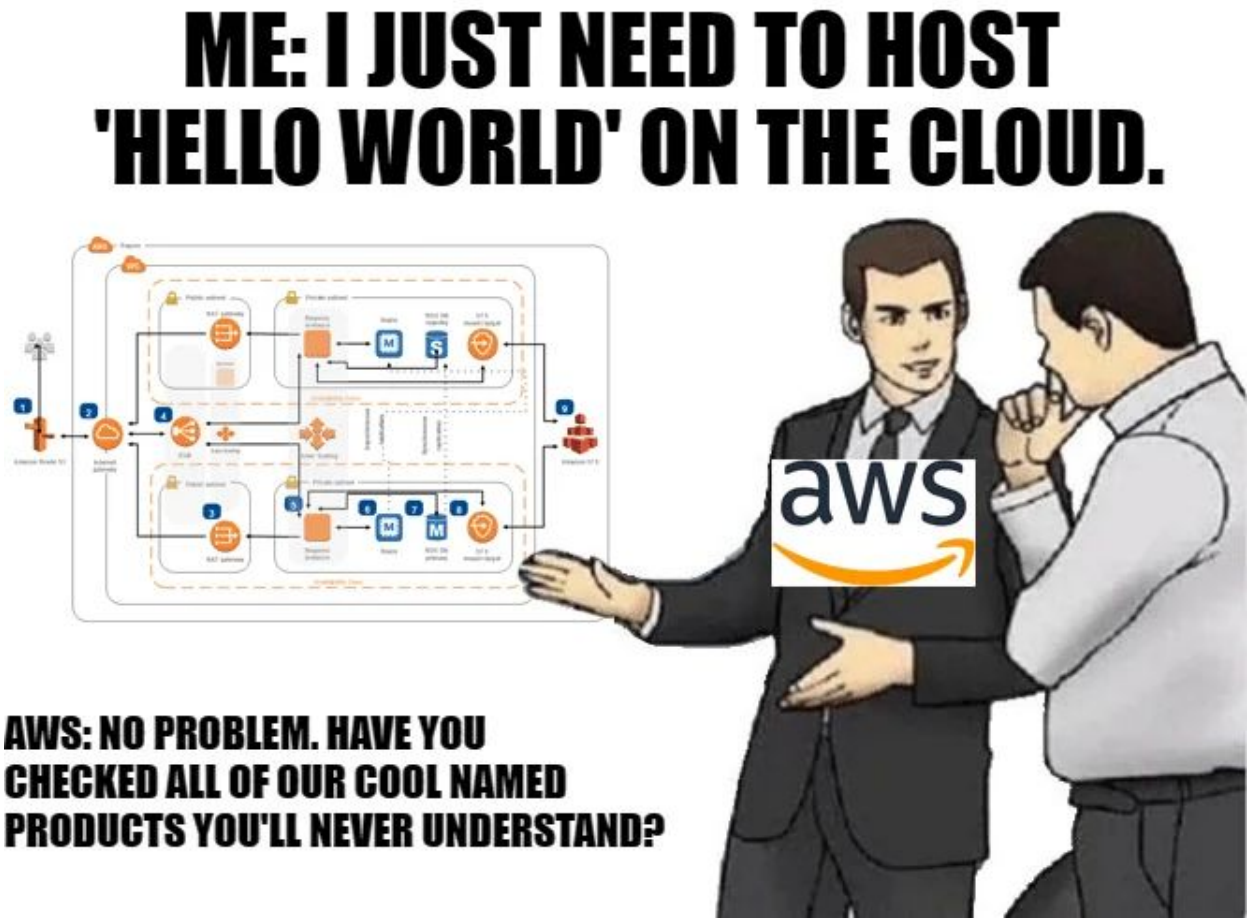
- Services
- Tutorials
- Know before you go: words of caution
- $$$$
- Q&A



200+ Services

- Elasticache
- RDS
- SQS queueuuueueueuee
- kinesis
- ECS/EKS
- EC2
- lamba
- IAM
- S3
- Cloudfront
- Cloudwatch
- Api gateway

… and many more



r/ProgrammerHumor

What are you trying to do?

