§ Disaster Recovery

Disaster Recovery Overview

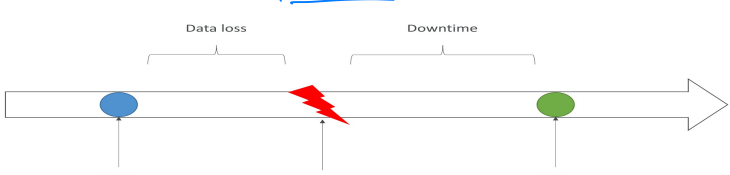
what is disaster ?

- Any event that has a negative impact on a company's business continuity or finances is a disaster
- Disaster recovery (DR) is about preparing for and recovering from a disaster
- What kind of disaster recovery?
  - On-premise => On-premise: traditional DR, and very expensive
  - On-premise => AWS Cloud: hybrid recovery (backup on cloud ?
  - AWS Cloud Region A => AWS Cloud Region B (just cloud)
- Need to define two terms:
  - RPO: Recovery Point Objective
  - RTO: Recovery Time Objective

Imp!

RPO → how often can you run a backup ! every 5 hrs, 1 hr so jab last backup से आई से recover करेंगे तो there will be a data loss ! so how much of this type of data loss is RPO !!

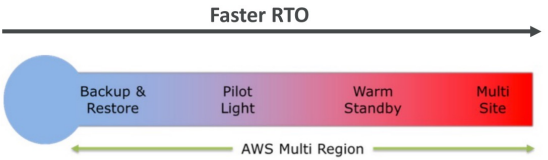
RTO → When you recover from the disaster, कितने disasters happen and what is downtime for recovery !



∴ We need to optimize the RTO & RPO as per our use case smaller the time higher the cost.

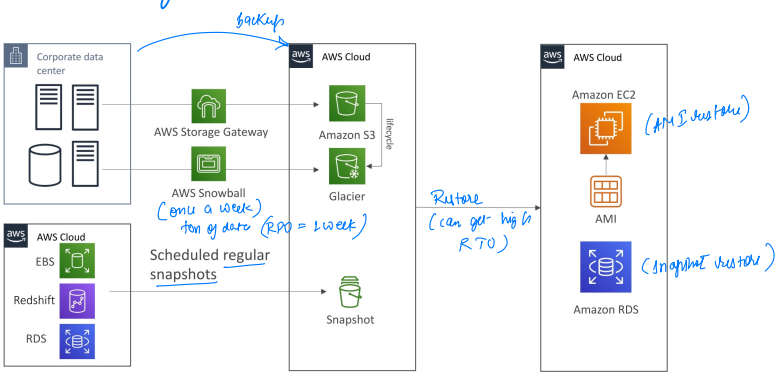
Disaster Recovery strategies

1. Backup & restore
2. Pilot light
3. Warm standby
4. hot site / Multi site approach



(1) Backup & restore:

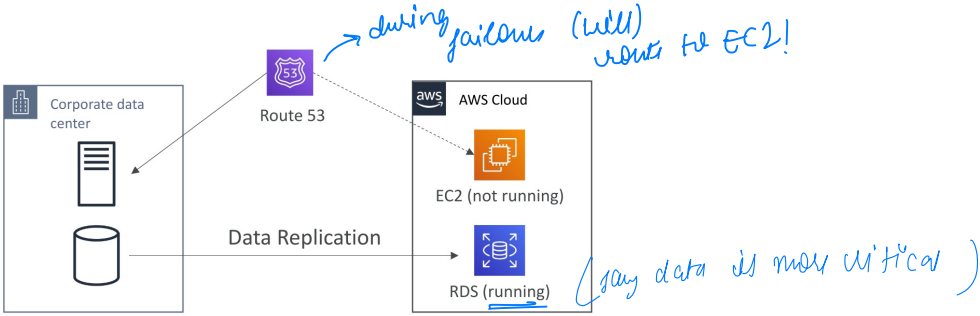
→ high RPO (data time)



→ cheapest
→ only cost of storing the backups!
→ very cost, high RPO, high RTO

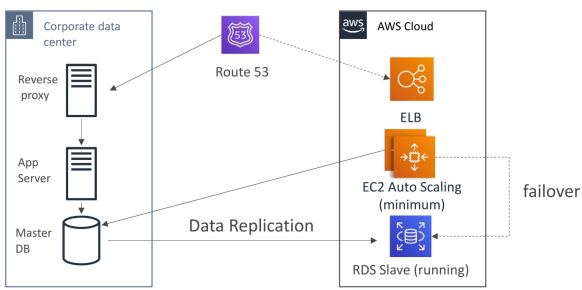
(2) Pilot light

→ A small version of app is always running in cloud.
(i.e. critic core (pilot light))
→ more expensive than backup & restore.
→ faster bc. system is already running.
→ lower RPO, lower RTO.



