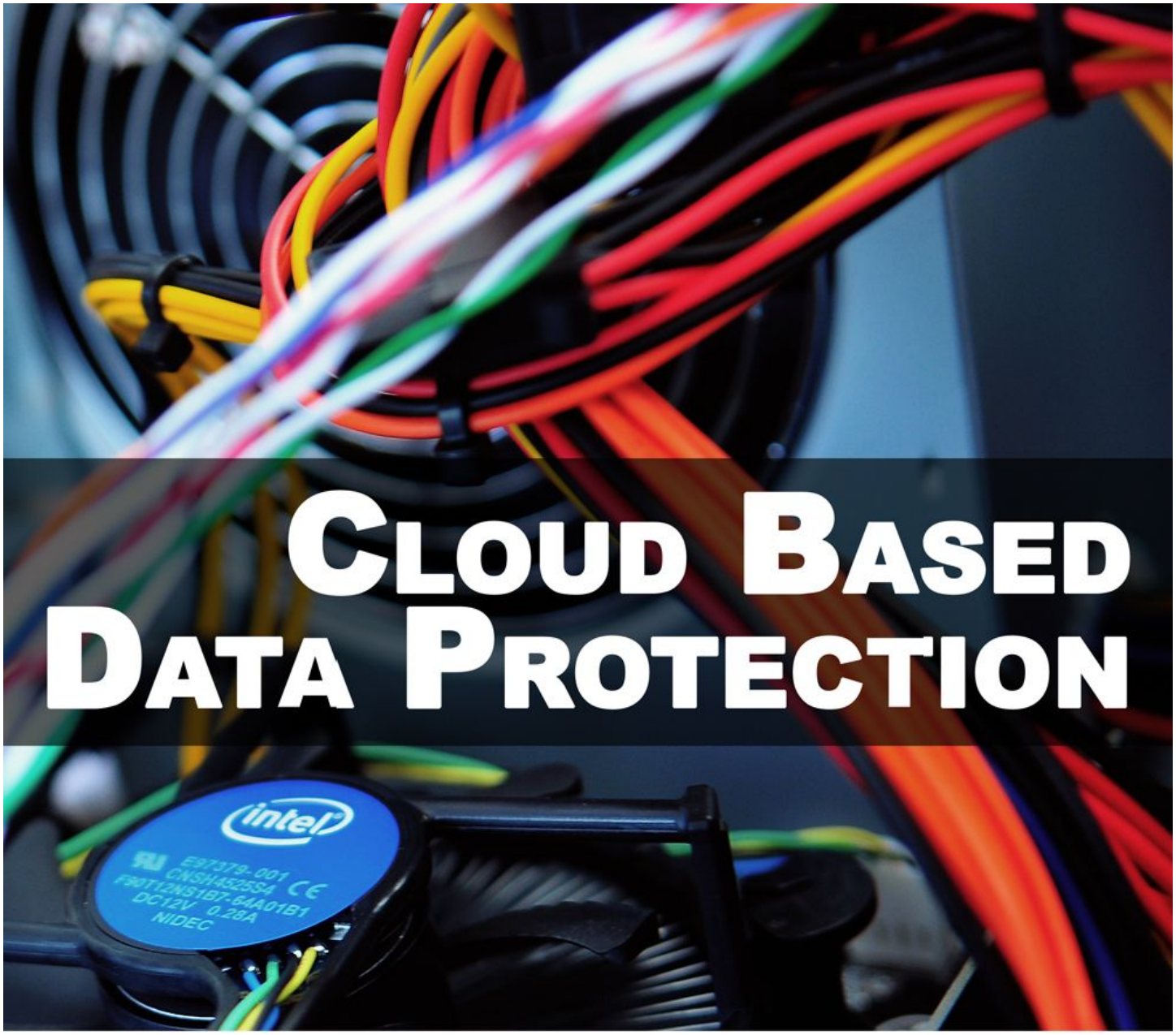


Cloud Based Data Protection

Back-up and Disaster Recovery
WHITEPAPER

May 2016

Infrastructure and Data Risk Management

AximCloud
Cloud Solutions for Business

TABLE OF CONTENTS

Introduction

Disaster Recovery entails a set of procedures and policies that enables a business to continue to function and recover following a disaster, either natural or man-made. Any event that has a negative impact on a business' continuity or finances could be termed a disaster. These events include hardware or software failure, a network outage, a power outage, physical damage to a building like fire or flooding, human error, or some other significant disaster.

For Disaster Recovery, proper preparation is a necessity. This paper summarizes some of the best practices to improve Backup and Disaster Recovery plans and processes in AWS.

As business and systems evolve, a business's Disaster Recovery must evolve as well. It is a continual process of analysis and improvement. For each service provided and available, businesses must establish an acceptable recovery point and time, and then build an appropriate DR solution.

Recovery Time Objective and Recovery Point Objective:

Recovery time objective (RTO) - This is the duration of time and the service level to which a business process must be restored after a disaster (or disruption) in order to avoid unacceptable consequences associated with a break in business continuity. For example, if a disaster occurs at 9:00 am and the RTO is 6 hours, the expected time for the DR process to bring the business service back up to an acceptable service level would be by 3:00 pm.

Recovery point objective (RPO) - This describes the tolerable amount of data loss measured in time. For example, if the RPO was 1 hour and a disaster occurs at 2 pm, after the system was restored, data from 1 pm would available.