Store Some Stuff
- Databases: RDS, DynamoDB
- Blobs: S3
- Cache: Elasticache

Transfer Stuff (Information)
- SQS queueueueueees
- API Gateway
- Cloudfront

Compute Some Stuff
- Event-driven, small: Lambdas + Step functions
- Too big? Too long?: EC2 / Fargate

Containerize Your Stuff
- ECS / EKS

Manage Your Stuff
- AWS Identity Center
- IAM

Watch Your Stuff
- CloudWatch

Machine Learning, Analytics, etc: there are a bunch here we won't touch

Services - Unsung Heroes

AWS IAM

Permissions

VPC

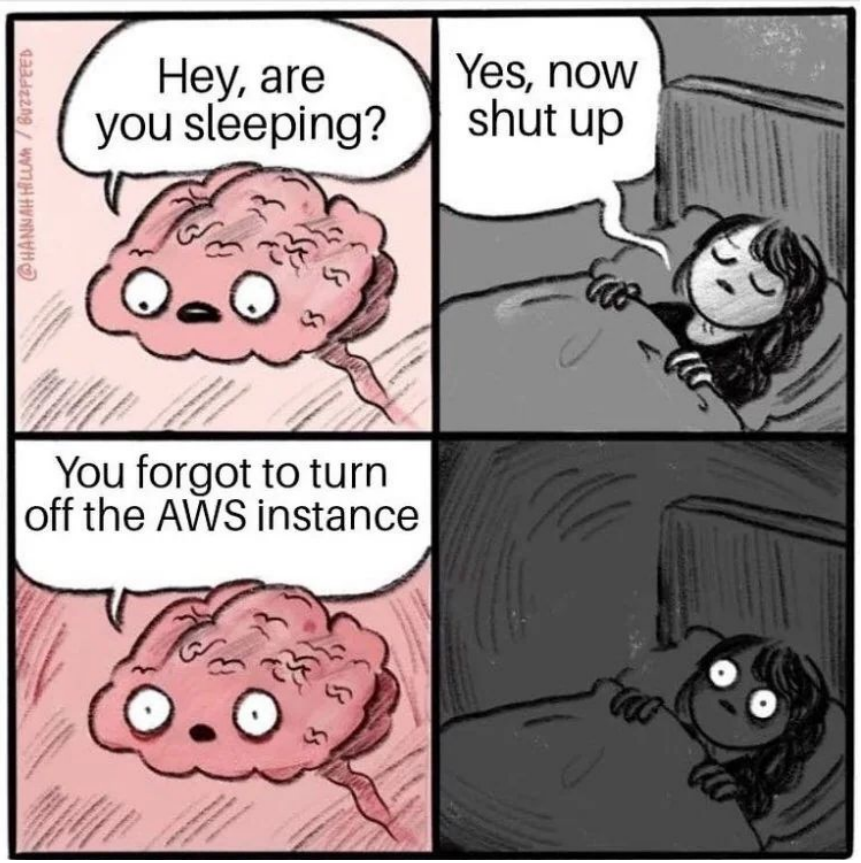
Networking

CloudWatch

Logging/Metrics

Where To Learn

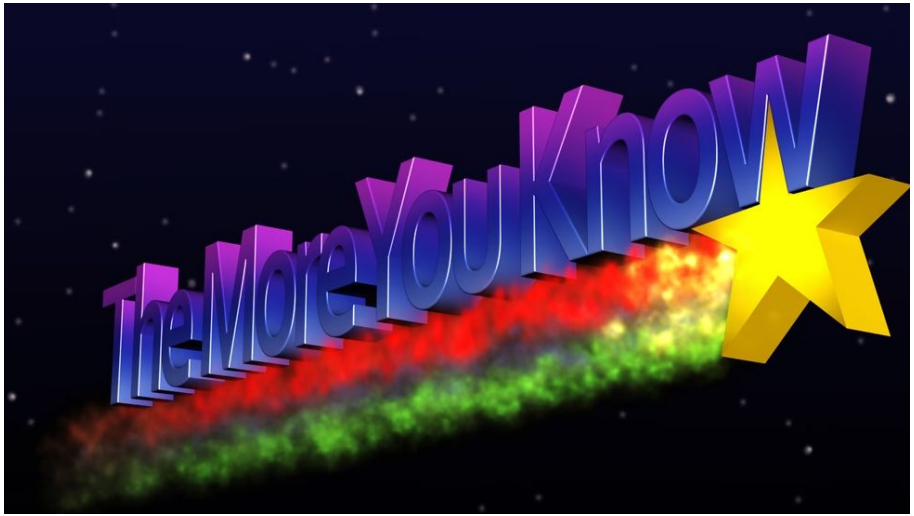
- AWS Tutorials
- Cloud Resume Challenge
- 100 Days of Cloud
- AWS Cloud Institute



Original comic by **Hannah Hillam**

Know Before You Go

- Don’t use Root Creds
- Don’t trust AWS IAM examples
- Multi-account
- Budgets
- Why is it so expensive?
