# Well Architected Framework: Reliability

## Well Architected Framework – Reliability

The reliability pillar covers the ability of a system to recover from service or infrastructure outages/disruptions as well as the ability to dynamically acquire computing resources to meet demand.

e.g. use of chaos monkey to test recovery procedure.

### Design Principles

1. Test recovery procedures
2. Automatically recover from failure
3. Scale Horizontally to increase aggregate system availability
4. Stop estimating/guesting capacity. e.g. No under provisioning or under provisioning.

### Reliability on the cloud consists of 3 areas

1. Foundations
2. Change Management
3. Failure Management

### Foundations:

Before you start using or migrating to the cloud you need to make sure that you have the fundamental necessary infrastructure to make this work. E.g. if you have a very media heavy application, how do you plan on moving the media to the cloud.

### Change Management

Awareness how changes are deployed and the planning around the changes. Monitoring and detecting changes and the reaction to change that are impacting the system. Use services like Cloud-Watch to monitor your environments and use services like auto-scaling to automate changes

environments and use services like auto-scaling to automate changes.

### Failure Management

In the cloud, you architect systems that are designed to handle failure. In an event when a failure occurs the application owner should be aware of the failures and should have a specific action to respond to failure.

Ideally you should also have a RCA (Root Cause Analysis) to prevent such failure from happening

**Foundations:** IAM, VPC

**Change Management**: AWS Cloud Trail

**Failure Management**: Cloud Formation, RDS multi-AZ.