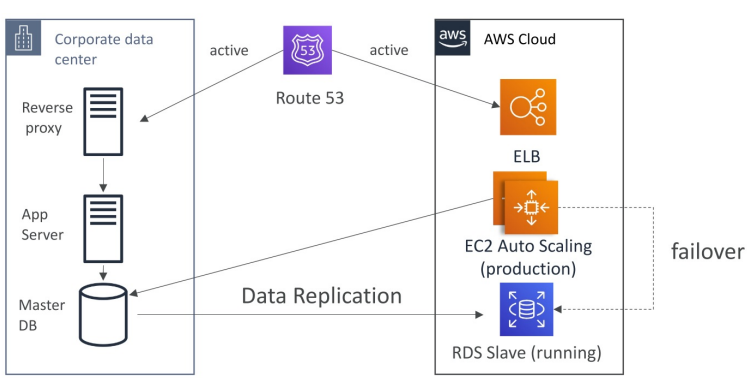
(3) Warm standby

→ A small sized application already running in cloud
→ upon disaster, the production load is scaled

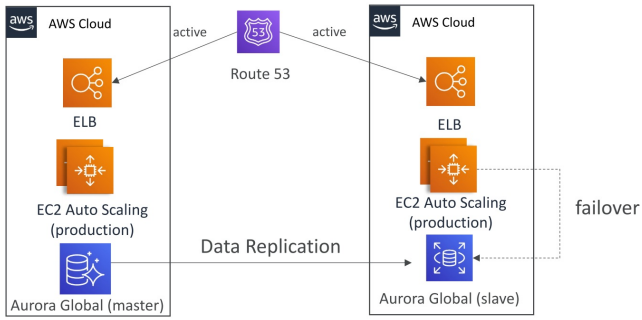


(4) Hot site

→ Very low RTO, very expensive
→ full production scale running on AWS & on premises



(5) All AWS Multi site



- Backup
  - EBS Snapshots, RDS automated backups / Snapshots, etc…
  - Regular pushes to S3 / S3 IA / Glacier, Lifecycle Policy, Cross Region Replication
  - From On-Premise: Snowball or Storage Gateway
- High Availability
  - Use Route53 to migrate DNS over from Region to Region
  - RDS Multi-AZ, ElastiCache Multi-AZ, EFS, S3
  - Site to Site VPN as a recovery from Direct Connect
- Replication
  - RDS Replication (Cross Region), AWS Aurora + Global Databases
  - Database replication from on-premises to RDS
  - Storage Gateway
- Automation
  - CloudFormation / Elastic Beanstalk to re-create a whole new environment
  - Recover / Reboot EC2 instances with CloudWatch if alarms fail
  - AWS Lambda functions for customized automations
- Chaos Engineering works
  - Netflix has a "simian-army" randomly terminating EC2 ; facebook uses "chaos monkey"

Database Migration Service (DMS)

→ migration from on-premise to AWS,

→ Src DB remains available, self healing

→ supports both homogeneous & heterogeneous migration

→ Continuous Data Replication using CDC (change Data Capture)

→ need to provision & EC2 instance for running DMS.

DMS Sources and Targets

SOURCES:

- On-Premises and EC2 instances databases: *Oracle, MS SQL Server, MySQL, MariaDB, PostgreSQL, MongoDB, SAP, DB2*
- Azure: *Azure SQL Database*
- Amazon RDS: all including Aurora
- Amazon S3
- DocumentDB

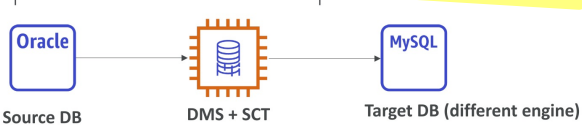
TARGETS:

- On-Premises and EC2 instances databases: Oracle, MS SQL Server, MySQL, MariaDB, PostgreSQL, SAP
- Amazon RDS
- Redshift, DynamoDB, S3
- OpenSearch Service
- Kinesis Data Streams
- Apache Kafka
- DocumentDB & Amazon Neptune
- Redis & Babelfish

AWS Schema Conversion Tool (SCT)

→ to convert from one DB engine to other!