- Well Architected Framework



Don’t Use Root Creds

Basically God Mode in the Cloud

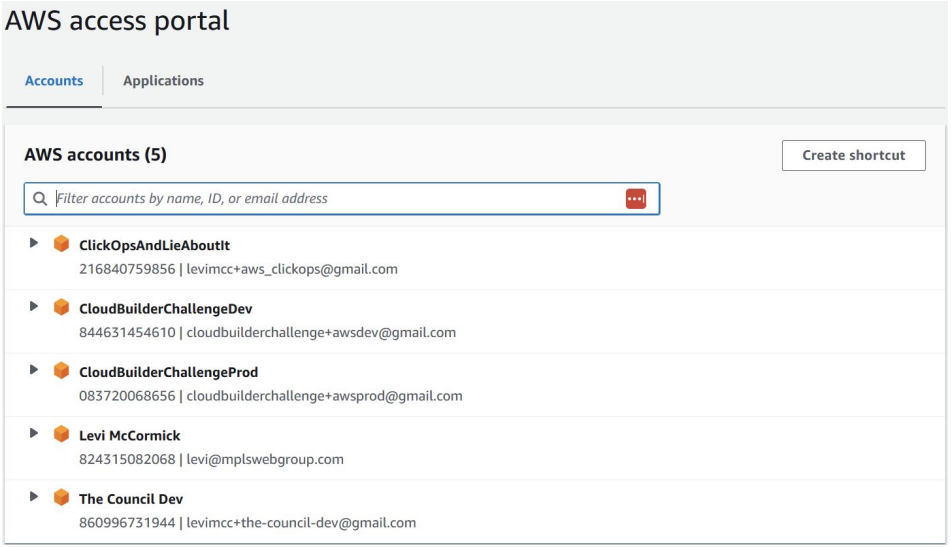
Most Reddit posts of compromised accounts are due to people exposing root credentials on Github.

Use restricted IAM users and/or role assumption instead.

AWS Identity Center

Service formerly known as AWS SSO. Manages Permission Sets and assigns them to users across accounts in an AWS Organization.

If you can't/won't use SSO, use IAM users with limited permission sets.



Don’t Trust AWS IAM Examples

```
s3_client = boto3.client(
    "s3",
    aws_access_key_id=AWS_ACCESS_KEY_ID,
    aws_secret_access_key=AWS_SECRET_ACCESS_KEY,
    )
```

AWS SDK Credential Chain Resolution

SDKs automatically resolve AWS credentials following a predictable pattern, prioritizing environment variables, then credential files, then a metadata API when running in the cloud.