The decision on an accepatble RTO and RPO depends on a variety of factors, with the most important being the financial impact to the business while their systems are down. The financial impact is typically assessed by considering many factors such as the loss of business and damage to its reputation due to downtime and the lack of systems availability.

Once the decision is made, then the IT organization has a clear path to develop the most cost-effective method to restore the system in case of disaster based on the RPO and RTO.

AWS Services and Features Essential for Disaster Recovery

To implement a suitable backup and disaster recovery strategy, it is important to review the AWS services and features that are the most relevant to disaster recovery. The following section provides a summary of AWS features and services.

When in the preparation phase of DR, it is critical to consider the use of services and features that support data migration and durable storage because they enable you to restore backed up critical data to AWS when disaster strikes. If a business's DR plans consists of either a scaled-down or a fully-scaled deployment of the business's system in AWS, compute resources will be necessary as well.

When in the process of reacting to a disaster, it will be necessary to either quickly provision compute resources to run your system in AWS regions or to switch over to already running resources in AWS. The critical AWS infrastructure pieces here include DNS, networking features, and various Amazon Elastic Compute Cloud (Amazon EC2) features described below.

Regions

Amazon Web Services are available in multiple geographic regions, so you can choose the most appropriate location for your recovery site, in addition to the site where your system is fully deployed. At the time of writing, AWS is available in eight regions: US East (North Virginia), US West (Oregon), US West (North California), EU (Ireland), Asia Pacific (Singapore), Asia Pacific (Tokyo), Asia Pacific (Sydney) and South America (Sao Paulo).

Storage

Amazon Simple Storage Service (Amazon S3) provides a highly durable storage infrastructure designed for mission-critical and primary data storage. Objects are redundantly stored on multiple devices across multiple facilities. AWS provides further protection for data retention and archiving via Versioning in Amazon S3, AWS Multi-Factor Authentication, bucket policies, and ***Identity and Access Management*** (IAM).

Amazon Elastic Block Store (Amazon EBS) provides the ability to create point-in-time snapshots of data volumes. Snapshots can be used as the starting point for new Amazon EBS volumes, and to protect data for long-term durability. Once a volume is created, it is then attached to a running Amazon EC2 instance. Amazon EBS volumes provide off-instance storage that persists independently from the life of an instance.

Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. In order to keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. With Amazon Glacier, customers can reliably store large or small amounts of data for as little as $0.01 per gigabyte per month, a significant savings compared to on-premises solutions.

AWS Import/Export accelerates the moving of large amounts of data into and out of AWS using portable storage devices for transport. AWS transfers your data directly onto and off of storage devices by using Amazon's high-speed internal network and bypassing the Internet. For data sets of significant size, AWS Import/Export is often faster than Internet transfer and more cost effective than upgrading your connectivity. You can use AWS Import/Export to migrate data into and out of Amazon S3 buckets or into Amazon EBS snapshots.

AWS Storage Gateway enables seamless migration of data to and from AWS's cloud storage and on-premises applications. AWS Storage Gateway stores volume data locally in your infrastructure and in AWS. This enables existing on-premises applications to seamlessly store data in the cost-effective, secure, and durable storage infrastructure of AWS while preserving low-latency access to this data.

Compute

Amazon Elastic Compute Cloud (Amazon EC2) provides resizable compute capacity in the cloud. Within minutes you can create EC2 instances which are virtual machines over which you have complete control. In the context of DR, this ability to rapidly create virtual machines that you can control is critical. To describe every feature of Amazon EC2 is outside the scope of this document and we have only focused on the aspects of Amazon EC2 that are most relevant to DR.

Amazon Machine Images (AMIs) are preconfigured with operating systems and some preconfigured AMIs may also include application stacks. You can also configure your own AMIs. In the context of DR, we strongly recommended that you have your own AMIs configured and identified so that they can launch as part of your recovery procedure. Such AMIs should be preconfigured with your operating system of choice plus appropriate pieces of the application stack.

Amazon EC2 Reserved Instances are often used to receive a significant discount on the cost of running an EC2 instance and have advantages that are especially relevant to DR. Reserved Instances help to ensure that the capacity you need is available to you when required.

Availability Zones are distinct locations that are engineered to be insulated from failures in other Availability Zones and provide inexpensive, low latency network connectivity to other Availability

Zones in the same Region. By launching instances in separate Availability Zones you can protect your applications from the failure of a single location. Regions consist of one or more Availability Zones.

Amazon EC2 VM Import enables you to import virtual machine images from your existing environment to Amazon EC2 instances.

Networking

When dealing with a disaster, it's very likely that you will have to modify network settings as you are failing over to another site.

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost-effective way to route end users to Internet applications.

Elastic IP Addresses are static IP addresses designed for dynamic cloud computing. Unlike traditional static IP addresses, Elastic IP addresses enable you to mask instance or Availability Zone failures by programmatically remapping your public IP addresses to instances in your account in a particular region. For DR, you can also pre-allocate some IP addresses for the most critical systems so that their IP addresses are already known before disaster strikes. This can simplify the execution of the DR plan.

Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve even greater fault tolerance in your applications by seamlessly providing the amount of load balancing capacity needed in response to incoming application traffic. Just as you can pre-allocate Elastic IP addresses, you can pre-allocate your Elastic Load Balancer so that its DNS name is already known, which can simplify the execution of your DR plan.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a private, isolated section of the Amazon Web Services cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. This would enable you to create a VPN connection between your corporate datacenter and your VPC to leverage the AWS cloud as an extension of your corporate datacenter. In the context of DR, you can use Amazon VPC to extend your existing network topology to the cloud. This can be especially appropriate when recovering enterprise applications that are typically on the internal network.

