Rehydration Process of YugabyteDB Nodes Running on GCP

Purpose

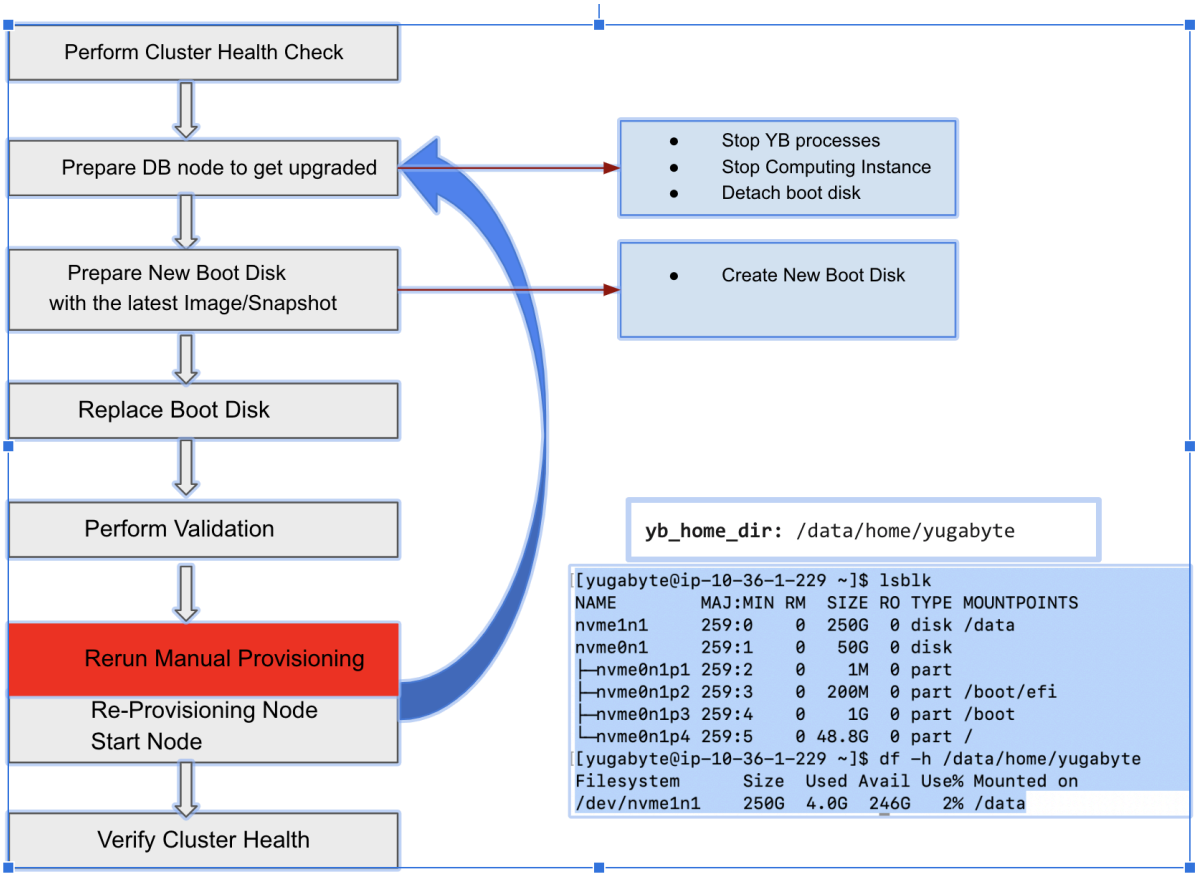
This runbook defines a fully automated, step-by-step workflow for rehydrating YugabyteDB universe nodes running on GCP by deploying the latest image, while ensuring the preservation of private IP addresses and associated data volumes. In accordance with YugabyteDB operational best practices, each node's rehydration is designed to complete within approximately 15 minutes, minimizing disruption and maintaining cluster availability. Key requirements include:

- Preserve current private IP
- Preserve /Data mount point and data disks
- Complete the entire process of each node under 15 mins

Current Deployment Architecture of the customer on AWS

- Multi-region deployment: us-east-1 (2 azs), us-east-2 (1 az), us-west-2 (2 azs)
- Instance type: r7i.4xlarge
- Node count: 50 nodes (13 on us-east-1a, 13 on us-east-1b, 8 on us-east-2a, 8 on us-west-2a, 8 on us-west-2b); on us-east-1a and us-east1b, 3x2 TB gp3 EBS volumes on each node using one RAID-0 array and 6TB YB filesystem will be mounted on this array. on us-east-2 and us-west, 3x3.2 TB gp3 EBS volumes on each node using one RAID-0 array and 9.6TB YB filesystem will be mounted on this array.
- Total storage will be 78TB for each zone on US-east-1; 76.8TB for each zone on US-east-2 and Us-west-2
- Universe is configured using A Record
    - A record for each EC2 instance
    - A record per region (App team uses Regional A record in smart driver)