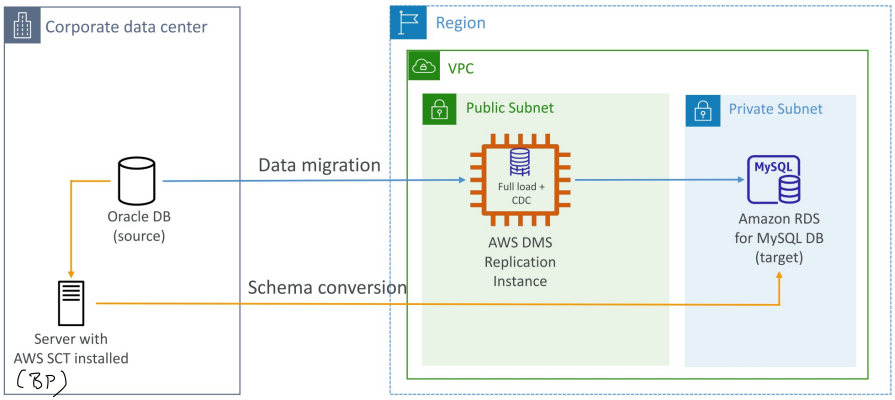
- Example OLTP: (SQL Server or Oracle) to MySQL, PostgreSQL, Aurora
- Example OLAP: (Teradata or Oracle) to Amazon Redshift
- Prefer compute-intensive instances to optimize data conversions



- You do not need to use SCT if you are migrating the same DB engine
  - Ex: On-Premise PostgreSQL => RDS PostgreSQL
  - The DB engine is still PostgreSQL (RDS is the platform)

DMS Continuous Replication

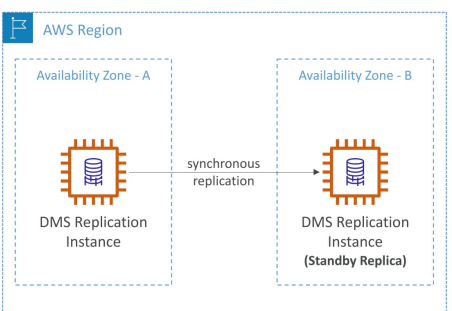
→ uses CDC (change Data Capture)



AWS DMS – Multi-AZ Deployment

- When Multi-AZ Enabled, DMS provisions and maintains a synchronously stand replica in a different AZ

- Advantages:
  - Provides Data Redundancy
  - Eliminates I/O freezes
  - Minimizes latency spikes



DMS → discover & assess [planning by looking @ on premises]
→ convert [schema conversion]
→ ① migrate [actual migration]
→ homogen
→ heterogen
→ instance-based
→ configure the instance
→ DMS engine version
② setup src & tg endpoints?
③ DB migration task

RDS & Aurora MySQL Migrations

same for RDS PostgreSQL to Aurora PostgreSQL.

(1) RDS to Aurora MySQL :

(a) DB snapshot & restore { downtime }

(b) Create a Aurora read replica on top of RDS DB & once the replication lag is 0 meaning all that data in RDS is in Aurora then promote Aurora as its own cluster. { can take time + network cost }

(2) On-premises MySQL to Aurora MySQL

(a) User Percona XtraBackup to create a backup file on S3 & then import from S3 bucket to create a Aurora DB

(b) create an Aurora MySQL & use mysql dump to create a dump file of the database & restore it later (slower)

(3) use DMS if both DB are running to do a continuous replication without disturbing them.

; for postgreSQL,

(1) create a backup and put it to S3

(2) use aws_s3 extension to import it.

On premises setups with AWS

- Ability to download Amazon Linux 2 AMI as a VM (.iso format)
  - VMWare, KVM, VirtualBox (Oracle VM), Microsoft Hyper-V
- VM Import / Export
  - Migrate existing applications into EC2
  - Create a DR repository strategy for your on-premises VMs → and back them up on cloud
  - Can export back the VMs from EC2 to on-premises
- AWS Application Discovery Service
  - Gather information about your on-premises servers to plan a migration
  - Server utilization and dependency mappings
  - Track with AWS Migration Hub
- AWS Database Migration Service (DMS)
  - replicate On-premise => AWS , AWS => AWS, AWS => On-premise
  - Works with various database technologies (Oracle, MySQL, DynamoDB, etc..)
- AWS Server Migration Service (SMS)
  - Incremental replication of on-premises live servers to AWS

AWS Backup

→ fully managed & allows to centrally manage & automate backups across AWS services.

→ central view & no manual processes.