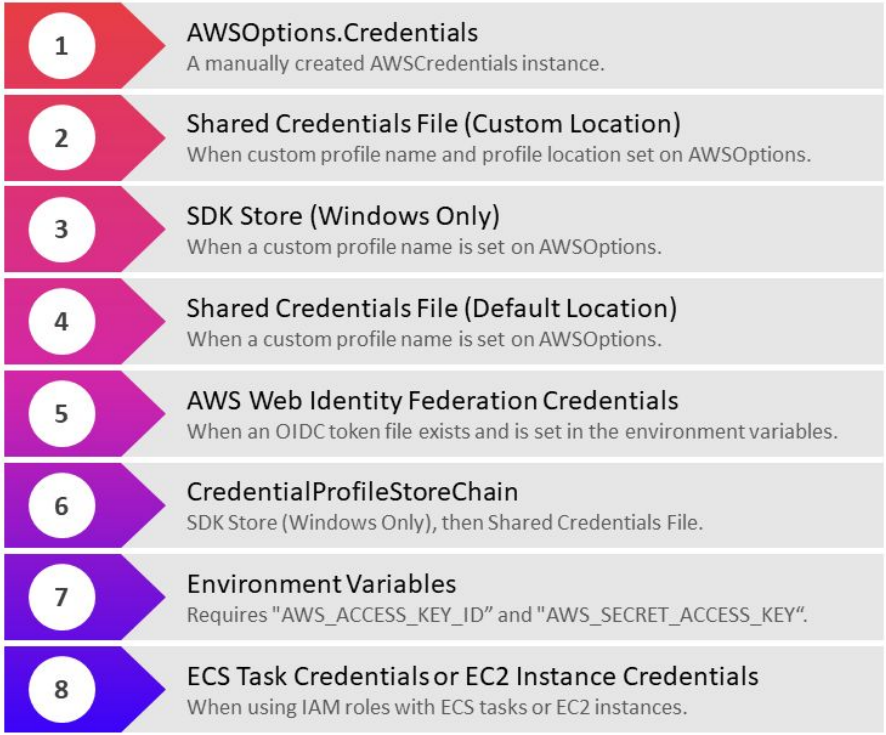


Image courtesy of Steve Gordon
https://www.stevejgordon.co.uk/credential-loading-and-the-aws-sdk-for-dotnet-deep-dive