Amazon Direct Connect makes it easy to set up a dedicated network connection from your premise to AWS. In many cases, this can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections.

Databases

For your database needs, consider using these AWS services:

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. You can use Amazon RDS either in the preparation phase for DR to hold your critical data in a running database already and/or in the recovery phase to run your production database.

Amazon SimpleDB is a highly available, flexible, non-relational data store that offloads the work of database administration. It can also be used in the preparation and the recovery phase of DR.

Redshift is a fast, fully-managed petabyte-scaled data warehouse service.

You can also install and run your choice of database software on Amazon EC2 and can chose from a variety of leading database systems.

Deployment Orchestration

Deployment automation and post-startup software installation/configuration processes and tools can be used in Amazon EC2 as well as other AWS resources. Investments in this area are highly recommended. These can prove useful in the recovery phase to create the required set of resources in an automated fashion.

AWS CloudFormation gives developers and systems administrators an easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. You can create templates for your environments and deploy associated collections of resources, called a stack, as needed.

Example Disaster Recovery Scenarios with AWS

This section explores different DR scenarios that highlight the use of AWS resources and compare AWS with traditional Disaster Recovery methods:

- AMI backup and restore
- Pilot Light for Simple Recovery in AWS
- Warm Standby Solution
- Multi-site Solution

Amazon Web Services enables businesses to cost effectively operate each of these example DR strategies. It's important to note that these are just examples of possible approaches, and variations and combinations of these are feasible.

EBS Backup and Restore

This method stores the data for an AWS instance in an EBS Snapshot. The Snapshot can be used to recover data and be attached to an EC2 instance. This method makes recovery processes efficient and repeatable.

A backup script can be written to automate this process. The backup script can create snapshots volumes and copy across AWS Regions. AWS EBS volumes are then created from these snapshots. Recovery time in this scenario tend to be the longest.

The backup of data is only part of the process. Recovery of data in a disaster scenario needs to be done quickly and reliably. It should ensure that business service infrastructure is configured for the appropriate retention and security of data. The data recovery processes must be tested thoroughly.

Key steps for EBS backup and restore:

- Develop an appropriate backup script to create a snapshots and copy the necessary data across AWS regions.
- Ensure appropriate retention policy for this data.
- Ensure that appropriate security measures are in place for this data, including encryption and access policies.
- Regularly test the recovery of this data and restoration of your system.

Pilot Light for Quick Recovery into AWS

The pilot light concept comes from a gas furnace. A pilot light is a small flame that always remains burning. At any time, the gas can be turned up and the furnace turns on and heats the house. So like that gas furnace, the critical components of your system are already provisioned and configured in AWS. When a disaster occurs, the system is rapidly provisioned to provide a full-scale environment with the critical components as it's core. The other systems can be brought online through the use of techniques described in EBS backup and restore.

The critical components that comprise the core of the pilot light are most likely database servers which receive replicated data from the business's primary datacenter - either a rack metal datacenter or another region in AWS. There may also be other forms of data that need to be replicated such as FTP files. This is the system core around which the other parts of the infrastructure can be created and configured quickly in AWS to restore the system.

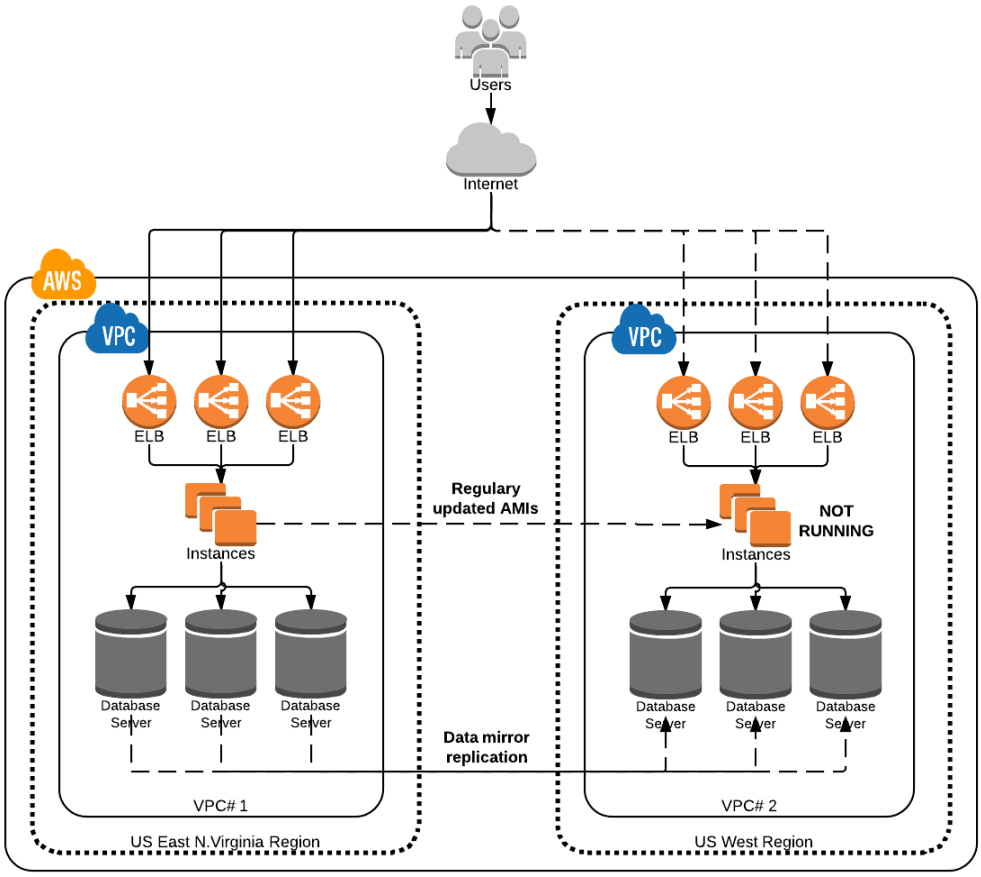
One of the recommended ways to quickly restore the non-critical (non-core) parts of the infrastructure is with Amazon Machine Images (AMIs). AMIs are images of entire servers, such as a web server. These AMIs should be provisioned beforehand, either from another AWS region or created from scratch from the business's servers. Elastic IP Addresses can be associated with the newly provisioned instances or Elastic Load Balancers can be used to balance and route traffic to multiple instances. Once the infrastructure is completely provisioned, updating the DNS records for the application is required.

