LTIMindtree

WHITEPAPER

Surviving the Storm:
A Comprehensive Guide to Effective Disaster Recovery in Cloud Environments

Author

Ashutosh Dixit

Technical Director, Consulting, LTIMindtree

Contents

01 Executive summary

This comprehensive whitepaper offers an in-depth analysis of the challenges and best practices associated with disaster recovery in cloud environments. It provides valuable insights into the latest tools and strategies for ensuring uninterrupted business operations during unexpected events. Covering various aspects such as data backup and recovery, business continuity planning, and disaster response, this paper equips readers with the knowledge and practical advice they need to effectively prepare for and overcome potential disruptions. Organizations can harness the power of cloud-based disaster recovery to maximize their resilience and thrive in the face of adversity. Discover the key to unlocking the full potential of cloud-based disaster recovery with this resource.

02 What is a disaster?

In the realm of traditional data center environments, we have witnessed various causes leading to application and business outages. These include power grid failures, insufficient UPS (Uninterruptible Power Supply), cable cuts, internet disruptions, and hardware crashes, among others. Such outages, particularly those that occur without warning, are universally despised. Although some of these issues may be resolved relatively quickly, many enterprises have suffered considerable damage and significant financial losses because of these failures.

Many businesses adopt cloud computing to eliminate the need for costly disaster recovery (DR) setups and to maintain physical data center infrastructure. However, in recent years, it has become evident that even cloud providers are affected by outages. For instance, AWS (Amazon Web Services) experienced various outages in the last couple of years, causing disruptions during critical sales periods. Businesses like Hulu, Adobe, Amazon.com, Roku, and Coinbase were among the affected ones.

Businesses cannot afford outages of this nature. Establishing a completely isolated DR site in a separate cloud region ensures uninterrupted operations. The concepts of high availability, fault tolerance, disaster recovery, and self-healing are increasingly intertwined when designing effective disaster mitigating and recovery plans.

Before delving into the technical aspects of setting up a DR site, it is important to review some commonly used terminologies associated with this subject.

03 Preparing for recovery

Disaster in the world of computing refers to any event that unexpectedly disrupts normal business operations. It causes either the entire application or critical components of applications to become unavailable. This disruption can be caused by various factors, such as the failure of application components, hardware malfunctions, or misconfigurations in the Hypervisor. Even power outages, outages of Availability Zones or Regions, or even natural disasters can cause disruptions. For the purpose of this whitepaper, we will define a disaster as a scenario where the entire cloud region becomes unavailable due to any reason.

A Disaster Recovery Region is a predefined cloud region specifically designated as a backup location where infrastructure can be hosted in the event of a disaster. It is typically located a considerable distance away from the primary region to avoid the impact of any disaster that strikes this region. To comply with regulations and minimize latency, it is recommended that the disaster recovery region be located reasonably close to the primary region. In some cases, it may even be within the territorial boundaries of the country.

Recovery Time Objective (RTO), as defined in the service level agreement (SLA), is the time frame for an application's recovery from a disaster. This time is utilized to bring up and operationalize applications at the disaster recovery region using backups and to redirect traffic to this region.
Recovery Point Objective (RPO) is the maximum allowable data loss, measured in minutes, as agreed upon in the SLA, in the event of a disaster. This can be minimized by implementing frequent backups and synchronization. Ideally, every application owner prefers a zero RPO, which means no loss of committed changes during application recovery from a disaster in another region.

Every organization possesses a unique array of applications, each with its own distinct priorities. Some applications may not require constant availability and can tolerate extended periods of downtime. There are others so crucial to the business that even a few minutes of outage could lead to significant financial losses. While data loss in a development environment may not pose a major issue, the production environment cannot afford any loss of financial data. Therefore, the configuration of disaster recovery measures should be tailored to suit the specific needs of each application class.

04 Types of disaster recovery setups

Although there is no universal solution, there are four widely recognized categories of disaster recovery setups that can effectively accommodate most applications.

