

AWS Architecting & Ecosystem

General guiding principles: Well Architected framework.

- 1) Stop guessing your capacity needs
- 2) Test systems at production scale
- 3) Automate to make architectural experimentation easier.
- 4) Design based on changing requirements
- 5) Drive architecture using data

Improve through game days

AWS Cloud Best Practices - Design Principles

- 1) Scalability: Vertical & Horizontal
- 2) Disposable resources: Servers should be disposable & easily configured.
- 3) Automation: Serverless, Lambda, Auto Scaling
- 4) Loose Coupling:

Monoliths are applications that do more and more over time, become bigger.
Break it down into smaller, loosely coupled components.
A change or a failure in one component should not cascade to other components.
Services → not servers. terminal 1 > WST
Don't just use EC2.
Use managed services, databases, servers etc.

- Well Architected Framework: 96 Pillars.
- 1) Operational Excellence
 - 2) Security
 - 3) Reliability
 - 4) Performance Efficiency
 - 5) Cost Optimisation
 - 6) Sustainability

I)

Operational Excellence.

Includes reliability, failover and monitor systems to deliver business availability and continually improve supporting processes and procedures. (II)

Design Principles:

Perform Operations as code - Infrastructure as Code

Annotate documentation - Automate the creation of annotated documentation after every build.

Make small, frequent, irreversible changes

Refine operations procedures frequently

Anticipate failures and process along well

Learn from all operational failures

What influenced me?

aws cloudformation AWS config

- 2) - Operate. *translates set of services*
- AWS CloudFormation *for dev, build & config*
- CloudWatch *CloudWatch Metrics, CloudWatch Logs*
- CloudWatch *CloudWatch Metrics, CloudWatch Logs*
→ Rollback → Ray *CloudWatch Metrics, CloudWatch Metrics, CloudWatch Metrics, CloudWatch Metrics*

3) - Evolve *enabling monitoring*
- CloudFormation *CloudFormation*
- CodeBuild *CodeBuild*
- CodeCommit *CodeCommit*
- CodeDeploy *CodeDeploy*
- CodePipeline *CodePipeline*

II)

Security: envelope monitoring (T)

Ability to protect information systems and assets while delivering business value through risk assessments and mitigation strategies.

Design Principles: Beginning menu

- Implement a strong identity foundation
 - Enable traceability throughout the supply chain
 - Apply security at all layers
 - Automate security best practices
 - Protect data in transit and at rest
 - Keep people away from data at storage
 - Prepare for security events like ransomware
 - Shared Responsibility Model.

1) IAM:
- IAM: Identity and Access Management
- IAM API token: token of AWS user for programmatic access
- IAM roles: roles of AWS users to utilize services
- IAM policies: policies allowing specific actions to be taken by users

2) AWS Organizations:
- Detective Controls: structure, training, configuration
- CloudTrail: tracking programmatic API calls
- CloudWatch Metrics: metrics monitoring
- Infrastructure Protection: IAM role, CloudFront, VPC, Shield, WAF
- CloudFront: delivery network
- VPC: virtual private cloud
- Shield: network protection
- WAF: web application firewall
- Inspector: security audit

3) Data Protection:

- KMS: key management service
- S3: storage service
- ELB: load balancer
- EBS: storage volume
- RDS: relational database service

4) Incident Response:

- IAM: identity and access management
- CloudFormation: template for infrastructure
- CloudWatch Events: event source for Lambda functions
- CloudWatch Metrics: metrics monitoring

- CloudWatch Logs: log storage
- CloudWatch Metrics: metrics monitoring
- CloudWatch Metrics Insights: machine learning
- CloudWatch Metrics Insights Metrics Insights

III

Reliability:

MAII

Ability of a system to recover from infrastructure or service disruptions, dynamically acquire computing resources to meet demand, and mitigate disruptions such as misconfigurations or transient network issues.

Design Principles:

Test Recovery procedures

Automatically recover from failures

Scale horizontally to increase aggregate system availability.

Stop guessing capacity

Manage change in automation.

1)

Foundations

FAW

IAM

contingent

VPC

working state

Service Quotas

EMR

Trusted Advisor

E2

2)

Change Management

ACM

Auto Scaling

ESB

CloudWatch

CDG

CloudTrail

changes transient

Config

MAI

3)

Failure Management

working state

Backups

new state

CloudFormation

S3

S3 Glacier

Route 53.

Infrastructure

are

and

choice

(c)

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

-

IV)
Characteristics
of AWS

Cost Optimisation, with automation (VI)
Includes the ability to run systems to
deliver business value at the lowest price

Low TCO

pointers to AWS best practices
Design principle: AWS Cloud Cost

Adopt a consumption model (BX)

Measure overall efficiency (TACO)

Stop spending money on datacenter operations

Analyze and attribute expenditure (SAR)

Use managed and application services
to reduce cost of ownership (Lambda)

Expenditure Awareness:

Budgets

Cost and Usage Report

Cost Explorer

Reserved Instance reporting (SAR)

Cost-Effective resources: spot instances

Spot Instances

Reserved Instances

S3 Glacier

Matching supply and demand (SAR)

Auto Scaling

Lambda

Optimizing over time.

Cloud Advisor

Cost and Usage report

workshop (1)

parallel tasks

abnormal

20%

divided

(S)

workaround (local)

optimization (S)

global

abnormal

Hosted (f)

20%

adjustment

workaround

Local

AWS Right Sizing

Process of matching instance types and sizes to your workload performance and capacity requirements at the lowest possible cost.

Scaling up is easy so always start small.

Identifying opportunities to eliminate or downsize without compromising capacity or other requirements, which results in lower costs.

It's important to right size.

Before a cloud migration and always.

Continuously after the cloud on-boarding process.

CloudWatch, Cost Explorer, Trusted Advisor, 3rd party tools can help you to monitor usage.

AWS Ecosystem - Free Resources

AWS Marketplace: virtual machine images.

Digital catalog with thousands of software listings from independent software vendors.

(3rd party) CloudWatch, CloudFront, CloudWatch Metrics.

Eg: CloudWatch Metrics, CloudFormation

Templates, SaaS, Containers.

If you buy through AWS Marketplace, it goes to your AWS bill.

Sell your own solution on the AWS Marketplace.

Setup and build workflow using CI/CD.

Integrate with Jenkins, CircleCI, Travis CI, GitHub Actions, etc.

→ **AWS Professional Services & Partner Network**
Global team of experts

- APN = AWS Partner Network.

- APN Technology Partners: Providing hardware, connectivity & software

- APN Consulting Partners: Professional services firm to help build on AWS.

- APN Training Partners: Help you learn AWS.

- AWS Competency Program: AWS Competencies are granted to APN partners who have demonstrated technical proficiency and proven customer success in specialized solution areas.

- AWS Navigate Program: Help partners become better partners.

→ **AWS Knowledge Center:**

Contains the most frequent & common questions and requests

- | | |
|----------------------------------|-------------------------------------|
| 1) Popular Services | 11) Developer Tools |
| 2) Analytics | 12) Networking & Content Delivery |
| 3) Compute | 13) Business Applications |
| 4) Customer Engagement | 14) Front-end Web & Mobile |
| 5) IoT | 15) Security, Identity & Compliance |
| 6) Management (Governance) | |
| 7) Application Integration | |
| 8) Database | |
| 9) Migration & Transfer | |
| 10) Account & Billing Management | |