Configuration information and installation packages for peripheral systems can also be stored in AWS as an EBS Snapshot. Using the EBS Snapshot to create multiple volumes in multiple Availability zones can allow the application server setup to be done quickly and efficiently. Once the instances are provisioned, they can be installed and configured.

The Pilot Light method will allow for a quicker Recovery Time than the "EBS Backup and Restore" scenario. This is due to the critical parts of the system are already running and are continually replicating data and otherwise being kept up to date. There are still some installation and configuration tasks to fully recover the applications. AWS allows for the automation of the provisioning and configuration of the infrastructure resources which can be a significant benefit to save time and help protect against human errors.

Preparation Phase:

The following figure shows an example of the Pilot Light. In the example, the database servers in the DR environment are running and receiving replicated data from the primary datacenter (in this example, another Region in AWS). The preparation phase will provision the database servers and configuring replication. It will also involve keeping the instance AMIs up to date.



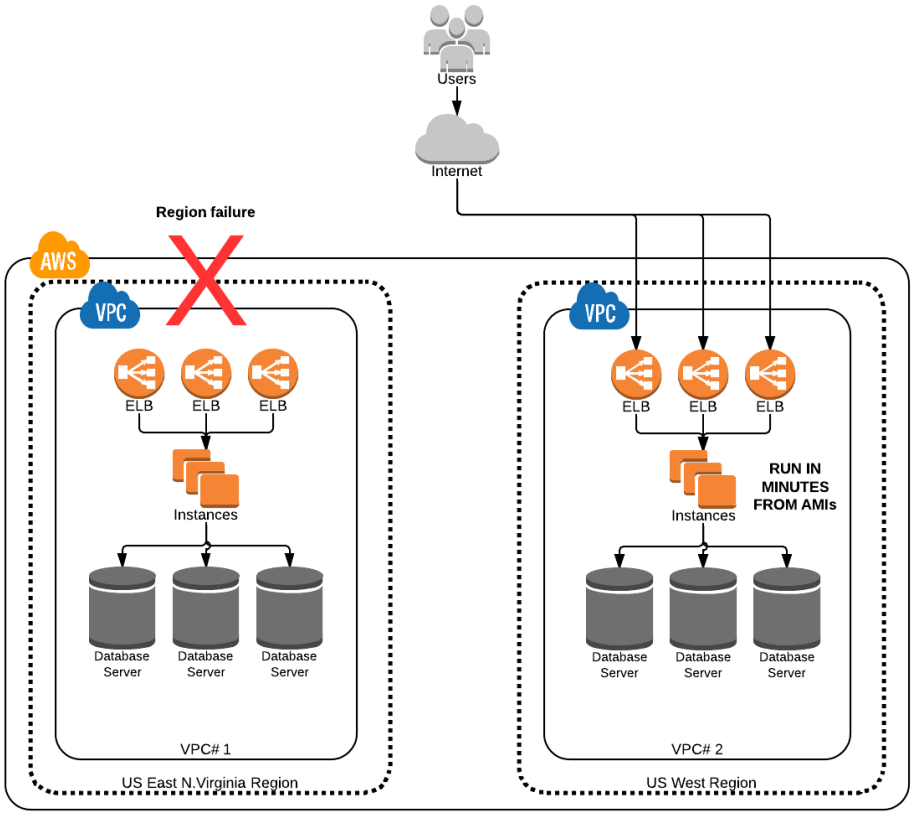
Key points for preparation:

- Setup EC2 instances/RDS to replicate or mirror data.
- Ensure that supporting custom software packages are available in region hosting the DR environment.
- Create and keep up to date Amazon Machine Images (AMI) of key servers where fast recovery is required (such as web servers).
- Regularly run these servers, test them, and apply any software updates and configuration changes.
- Consider automating the provisioning of AWS resources.

Recovery Phase:

Once a disaster occurs and the recovery phase in a pilot light environment begins, the non-core of the system must be restored. Amazon Machine images (AMIs) can be started in minutes. Additional capacity or an increase in size is possible for your data servers. Although horizontal scaling is the most cost-efficient, it may not always be the easiest. Vertical scaling is more easily accomplished through a resize of the EC2 or RDS instance. Any required DNS updates should be done at the same time.

