Introduction to the Well-Architected Framework

Kevin Su

Solutions Architect
Amazon Web Services
Oct. 8th, 2020

When you look at the system your team is building, can you answer the question:

“Are you Well-Architected?”

Are you Well-Architected?

Operations

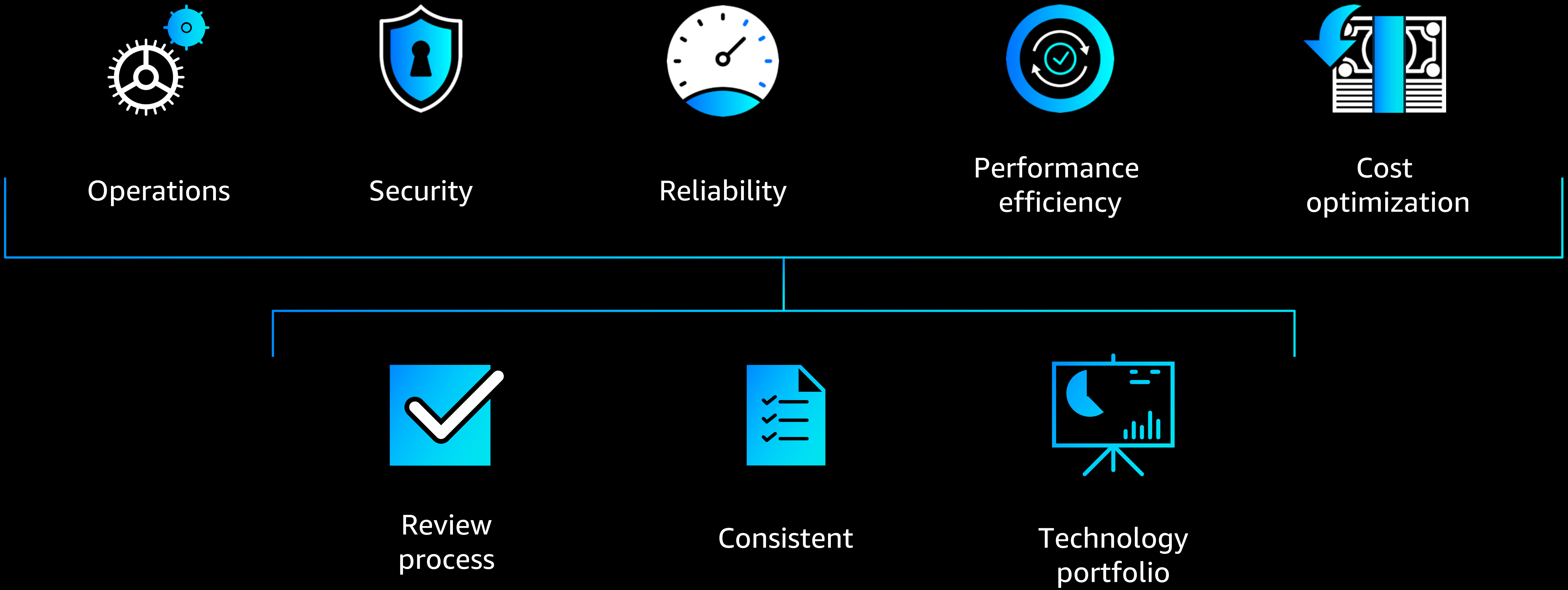
Security

Reliability

Performance efficiency

Cost optimization

Are you Well-Architected?



Operations

Security

Reliability

Performance efficiency

Cost optimization

Review process

Consistent

Technology portfolio

AWS Well-Architected

https://aws.amazon.com/well-architected/

Why AWS Well-Architected Framework?

- Build and deploy faster
- Lower or mitigate risks
- Make informed decisions
- Learn AWS best practices

A mechanism for your cloud journey

Learn

Measure

Improve

What is the AWS Well-Architected Framework?

