Disaster Recovery

Disaster Recovery is a process of how your application will recover from a disaster.

When designing a disaster recovery solution one needs to consider these two criterias:

1. Recovery Time Objective (RTO)
2. Recovery Point Objetive (RPO)

1. Recovery Time Objective (RTO):

It is the maximum time it will take for the service to become re-operational after a disaster has occurred.
So for an example, if the RTO for our service is 1 hour.
Then the disaster recovery should be designed in such a way that the service can become operational within 1 hour of a disaster.

2. Recovery Point Objective (RPO) :

It is the amount of Data Loss that is bearable once the service becomes operational after a disaster.

So, if a service has RPO of 2 Hours. And the disaster took place at 9AM in morning and the service became re-operational at 10PM.
Then, as per RPO the service should be able to recover all data that was present at or before 7AM.

Data recovered = Time of Disaster - RPO = 9 - 2 = 7 AM.

Disaster Recovery solutions:

There are 4 types of disaster recovery solutions:

1. Backup and Recover.
2. Pilot Light Approach.
3. Warm-StandBy Approach.
4. Multi-Site Approach.

1. Backup and Recover:

This solution implies that the data is backed up at specific intervals.
So that during the time of disaster these backups can be used to recover data.

The backups interval needs to be decided based on the RPO (Recovery Point Objective) of the service.
For faster backups to S3 from on-premise, one should use Direct connect or AWS Import/Export.

Once the Backup mechanisms are setup, you need to setup pre-configured AMI's. So that in the even of disaster new EC2 instances can be launched based on these AMI's which can use the EBS volumes

created from backup snapshots stored in S3.

Fig: AWS disaster recovery whitepaper pg 10

2. Pilot Light Approach:

In a Gas furnance, a small flame is always kept running and when the furnance needs to be ignited this small flame is used to ignite the whole furnance. This small flame is also called Pilot Light.
The Pilot light approach says that, keep the core set of resources always running and during disaster use this core set to create and run full set of resources.

Mostly a Database replica serves as a core for our service and is always kept running with minimal resources. This is our Pilot light.
The Application structure is created but kept as a StandBy, this can be in the format of a Cloudformation template and AMI's. This is our furnance.

At the time of disaster when the on-premise system fails. Our StandBy system is made active and DNS configuration is changed to route traffic to our newly created resources.
Fig: pg 13

3. Warm-StandBy Approach:

This approach is the second level of Pilot Light Approach, and attempts to bring the recovery time to almost **Zero**.

In this instead of only having the core resources active. All the resources required by an application are kept in active state but only having minimal sizes. So, they cannot take up production load but can be used for certain dev activities.

At the time of disaster, the size of the resources are scaled up and scaled out so that it can take up the production load.

4. Multi-Site Approach:

This is the next step to Warm-StandBy, the resources required by application are not only kept active but is also serving production load.

When a disaster strikes, the rest of the traffic that was served by the on-premise infrastructure is also routed to this infrastructure.