Once the new DR environment is up and running, the next step should be to restore redundancy as soon as possible. Although it is unlikely that the DR environment will fail, it is a possibility.



Key points for recovery:

- Use custom AMIs to provision application EC2 instances.
- Resize and/or scale any database/data store instances, where necessary.
- Change DNS to point at the EC2 servers/Elastic Load Balancer.
- Install and configure any non-AMI based systems, ideally in an automated fashion.
- Consider a new DR environment, in the unlikely case the current DR environment goes down.

Warm Standby Solution in AWS

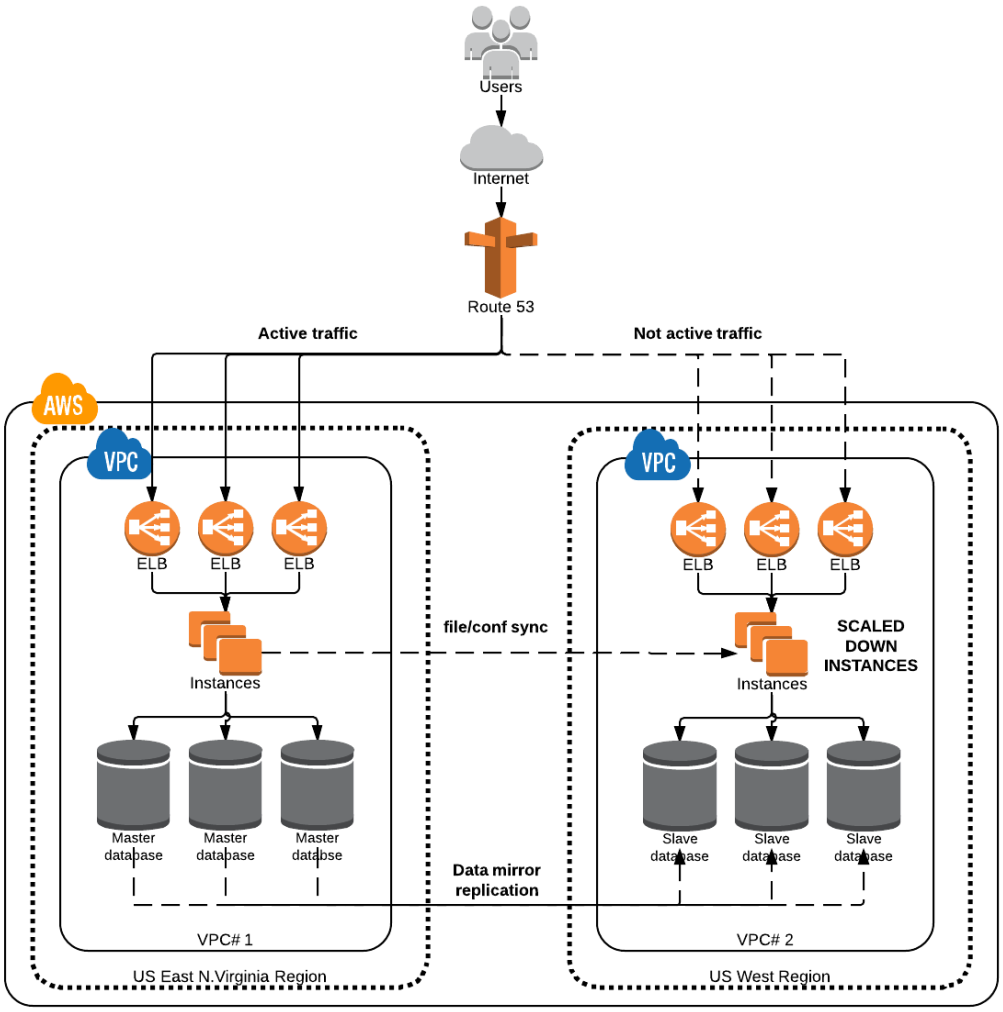
The warm standby solution is an extension of the pilot light scenario. Recovery time is decreased even further by having some additional services running. An analysis of the business's system is required to identify mission-critical systems. These systems will always be running in the DR environment.

The EC2 and RDS instances will not be running at full production size. The instance in the DR environment will be running at the smallest allowable size for the resource. Although the solution is not scaled, it is functional. It is used for development, testing or QA work in addition to being a DR environment.

Once a disaster occurs, the environment is scaled up in size so it can handle a production load. Rescaling is accomplished through either resizing the undersized instances or adding additional instance to the load balancer. Horizontal scaling is preferable to vertical scaling, as mentioned above.

Preparation Phase:

The following diagram shows an example of the Warm Standby DR solution. Both the production environment and the DR environment are running in AWS. The production environment is located in the US East region and the DR environment is located in US West.