Backup and Restore

Pilot Light

Warm Standby

Active-Active Multi-Site Stand by

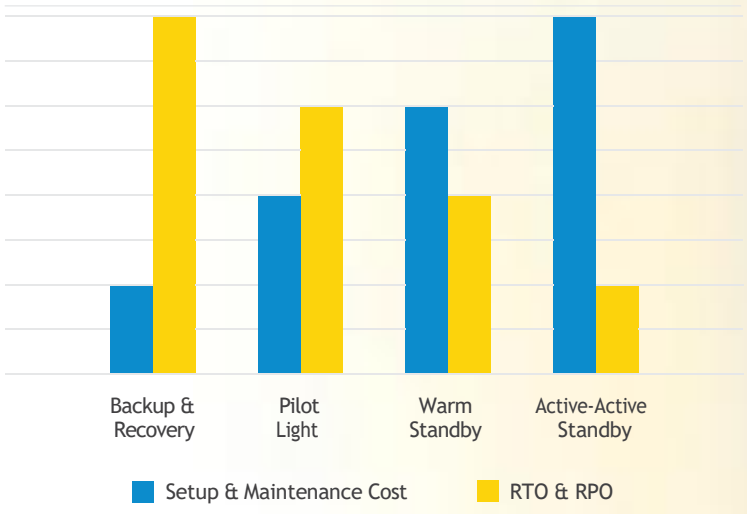

Figure 1: Cost vs. Resiliency

As you progress through these setups, from top to bottom, the cost and complexity of implementing disaster recovery measures increase. However, the benefits of reduced RTO and RPO make it a worthwhile investment. The key, now, is to determine the appropriate disaster recovery model that aligns with your specific business and application needs.

While the setup described in this whitepaper can be implemented on any cloud platform, AWS icons, and services have been used for context. Equivalent services offered by other cloud providers can be utilized to replicate a similar setup within their respective environments.

Backup and restore

This form of disaster recovery planning is the most economical and suitable for projects that are non-critical or related to production. They can afford a few hours of outage and accept some degree of data loss. This approach is applicable to applications where data changes infrequently or is frequently synchronized with other locations.

Plan of action

In this setup, the various application components are backed up during predetermined time intervals. The components could be application data, databases, media files, and documents. A copy of these backups is stored in a separate region known as the disaster recovery (DR) region. At this stage, the DR region does not host the actual application. However, the customer is responsible for preparing the landing zone, including the cloud account, virtual private cloud (VPC), security configurations, etc. In the event of a disaster, the application stack can be restored using these backups.

Recovery Time Objective (RTO)

Time taken to spin off a copy of production resources in the DR region and time taken to restore the backup into those resources

Recovery Point Objective (RPO)

Changes made since the last backup.

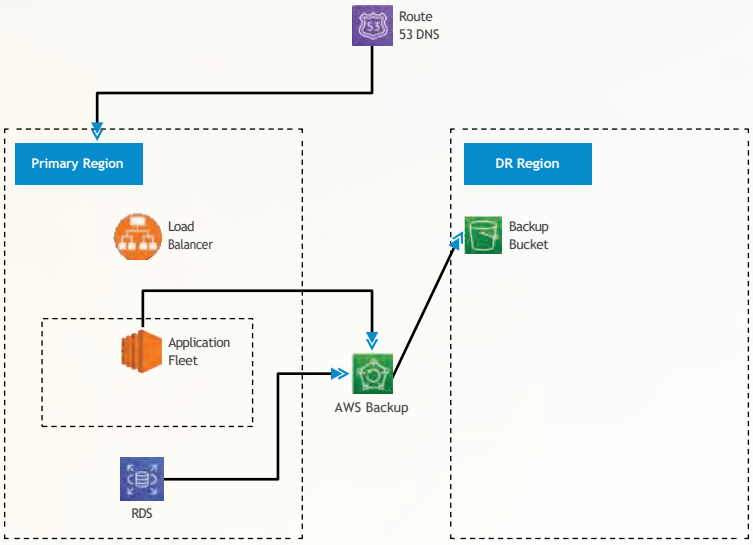

Figure 2: Backup & Recovery

This represents the most basic form of disaster recovery setup, requiring minimal cost and pre-configuration. While the process itself is straightforward, it is susceptible to human error. Even with manual DR drills in place, it has been observed that certain steps may be overlooked during an actual disaster. It is highly recommended that a comprehensive runbook detailing the execution procedure be included.