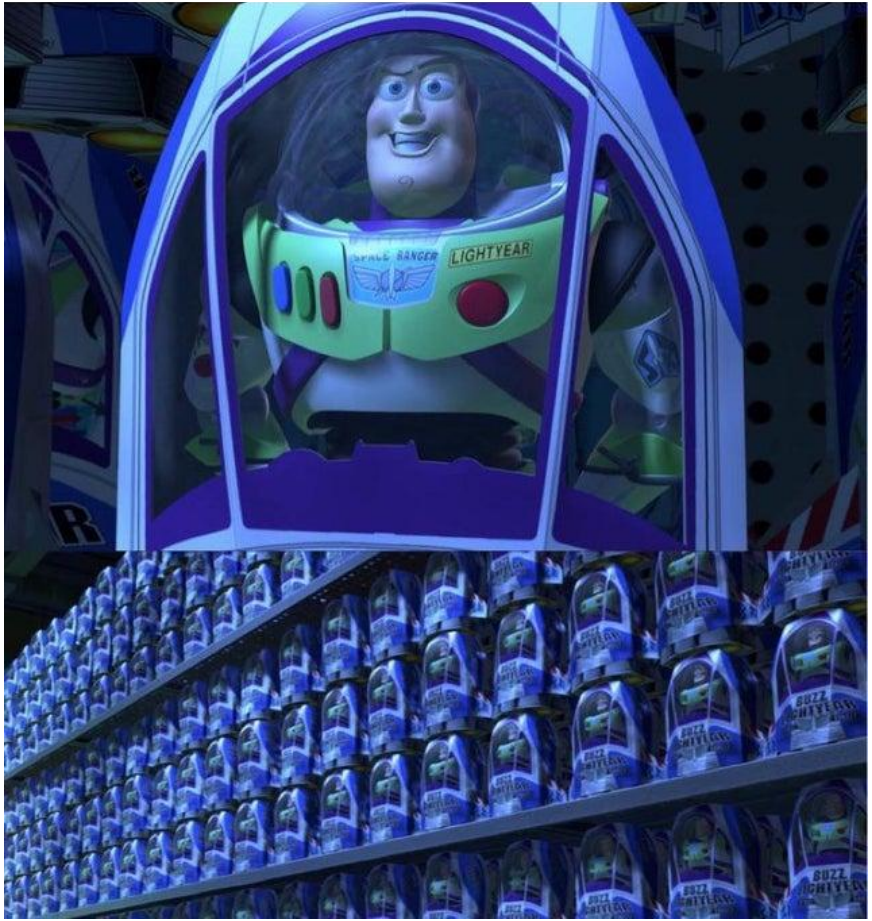
Not everything needs to be admin

Scoped roles are good policy and prevent you from contributing to someone's bitcoin wallet.

Multi-account

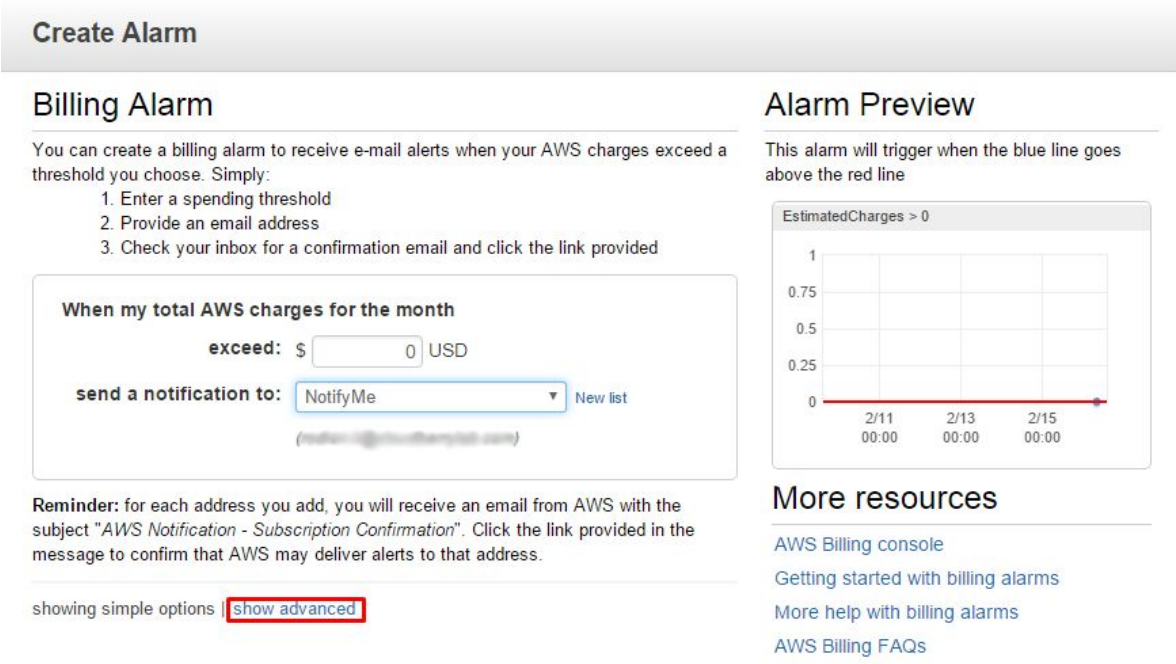
AWS best practices recommend using Accounts as a strong boundary to define service domains.

Environments should live is separate Accounts.



Budgets

Set a budget alert to let you know when exceed your comfort threshold. Then, TAKE ACTION.



Why Is It So Expensive?

You're paying for scaled operations. AWS is far better at replacing failed hard drives than you are.