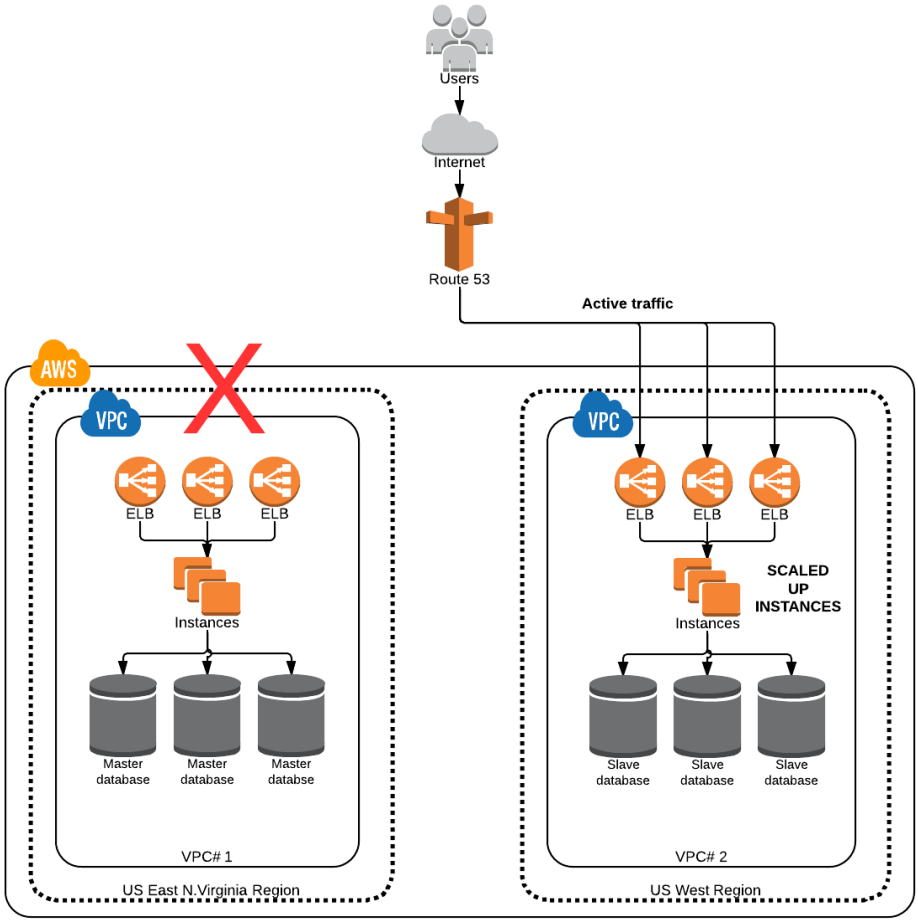


Key points for preparation:

- Configure EC2/RDS instances to receive replicated or mirrored data.
- Create and keep up to date Amazon Machine Images (AMIs) for the environment.
- Run your application using a minimal footprint of EC2 instances or AWS infrastructure.
- Patch and update software and configuration files in line with your live environment.

Recovery Phase:

When a disaster takes places, the standby environment will be scaled up for production load and DNS records are changed to route all traffic to the AWS DR Environment.



Key points for recovery:

- Implement vertical scaling by increasing the size of the EC2/RDS instances.
- Implement horizontal scaling by provisioning additional EC2 instances and adding them to the Load Balancer.
- Alter the DNS records so that all traffic is routed to the DR environment.
- Consider using Auto scaling to right-size the fleet or accommodate the increased load.

Multi-Site Solution deployed on AWS and on-Site

A multi-site DR solution is when the business's data center as well as the AWS DR site are splitting traffic and both are in use. The recovery point objective determines the method used to replicate the data. There are several methods available to replicate data.

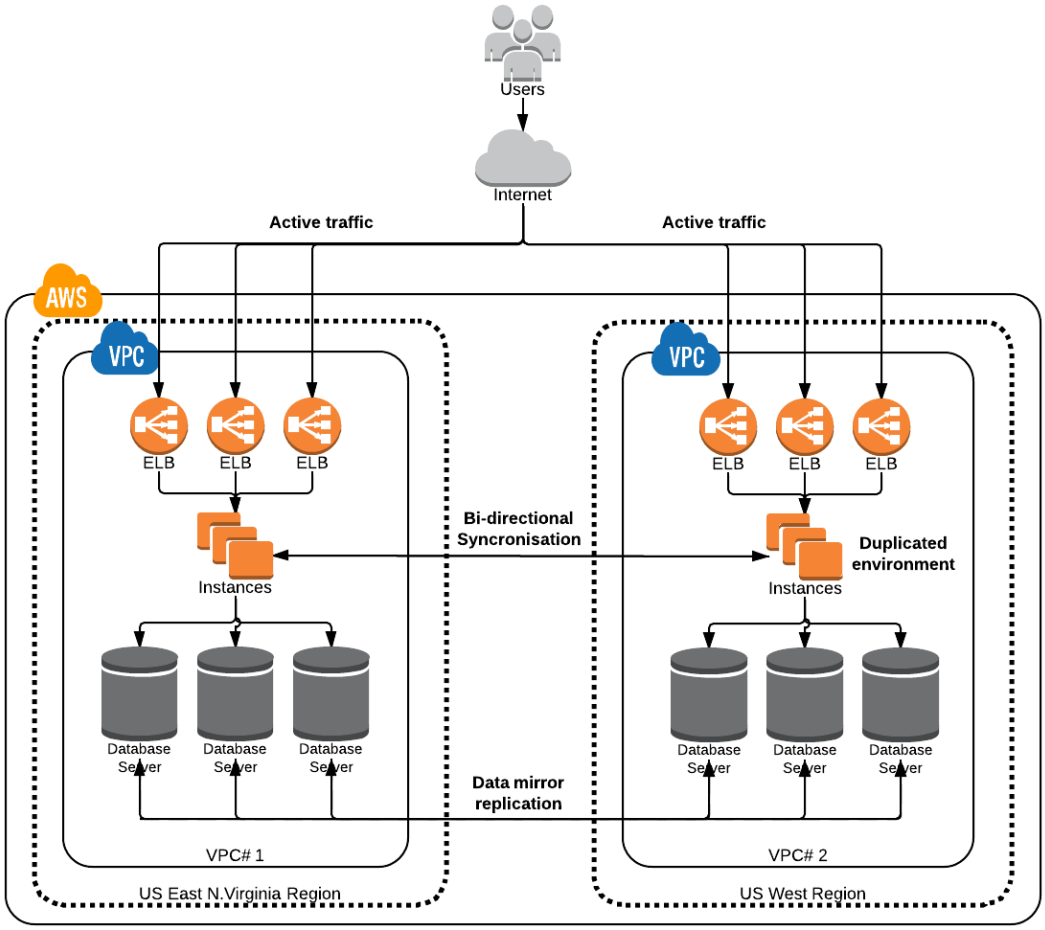
A weighted DNS service, such as Amazon Route 53, is used to route production traffic to the different sites. Part of production traffic will go to the DR site and the remainder will go to the primary production site.