Pillars

Design principles

Questions

Pillars of AWS Well-Architected

Operational excellence

Security

Reliability

Performance efficiency

Cost optimization

General design principles

Stop guessing your capacity needs

Test systems at production scale

Automate to make architectural experimentation easier

Allow for evolutionary architectures

Drive architectures using data

Improve through game days

Design principles for operational excellence

Perform operations as code

Annotate documentation

Make frequent, small, reversible changes

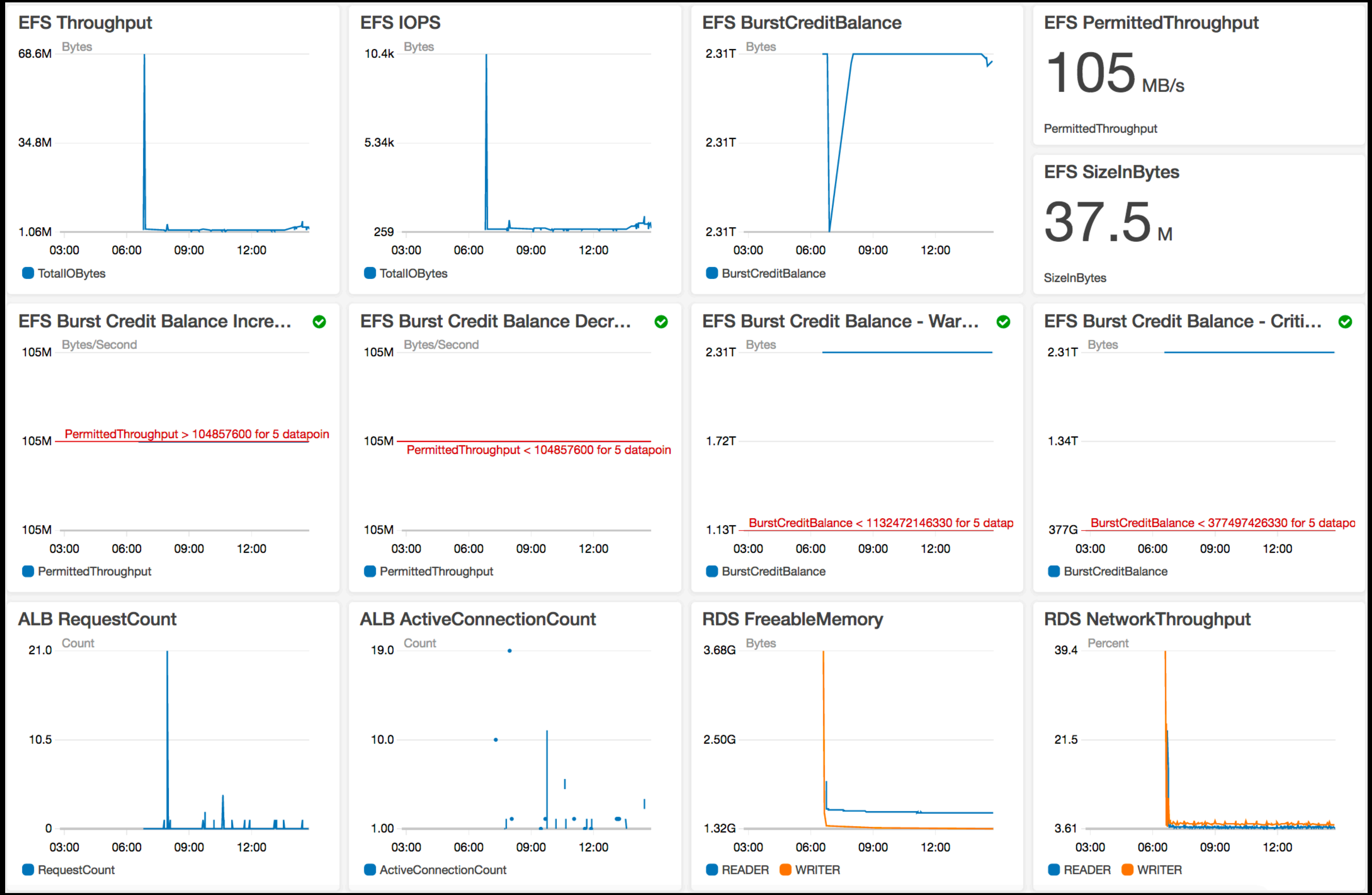
Refine operations procedures frequently

Anticipate failure

Learn from all operational failures

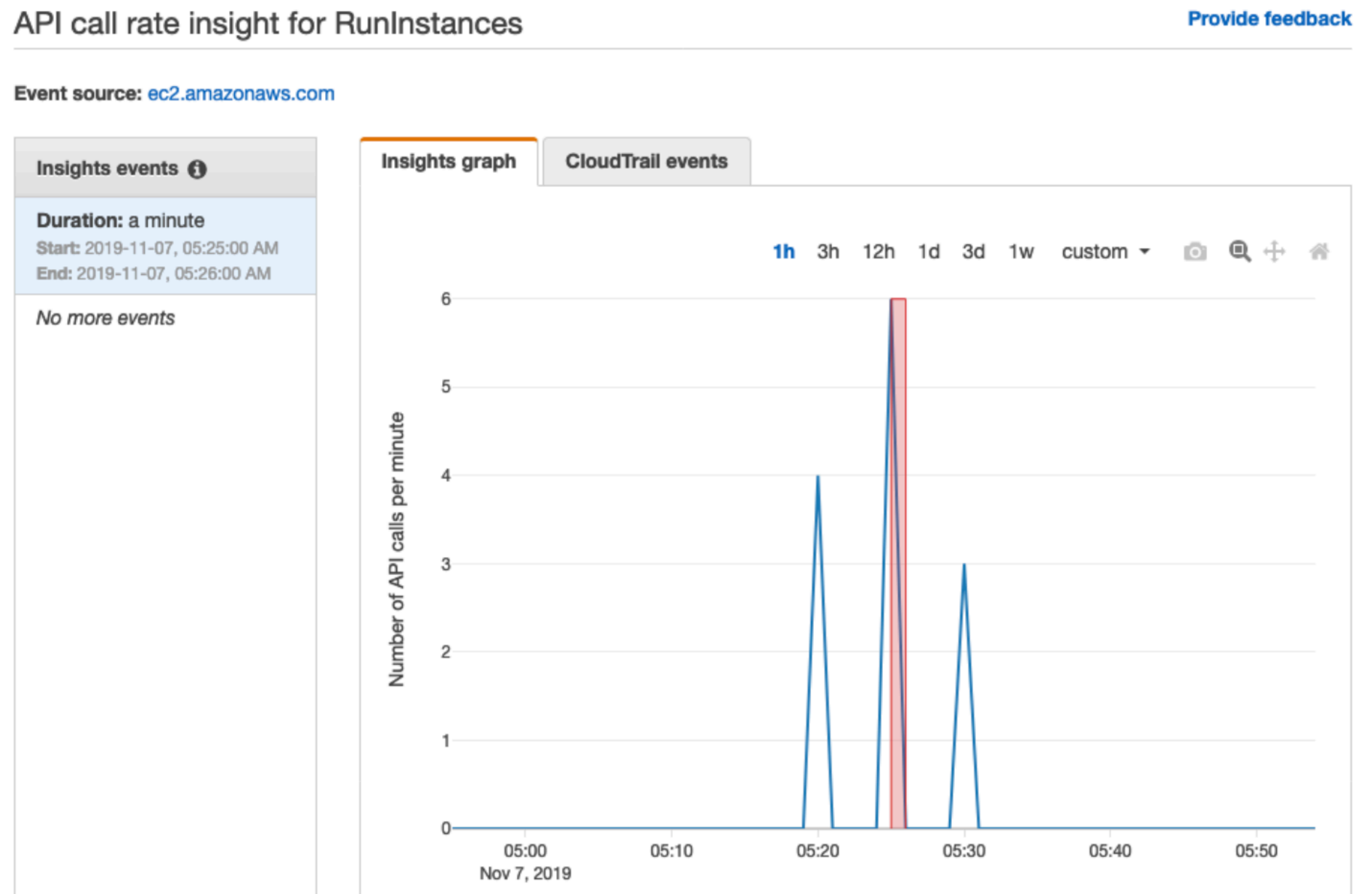
Operational excellence

- Configure a CloudWatch dashboard to monitor your resources in a single view, even across AWS regions.