Use cases

This setup can be applied to a wide range of development environments or reporting applications that operate for a limited duration each month. It is particularly suitable for low-budget and low-priority applications.

Warm standby setup

In the Warm Standby approach, the application stack operates in the primary region, while the landing zone for the application remains on standby in the disaster recovery (DR) region. Continuous backup and synchronization mechanisms are established between the two regions. Automation scripts using tools like Terraform or CloudFormation (in AWS) are readily available to swiftly create the application infrastructure and associated resources in the event of a disaster. Additionally, another segment of the automation script is configured to retrieve the most recent backup copy and synchronize resources to ensure they are up to date.

Plan of action

In the event of a disaster, the automation script takes charge of provisioning all resource components and restoring the application and database. To direct traffic to the new DR region, DNS records can be altered either manually or automatically.

Recovery Time Objective (RTO)

Time taken to execute automation scripts and restore infrastructure on the DR site

Recovery Point Objective (RPO)

The gap in copied data since the last sync, typically, less than 5 mins for change data in MBs

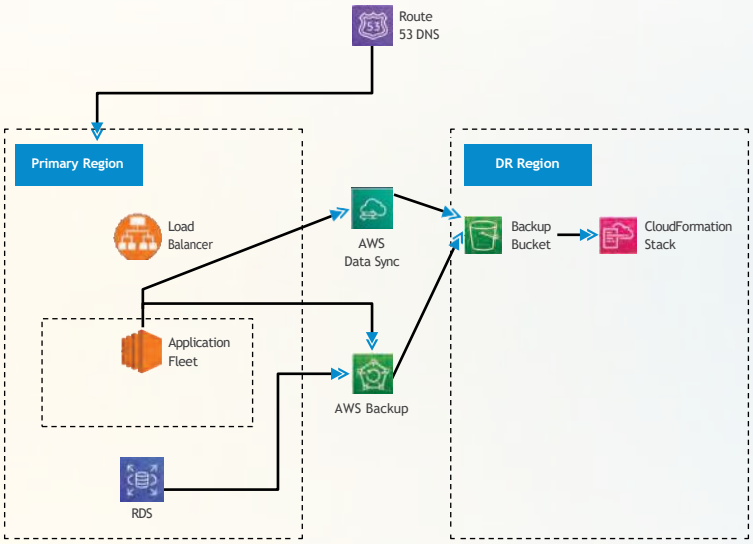

Figure 3: Warm & Standby

This represents a widely adopted setup that strikes a balance between cost and performance. While the application may experience brief outages of up to 15 minutes, no additional cost is incurred by the customer as the disaster recovery (DR) site remains inactive until a disaster occurs. Native cloud Platform as a Service (PaaS) applications, such as Relational Database Service (RDS) and DynamoDB, can synchronize data across regions in real time, ensuring minimal data loss (Recovery Point Objective - RPO). The cost of implementing a warm standby setup may be slightly higher than a backup, but the process of setting up automation scripts and thoroughly testing them requires diligent effort. If application components are modified at the resource level, corresponding updates must be made to these scripts as well. Regular mock drills should be conducted to ensure seamless restoration. A single misstep in terms of synchronization can cause significant concerns.