Once a disaster occurs, adjustments to DNS weighting can be made and all traffic can be sent to the DR environment. If necessary, the environment can be scaled, either horizontally, vertically or both, to handle the increase in production traffic. EC2 Auto Scaling can be used to automate horizontal scaling. Additional scripts or logic may be required to run constant health checks on the primary production environment to check if switch over to the DR site is needed.

Since at least a portion of production traffic is being routed to the DR site, the cost of this solution tends to be higher than the other solution. When in the recovery phase, the cost is comparable to the other scenarios. During the preparation phase, cost can be reduced through the use of Reserved Instances for the instances that are required to be on.

Preparation Phase:

In the figure below, DNS is used to route a portion of the traffic to the DR site. The application running in the DR environment may access data sources in the on-site production system. Data is still replicated or mirrored to the DR environment.

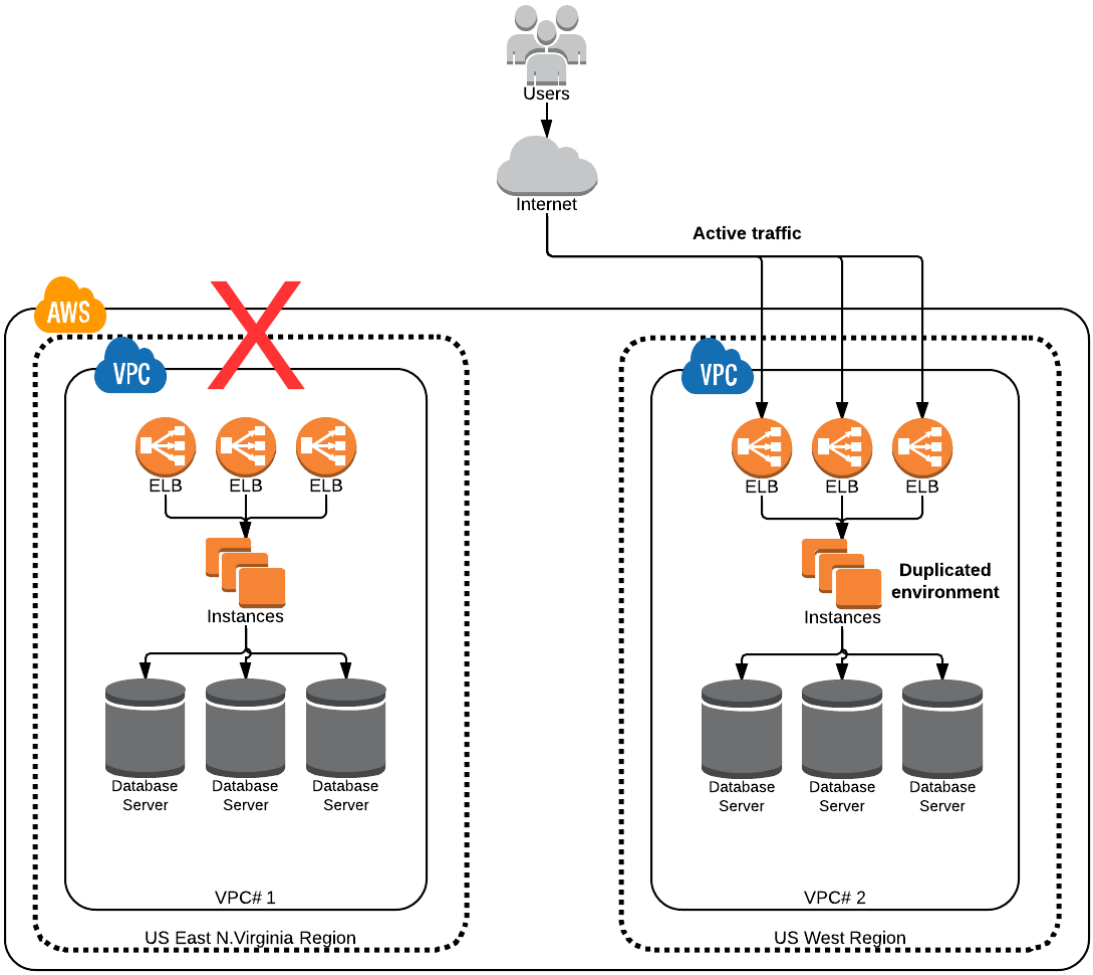


Key points for preparation:

- Duplicate the production environment in the DR environment.
- Configure DNS weighting or similar technology to distribute incoming traffic to both sites.

