DOMAIN NAME: CLOUD APPLICATION DEVELOPMENT
PROJECT NAME: DISASTER RECOVERY WITH IBM CLOUD VIRTUAL SERVERS.

COMPONENTS USED IN DISASTER RECOVERY WITH IBM CLOUD VIRTUAL SERVERS :

- IBM cloud virtual servers
- High Availability (HA) Architecture
- Load Balancers
- Data Replication
- Backup and Restore
- Disaster Recovery Plan
- Monitoring and Alerting
- Network Connectivity
- Security Measures

Solution Components and Requirements Components:

This solution uses the following components

- Open an IBM Cloud account.
- Create two Power PowerVS location Services and a private subnet in each PowerVS location.
- Provision IBM i VSIs in each PowerVS location a. A “production” IBM i cloud instance with an Independent ASP (IASP) that has been IASP-enabled (i.e. All changes/modifications allowing the IASP to function in a working environment

Dshould be completed before Geographic Mirroring is set up for a DR solution.) b. A “DR” IBM i cloud instance with non-configured disks to be used for Geographic Mirroring. It is highly recommended that the number, type and capacity of disks match that of the production IASP.

- Order Direct Link Connect Classic to connect each PowerVS location to IBM Cloud.

- Order two Vyatta Gateways one in each datacenter to allow for PowerVS location-to-location communication 3.

- Request a Generic Routing Encapsulation (GRE) tunnel to be provisioned at each PowerVS location.

- Configure three GRE tunnels in the Vyatta Gateways. Two to connect Vyatta Gateway to the PowerVS location GRE tunnels created in Step 6 above and one across Vyatta Gateways to connect Vyatta-to-Vyatta. This will allow end-to-end PowerVS location-to-location communication for the VSIs in the PowerVS locations and to the IBM Cloud VSIs and other services such as Cloud Object Storage (COS) (if used).

- Configure a Reverse-proxy Centos VSI to allow access to Private Cloud Object Storage endpoint from PowerVS location Requirements Open an IBM Cloud account Login to https://cloud.ibm.com and follow the procedure to open an Internal to external account. For internal accounts, you can use your IBM intranet ID and password. For external accounts you will need to provide a billing source such as a credit card.

ARCHITECTURE OF DISASTER RECOVERY WITH IBM CLOUD VIRTUAL SERVERS:

Primary Data Center :

- This is the primary location where your applications and data reside, running on IBM Cloud Virtual Servers.

- Data is constantly updated and synchronized in this primary data center or region.

Secondary Data Center :

- This is the secondary location, geographically separated from the primary data center or region.

- IBM Cloud Virtual Servers or equivalent infrastructure is set up in this secondary location.

- Data is replicated from the primary to the secondary data center in near real-time.

Failover Mechanism:

- Develop an automated failover mechanism that can detect issues in the primary data center and initiate the failover process to the secondary data center.

- This may involve using orchestration tools and scripts that can start the necessary virtual servers in the secondary data center.

Testing and Drills:

Conduct regular disaster recovery tests and drills to ensure that the failover process works as expected.
These tests help identify and address any issues in the architecture and procedures.

DNS Management:

Use a robust DNS management solution to facilitate the switchover from the primary data center to the secondary data center when a disaster occurs.
This can include setting low TTLs (Time To Live) on DNS records for rapid propagation.

Security Measures:

Implement security measures, including encryption, access control, and identity and access management, in both data centers.
Ensure that data remains secure during disaster recovery processes.

Documentation:

Maintain detailed and up-to-date documentation of the entire disaster recovery architecture, including configurations, contact information, and recovery procedures.

Communication Plan:

Have a communication plan in place for notifying relevant personnel and stakeholders in case of a disaster.

Third-Party Tools and Services:

Depending on your specific needs, you may integrate third-party disaster recovery tools and services, such as backup solutions or advanced failover mechanisms.

Terraform Code for Provisioning Virtual Servers (main.tf):

```
provider "ibm" {
  generation = 2
  region    = "us-south"
}

resource "ibm_is_instance" "primary_server" {
  name    = "primary-server"
  profile = "b-2x4"
  image   = "r014-2c-16gb"
  zone    = "us-south-1"
  // Add network configuration
}

resource "ibm_is_instance" "secondary_server" {
  name    = "secondary-server"
  profile = "b-2x4"
  image   = "r014-2c-16gb"
  zone    = "us-south-2"
  // Add network configuration
}
```

Bash Script for Failover (failover.sh):

```
#!/bin/bash

# Check the status of the primary server
```

```
if ! server_status=$(ibmcloud is instance-get primary-server --output json); then
  echo "Error checking primary server status."
  exit 1
fi

if [[ "$server_status" == "running" ]]; then
  echo "Primary server is running. No failover needed."
  exit 0
fi

# If the primary server is not running, initiate the failover process
echo "Primary server is not running. Initiating failover..."

# Update DNS records or reroute traffic as needed
# Example: You would need to update DNS records to point to the secondary server's IP address.

# Start the secondary server
if ! ibmcloud is instance-start secondary-server; then
  echo "Error starting the secondary server."
  exit 1
fi

# Notify administrators or relevant personnel about the failover

echo "Failover complete. Services now running on the secondary server."
```