Use cases

This setup is suitable for applications with high priority but not deemed mission critical. These typically include applications where the data does not involve banking transactions or where the latest committed data can easily be reproduced by requesting users to resubmit the information. Examples of such applications include HR portals, tax computation portals, e-learning platforms, timesheet applications, expense portals, and more.

Pilot light backup

The term "Pilot light" refers to a setup in which a small burner of a gas stove is kept permanently lit to ignite the larger burner when necessary. This arrangement ensures that things stay warm and avoids last-minute hassles. The concept of the Pilot Light disaster recovery (DR) setup is based on this idea.

Setup

The entire fleet of applications operates in the primary region as usual, while a scaled-down version of the same setup is maintained in the DR region. In the DR region, smaller EC2 and RDS instances are utilized. Alongside regular scheduled backups, live cloud-native or third-party services are used to synchronize data between the two environments.

Plan of action

In the event of a disaster, the infrastructure in the DR region is scaled up to full capacity to match the primary site. DNS re-routing is then implemented to direct traffic to the application infrastructure in the DR region.

Recovery Time Objective (RTO)

The time taken to upscale the infrastructure and re-route the DNS requests to the DR region

Recovery Point Objective (RPO)

The gap in copied data since the last sync, which is less than 5 minutes for change data in MBs

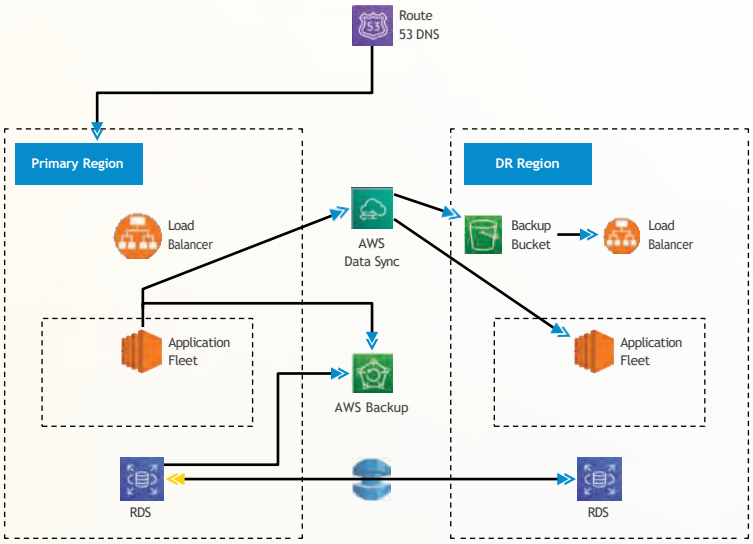

Figure 4: Pilot light backup

This setup represents an elevated version of the warm standby approach, offering significant advantages over the previous process. As the application is already operational in the disaster recovery (DR) region, testers have the opportunity to verify the stability of the DR infrastructure. Customers are responsible for the cost and maintenance of the additional infrastructure. Furthermore, application owners can periodically redirect a small portion of traffic to test the reliability of the infrastructure.

Use cases

This setup is suitable for high-priority applications with resources that require frequent updates and have a limited budget for disaster recovery.

Multi-site active-active standby

Setup

This is the ideal configuration for mission-critical applications that require zero downtime. The entire application environment in the primary region, including the landing zone, is replicated in the disaster recovery (DR) region. Both the primary and DR regions are active and capable of serving customers, with live synchronization established between them using a combination of native and third-party tools.

Plan of action

In the event of a disaster in the primary region, a routing policy automatically detects the failure and redirects new user requests to the DR region. While some users may experience a brief interruption in their connection for a few minutes, the overall transition to the DR region is expected to be seamless.

Recovery Time Objective (RTO)

Time required for routing policy to declare master region (Infrastructure) unhealthy and switch over to DR region, which is typically less than 1 minute

Recovery Point Objective (RPO)

The gap of copied data since the last sync. Less than 2 mins for change data in MBs.