Recovery Phase:

The figure below shows what happens when a disaster occurs on-site. All traffic is routed to the DR environment by adjusting DNS weighting.



Key points for recovery:

- Change the DNS weighting, so that all traffic is sent to the DR environment.
- Have application logic for failover to use the the DR environment database servers.
- Consider using Auto scaling to automatically spin up the proper amount of instances in the DR environment.

Replication of Data

There are two categories of approaches when replicating data: **synchronous** and **asynchronous**.

Synchronous Replication

Data is atomically updated in multiple locations. This means a write transaction is not considered complete until both local and remote locations acknowledge receiving the data. Synchronous replication is extremely reliant on network performance and availability.

Asynchronous Replication

Data is not atomically updated in multiple locations. It is transferred as network performance and availability allows. Applications will also continue to write data that may not be fully replicated yet.

Many database systems support asynchronous data replication. The database replica can be located remotely, and the replica does not have to be completely in sync with the primary database server. In many scenarios this is acceptable, such as a backup source or reporting/read-only use cases.

In AWS, Availability Zones within a Region are well connected, but physically separated. For example, when deployed in "Multi-AZ" mode, the Amazon Relational Database Service (RDS) uses synchronous replication to duplicate data in a second Availability Zone. This ensures that data is not lost if the primary Availability Zone becomes unavailable.

AWS Regions are completely independent of each other, but there are no differences in how they are accessed and used. This allows businesses to place disaster recovery environments that can cover continental or even global distances, without the cost or issues that might regularly happen. Even if a large-scale disaster occurs, the use of AWS Regions allows for creating a DR environment or backup of data to localities unaffected by the event.

Improving Your DR Plan

In order to have a concrete DR plan, some important steps must be followed. This section describes some of the main steps required.

Testing

Once the DR environment is provisioned and in place, it needs to be tested. "Game Day" is when you exercise a failover to the DR environment. It is necessary to ensure that sufficient documentation is in place to make the process as simple as possible should a real disaster happen. Spinning up a duplicate environment for testing the game day scenario is quick and cost-effective on AWS. It is important to note that the production environment should not need to be touched. AWS CloudFormation can be used to deploy complete environments in AWS. Cloudformation uses a template to describe the AWS resources, and any associated dependencies or runtime parameters, required to create a full environment.

A variety of tests is key to ensuring that the DR solution is covered against different types of disasters. The examples below show some typical "Game Day" scenarios:

- Power loss to a site or a set of machines
- Loss of ISP connectivity to a single site
- Virus impacting core business services affecting multi-sites
- User error that caused the loss of data requiring a point-in-time recovery

Monitoring and alerting

Scheduled checks and sufficient monitoring is required to alert the business users that the DR environment has suffered an event, such as application issues, server failure, and connectivity issues. Amazon CloudWatch provides access to metrics for AWS resources. Alarms can be provisioned to watch for defined thresholds on a variety of metrics. Notifications can also be created and sent out if a threshold is met or an unexpected event occurs.

There are a variety of monitoring solutions that can be used to monitor AWS. Businesses can also leverage any monitoring they have in place to monitor their instances and servers.

Backups

Once the failover has occurred and the environment is dealing with production data, it is essential to continue to make regular backups. Testing backup and restore regularly is critical as a fallback solution.

Since the DR infrastructure does not need to be 'always on', inexpensive and frequent DR tests are easy to configure.

User Access

Identity and AWS Access Management (IAM) can be used to secure access to resources in the DR environment. Role/user based security policies can be created to divide user responsibilities in the DR environment.

Automation

Automation of the deployment of applications onto AWS-based instances and on-premises servers is possible by using configuration management or orchestration software. The use of this software will allow the handling of application and configuration change management across both environments with ease. AWS CloudFormation works in conjunction with several tools to provision the infrastructure services in an automated manner. User data can be passed into the instance on first boot and then handed to a configuration management tool to determine the instance type or role to ensure that the correct software and configuration is deployed. The overall goal is to have the instances end up in the final required state as automatically as possible.

Auto Scaling can be used to ensure that the pool of instances is appropriately sized to meet the demand based on the metrics specified in CloudWatch. In a DR situation, as traffic increases, the solution can scale up dynamically to meet this increased demand. After the event is over and usage potentially decreases, the solution can scale back down to a minimum level of servers.

Software Licensing and DR

Making sure that licensing is correct in AWS is just as important as in any other environment. Amazon provides a variety of models to make licensing easier. For example, “Bring Your Own License” is possible for several software components or operating systems. Alternately, there is a range of software for which the cost of the license is included in the hourly charge. This is known as “License included”.

“Bring your Own License” enables a business to leverage existing software investments during a disaster. “License included” minimizes up-front license costs for a DR site that does not get used on a day-to-day basis, i.e. during a DR test.

Conclusion

Many options and variations for DR exist, and this paper highlights some of the common patterns, ranging from simple backup and restore to fault tolerant multi-site solutions. AWS allows fine grained control and many building blocks to build the appropriate DR solution, based on DR objectives (RTO and RPO) and budget. The AWS services are available on-demand and cost is only incurred on use. This is a key advantage for DR where significant infrastructure is needed quickly but only in the event of a disaster.

This paper has shown how AWS provides flexible, cost-effective infrastructure solutions, enabling a more effective DR plan.

HOW TO CONTACT A CLOUD SPECIALIST

If you would like more information about the Amazon Web Services Public Cloud or Backup and Disaster Recovery best practices - please send your request to:

business-development@aximCloud.com