Operational excellence

- Configure a persistent CloudTrail trail to be able to monitor, audit and alert on what is happening in your AWS accounts



API call rate insight for RunInstances

Event source: ec2.amazonaws.com

Insights events

Duration: a minute
Start: 2019-11-07, 05:25:00 AM
End: 2019-11-07, 05:26:00 AM

No more events

Design principles for security

- Implement a strong identity foundation
- Enable traceability
- Apply security at all layers
- Automate security best practices
- Protect data in transit and at rest
- Keep people away from data
- Prepare for security events

Security

- Switch to HTTPS with AWS Certificate Manager managed SSL/TLS certificates to encrypt customer data in transit (certificates are already provisioned in this workshop!)

- Encrypt EBS volumes to protect customer data at rest

- Enable AWS Web Application Firewall (AWS WAF) to protect your web application from known attacks (you can do it on Application Load Balancer or event better on the Amazon CloudFront distribution)

- Use Amazon GuardDuty to protect your AWS account and workloads with intelligent threat detection and continuous monitoring

Design principles for reliability

Test recovery procedures

Automatically recover from failure

Scale horizontally to increase aggregate system availability

Stop guessing capacity

Manage change in automation

Reliability

- Configure an Application Load Balancer to distribute Webserver traffic across multiple Availability Zones

- Configure Amazon EC2 Auto Scaling Group to enable auto-healing in case Webserver instances go down and to handle changing customer load

- Use Amazon CloudFront - a fast Content Distribution Network that securely delivers data to customers globally with low latency and high transfer speeds, integrating seamlessly with AWS Shield for DDoS mitigation.

Design principles for performance efficiency

Democratize advanced technologies

Go global in minutes

Use serverless architectures

Experiment more often

Mechanical sympathy

Performance efficiency

- Deploy Amazon Elastic File System to handle changes of files on Webservers
- Use Amazon CloudFront with AWS S3 as custom origin to distribute static content for lower latency for your customers and lower cost