Know Your Billing Dimensions

Application Load Balancers sound simple on the surface, but billing is super complicated.

Google:
“aws [service] pricing” to find out how a given service is billed before you put it into production.

For Application Load Balancers in the AWS Region:

- $0.0225 per Application Load Balancer-hour (or partial hour)
- $0.008 per LCU-hour (or partial hour)
- $0.005 per hour per Trust Store Associated with Application Load Balancer when using Mutual TLS (or partial hour)

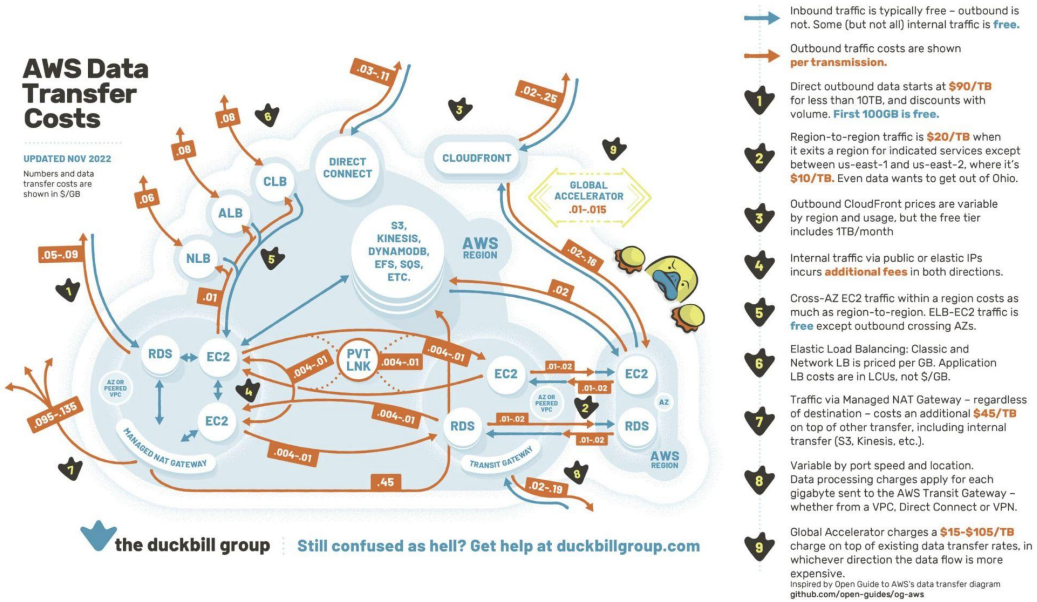
LCU Details

An LCU measures the dimensions on which the Application Load Balancer processes your traffic (averaged over an hour). The four dimensions measured are:

- **New connections:** Number of newly established connections per second. Typically, many requests are sent per connection.
- **Active connections:** Number of active connections per minute.
- **Processed bytes:** The number of bytes processed by the load balancer in GBs for HTTP(S) requests and responses.
- **Rule evaluations:** The product of the number of rules processed by your load balancer and the request rate. The first 10 processed rules are free (Rule evaluations = Request rate * (Number of rules processed - 10 free rules).

Traffic, the Silent Wallet Killer

AWS charges a nearly criminal amount for data transfer, especially out to the internet. Watch for these in high volume applications.



Free* Tier(s)

Always Free (First X of usage)

12 Months Free (new customers only)

Free Trials (time limited for new services)

COMPUTE

Free Tier — 12 MONTHS FREE

Amazon EC2

750 Hours

per month

Resizable compute capacity in the Cloud.

750 hours per month of Linux, RHEL, or SLES

STORAGE

Free Tier — 12 MONTHS FREE

Amazon S3

5 GB

of standard storage

Secure, durable, and scalable object storage infrastructure.

5 GB of Standard Storage

DATABASE

Free Tier — 12 MONTHS FREE

Amazon RDS

750 Hours

per month of database usage (applicable DB engines)

Managed Relational Database Service for MySQL, PostgreSQL, MariaDB, or SQL Server.

