System Design

Cloud Computing

Brenden west

Contents

Learning Outcomes

- Fundamentals of cloud services pricing
- AWS billing & cost management
- Cloud system design principles
- AWS Well-architected framework

Reading

- https://aws.amazon.com/architecture/well-architected/
- AWS Cloud Foundations - Module 2 - Cloud Economics & Billing
- AWS Cloud Foundations - Module 9 - Cloud Architecture

Fundamentals of cloud pricing

- **Compute** - charged per time unit & instance type
- **Storage** - typically charged per GB
- **Data transfer** - typically charged per GB,
  - Inbound is usually free
  - Data transfer between services in same region is free
  - outbound is aggregated across services
- **Utility pricing** - pay for what you use
- Volume discounts

Cost Analysis

- **Total Cost of Ownership (TCO)** - Considers total costs of building & operating a system over its lifespan. Includes capital, labor, & operational costs
- **Return on Investment (ROI)** - Considers value generated, as well as expenditure & savings

AWS Pricing Calculator

Tool to help estimate AWS services costs.

- Explore solution costs before building
- Estimate monthly costs
- Identify opportunities to reduce costs
- Find instance types and contract terms that meet your needs

AWS Billing & Cost Management

AWS Billing tool shows month-by-month usage with breakdowns by service.

Enables users to identify opportunities for cost reduction.

Can show trends & forecasts of expenditures.

AWS Budgets lets user create a budget & alerts for overages.

System Design Principles

AWS Well-Architected Framework describes core principles for good cloud system design based on 6 pillars:

- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization
- Reliability

Operational Excellence

Ability to run, monitor, & continually improve systems that deliver business value. Based on 6 key principles:

- Perform operations as code to reduce chance of human error
- Annotate documentation
- Make frequent, small, reversible changes
- Refine operations procedures frequently
- Anticipate failure & test failure scenarios
- Learn from operational events & failures

Security

Protect information, systems, & assets through risk assessments & mitigation strategies.

- Implement strong identity foundation
- Enable traceability on actions in real time
- Apply security at all layers
- Automate security best practices
- Protect data in transit & at rest
- Keep people away from data
- Prepare for security events

Reliability

Ability of system to scale on demand, recover from failures, and mitigate disruptions. Based on 5 design principles:

- Test recovery procedures
- Automatically recover from failure
- Scale horizontally to increase system availability
- Stop guessing capacity
- Manage change in automation

Performance Efficiency

Use IT resources efficiently & maintain efficiency as demand changes.

- Democratize advanced technologies
- Go global in minutes
- Use serverless architectures
- Experiment frequently
- Have mechanical empathy

Cost Optimization

Run systems as the lowest cost by controlling when $$ is spent and selecting appropriate type & number of resources.

- Adopt a consumption model
- Measure overall efficiency
- Stop spending money on data center operations
- Analyze & attribute expenditure
- Use managed & application-level services to reduce cost of ownership

Reliability & Availability

- **Reliability** - ability of system to provide functionality when needed
- **Availability** - % of time a system is operating normally (uptime). Often measured as mean time between failures (**MTBF**)

A **highly available** system can remain available with some degradation. Downtime & human intervention are minimized.

Determined by:

- Fault tolerance - built-in redundancy
- Scalability
- Recoverability