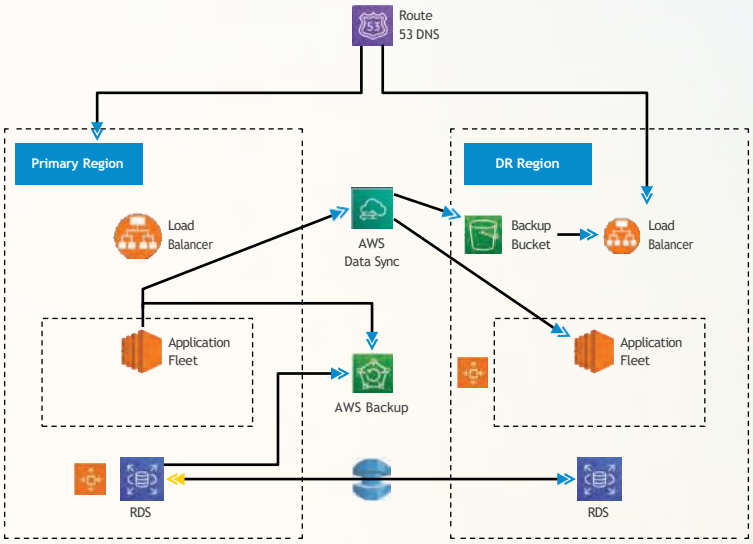

Figure 5: Multi-site active-active standby

This setup represents the most robust yet costly option available. Customers will need to allocate double the cost for hosting and maintaining the infrastructure. Developers must ensure that changes are implemented through automation scripts and that updates are applied to both sides of the infrastructure. When properly implemented and maintained, this setup can prevent severe application outages caused by regional failures without human intervention. By configuring crucial components in maximum protection mode, data can be replicated to the DR region before users receive confirmation that their data has been saved. This ensures zero data loss with a zero RPO. However, the drawback of this additional setup is that users may experience a delay before their data is successfully saved.

Use cases

This setup is suitable for business-critical applications with the highest priority and demand for maximum availability. Examples include payment systems, live tracking systems, order delivery systems, e-commerce websites, and more.

05 Conclusion

As you consider the available options for your disaster recovery setup, it is essential to clearly define expectations regarding RTO, RPO, application criticality, and budget. Architects play a crucial role in translating these requirements into suitable DR models that align with the specific needs of each application. It is not uncommon for application owners to agree on a hybrid DR setup that incorporates elements from multiple models, allowing for customization and creating a comprehensive framework.

Implementing a thorough strategy for DR, including careful planning, effective implementation, rigorous testing, and ongoing maintenance, can safeguard companies from significant setbacks and embarrassing incidents. The loss of an application can result in substantial business losses, and disappointed customers may never return. During application downtime, customers often seek alternatives, particularly in the e-commerce sector. I vividly recall an incident last year when the payment gateway of the Apple website was down for two hours, preventing me from completing my purchase of a new iPhone. Consequently, I turned to another e-commerce platform, Tata Cliq, which provided exceptional service. Ever since that experience, I have continued relying on Tata Cliq for my subsequent purchases, regardless of the occasional outages on Apple's website. Unless I encounter any negative experiences with Tata Cliq, I see no reason to return to the Apple website for future purchases.

In a nutshell, by carefully considering the DR options available, aligning them with business requirements, and implementing a robust DR strategy, companies can mitigate risks, protect their applications, and maintain the trust and satisfaction of their customers.

About the Author

Ashutosh Dixit
Technical Director, Consulting, LTIMindtree

For almost a decade and a half, Ashutosh has worked with clients across retail, educational services, banking, telecom, and FMCG sectors. As a Technical Director at LTIMindtree, he heads strategy roadmap assignments to solve modern business problems with the help of state-of-the-art IT technologies and tools. He helps customers on their journey towards digital transformation to the cloud, modernization of IT assets, tuning the cost of running IT Infrastructure (FinOps), and up the path to data-driven decisions, which are a few of our specialties.