DATABASE

Free Tier — ALWAYS FREE

Amazon DynamoDB

25 GB

of storage

Serverless, NoSQL, fully managed database with single-digit millisecond performance at any scale.

MACHINE LEARNING — NEW

Free Tier — FREE TRIAL

Amazon SageMaker

2 Months

free trial

Machine learning for every data scientist and developer.

250 hours per month of ml.t3.medium on

COMPUTE

Free Tier — ALWAYS FREE

AWS Lambda

1 Million

free requests per month

Run code without thinking about servers or clusters

1,000,000 free requests per month

Well Architected Framework

AWS documented guidance on how to build in the cloud.

AWS Well-Architected and the Six Pillars

Framework Overview

The AWS Well-Architected Framework describes key concepts, design principles, and architectural best practices for designing and running workloads in the cloud. By answering a few foundational questions, learn how well your architecture aligns with cloud best practices and gain guidance for making improvements.

HTML | Labs

Operational Excellence Pillar

The operational excellence pillar focuses on running and monitoring systems, and continually improving processes and procedures. Key topics include automating changes, responding to events, and defining standards to manage daily operations.

HTML | Labs

Security Pillar

The security pillar focuses on protecting information and systems. Key topics include confidentiality and integrity of data, managing user permissions, and establishing controls to detect security events.

HTML | Labs

Reliability Pillar

The reliability pillar focuses on workloads performing their intended functions and how to recover quickly from failure to meet demands. Key topics include distributed system design, recovery planning, and adapting to changing requirements.

HTML | Labs

Performance Efficiency Pillar

The performance efficiency pillar focuses on structured and streamlined allocation of IT and computing resources. Key topics include selecting resource types and sizes optimized for workload requirements, monitoring performance, and maintaining efficiency as business needs evolve.

HTML | Labs

Cost Optimization Pillar

The cost optimization pillar focuses on avoiding unnecessary costs. Key topics include understanding spending over time and controlling fund allocation, selecting resources of the right type and quantity, and scaling to meet business needs without overspending.

HTML | Labs

Sustainability Pillar

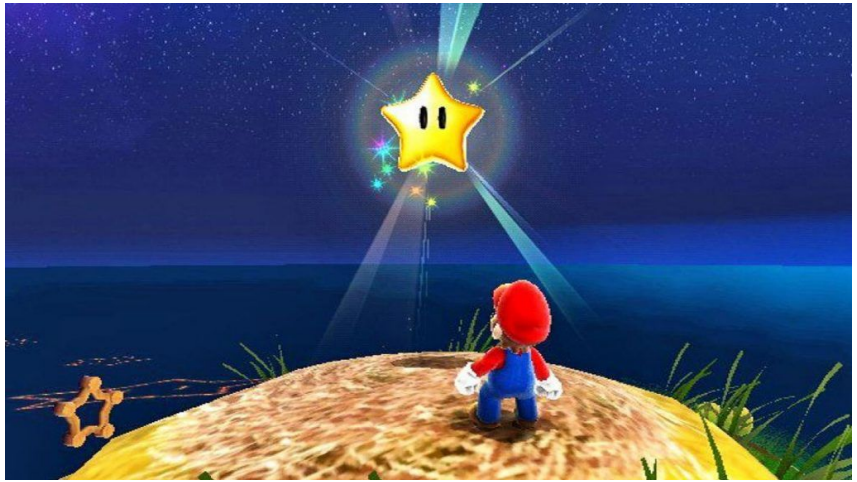
The sustainability pillar focuses on minimizing the environmental impacts of running cloud workloads. Key topics include a shared responsibility model for sustainability, understanding impact, and maximizing utilization to minimize required resources and reduce downstream impacts.

HTML | Labs

Q&A

Levi: https://www.linkedin.com/in/levimccormick/

Kisha: https://www.linkedin.com/in/kishadelain/



Slides