Design principles for cost optimization

Adopt a consumption model

Measure overall efficiency

Stop spending money on data center operations

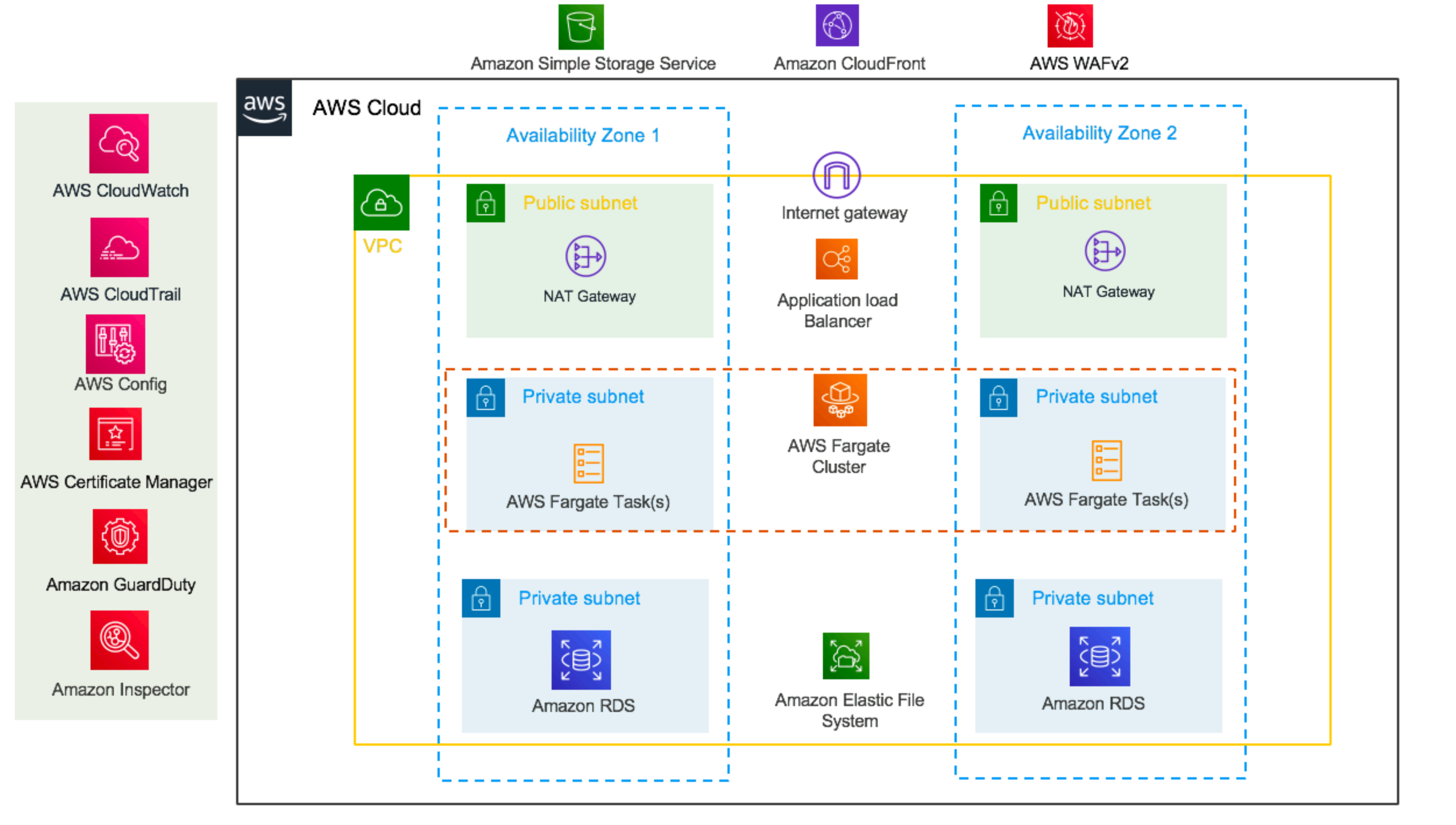
Analyze and attribute expenditure

Use managed services to reduce cost of ownership

Cost optimization

- Use Amazon EC2 Spot instances - select machine type using EC2 Spot Advisor, some instances allow for 90% savings with <5% interruption frequency
- Use the most cost-optimized machine types

Reference architecture



Getting started

https://aws.amazon.com/well-architected/

Read online resources

Account team or partner

Review first workload

Takeaways

- AWS Database Migration Service
  - Migrate your databases to AWS with minimal downtime.
  - Supports homogeneous migrations such as Oracle to Oracle, as well as heterogeneous migrations between different database platforms, such as Oracle to Amazon Aurora.

- CloudEndure
  - Conducts continuous, block-level data replication of your source machines into a staging area in your AWS account without causing downtime or impacting performance.
  - Agent-based solution, you can migrate legacy applications run on Windows Server versions 2003/2008/2012/2016/2019 and Linux distributions, such as CentOS, RHEL, OEL, SUSE, Ubuntu, and Debian.

Takeaways

- Amazon ECS + AWS Fargate
  - Fully managed container orchestration service with serverless compute engine.
  - Highly secure, reliable, and scalable way to run containers without having to manage servers or clusters of Amazon EC2 instances.

- Well-architected Framework
  - Based on five pillars — operational excellence, security, reliability, performance efficiency, and cost optimization — AWS Well-Architected provides a consistent approach for customers and partners to learn, measure, and build using architectural best practices.

Thank you!

Survey