- EC2 / EBS
- S3
- RDS, Aurora, DynamoDB, Document DB, Neptune,
- EFS / FSx
- Storage Gateway (volume gateway)

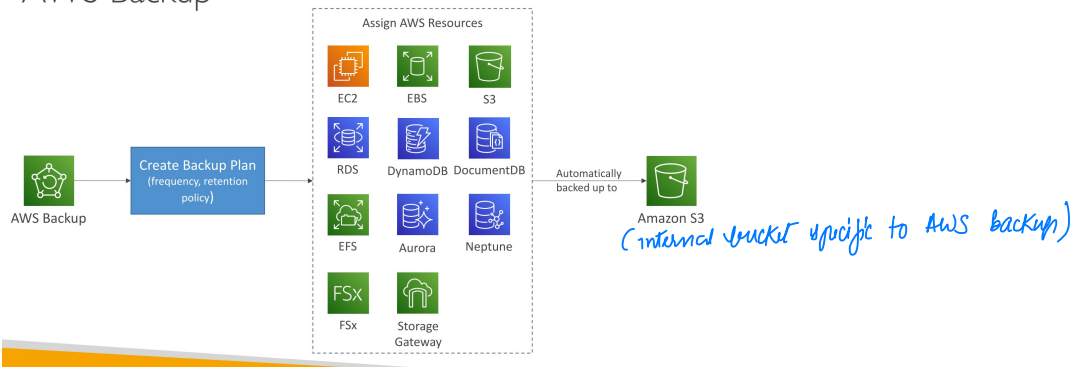
→ cross region backups & cross accounts.

→ PITR (Point in time recovery)

→ on-demand & scheduled

→ Backup policies (Backup plans)
- tag based
- frequency based (cron)
- Backup window
- transition to cold store.
- Retention period.

AWS Backup



(internal bucket specific to AWS backup)

Backup Vault Lock

→ to enforce WORM (Write Once Read Many) state

→ to protect backup against:
  (1) malicious delete
  (2) altered retention period

→ even the root / aws cannot delete the backups locked in a vault lock.

AWS Application Discovery service

→ Plan migration of on premises data centers

→ resource estimation, dependency management are imp for migration.

(1) Agentless discovery connector
(VM ware) VM; config, perfo. history such as CPU, mem & disk usage.

(2) Application discovery agent (Agent based discovery)
(Windows) sys. config., sys. perfo., running process, & network details,

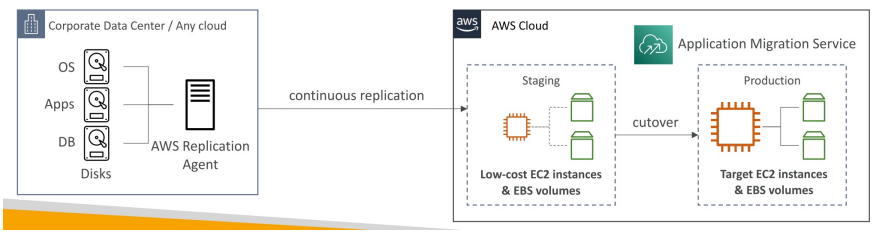
how are used to move & how is it inter-connected

→ AWS migration hub (centralized view)

AWS Application migration service (MGN)

→ lift-and-shift (rehosting) solution

→ converts physical or cloud based servers to run on AWS.



→ minimal downtime & lower cost.

Transfering large amount of data into AWS

" 200 TB of data over a 100 Mbps connection "

(1) over internet / S2S VPN

→ Imidiate setup

→ but will take longer time for data transfer.

$$\frac{200 \times 1000 \times 1000 \times 8}{100} \approx 185 \text{ days.}$$

(2) Direct connect provisioned with 1 Gbps

→ take longer time to setup. Costs a month

$200 \times 1000 \times 8 = 18.5$ days.

(3) Over snow ball

→ order (take 2 to 3 in parallel)

→ take about 1 week for the end to end transfer.

→ if database is being transferred then combine with DMS.

④ For ongoing replication / transfer:

S2S VPN, DX with DMS or datasync.

VMware Cloud on AWS

- Some customers use VMware Cloud to manage their on-premises Data Center
- They want to extend the Data Center capacity to AWS, but keep using the VMware Cloud software
- …Enter VMware Cloud on AWS
- Use cases
  - Migrate your VMware vSphere-based workloads to AWS
  - Run your production workloads across VMware vSphere-based private, public, and hybrid cloud environments
  - Have a disaster recover strategy

Extend entire VMware infrastructure to AWS.

Because we can keep using the same software suites