YugabyteDB Node Boot Disk Replacement Workflow



Prerequisites:

- Access to GCP project
  - IAM permissions to manage Compute Engine disks and instances
  - Existing GCP instance with: Data disk (e.g., XFS-formatted, mounted at /data)
  - Replacement base image available (e.g., rhel-9-v20240415)
- Access to Yugayte Anywhere Instance and Universes provisioned with on-Prem provider
- /data mount point exists
  - Yugabyte user: /data/home/yugabyte
- Required tools installed:
  - Google Cloud SDK
  - Yugabyte Anywhere CLI
  - Python 3.8+

Execution Steps:

The current flow is designed to process one node at a time. However, it can be efficiently extended to handle multiple instances in parallel—such as those within the same availability zone or fault domain—thereby improving the overall rehydration performance across the universe.

1. Create a Configuration File (yba_config.yaml)

```
Java
project_id: "xxxx"
instance_type: "n2-standard-4"
zone: "us-central1-a"
image_family: "rhel-8"
image_project: "rhel-cloud"
disk_size: 150
extra_disk_size: 250
new_disk_name: "new-boot-disk"
new_image: "rhel-9-v20240415"
disk_type: "pd-balanced"
sh_user: "sh_user"  # or whatever user you use
ssh_wait_time: 30     # seconds to wait for SSH to be available
yba:
  yba_host: "https://xx.xxx.xx.xx/"
  customer_id: "customer_id"
  yba_api_token: "YBA_API_TOKEN"
  universe_name: "UNIVERSE_NAME"
  node_list:
    - "10.128.15.210": "NODE_NAME"
  yba_cli_path:
"/opt/yugabyte/software/active/yb-platform/yugaware/yba-cli/yba_cli-2024.2.2.2-b2-li
nux-amd64/yba"
```

2. Stop the Node via YBA CLI

```
Java
$YBA_CLI_PATH universe node stop \
  -H $YBA_HOST -a $YBA_API_TOKEN \
```

```
  --name $UNIVERSE_NAME --node-name $NODE_NAME
```

3. Stop the Computing Instance

```
Java
gcloud compute instances stop "$INSTANCE_NAME" --zone "$ZONE" --project
"$project_id"
```

4. Create the New Boot Disk

```
Java
gcloud compute disks create "$NEW_DISK_NAME" \
  --image "$NEW_IMAGE" \
  --type "$DISK_TYPE" \
  --size "$DISK_SIZE" \
  --zone "$ZONE" \
  --proejct "$PROJECT"
```

5. Detach the Old Boot Disk from the computing instance

```
Java
gcloud compute instances detach-disk "$INSTANCE_NAME" --disk="OLD_BOOT_DISK"
--zone "$ZONE" --proejct "$PROJECT"
```

6. Attach the New Boot Disk to the computing instance

```
Java
gcloud compute instances attach-disk "$INSTANCE_NAME" \
```

```
  --disk "$NEW_DISK_NAME" \
  --zone "$ZONE" \
  --boot \
  --device-name=boot-disk \
  --mode=rw \
  --proejct "$PROJECT"
```

7. **Restart the Computing Instance**

```
Java
gcloud compute instances start "$INSTANCE_NAME" --zone "$ZONE" --proejct
"$PROJECT_ID"
```

8. **Verify Boot & Attachments of the Computing Instance**

```
Java
gcloud compute instances describe "$INSTANCE_NAME" --zone "$ZONE" \
  --format='get(networkInterfaces[0].networkIP,disks.deviceName)' \
  --proejct "$PROJECT"
```

9. **Start the Computing Instance**

```
Java
gcloud compute instances stop "$INSTANCE_NAME" --zone "$ZONE" --project
"$PROJECT_ID"
```

10. **Provision Node - Manual provisioning Steps required to provision a node, here are some sample steps**
    - **Mount Data Disk**
        - Identify data disk UUID
        - Create /data directory
        - Mount data disk
        - Update /etc/fstable for persistence

- **Provision Node Agent**
    - Clean up existing yugabyte user if exists
    - Create new yugabyte user with correct home directory
    - Set proper permissions
    - Run node-agent-provision.sh

11. Reprovision & Start Node with YBA CLI

```
Java
$YBA_CLI_PATH universe node reprovision \
  -H $YBA_HOST -a $YBA_API_TOKEN \
  --name $UNIVERSE_NAME --node-name $NODE_NAME

$YBA_CLI_PATH universe node start \
  -H $YBA_HOST -a $YBA_API_TOKEN \
  --name $UNIVERSE_NAME --node-name $NODE_NAME
```

Expected Outcome:

- The computing instance is running with the new boot disk
- Private IP and external volumes are unchanged
- It joined the universe successfully
- Old boot disk (if detached) can now be deleted or archived

Validation:

- Check node status in YBA UI and verify node is healthy
- Verify data disk is mounted
- Ensure `/data` is populated: `ls -l /data`
- Confirm `/etc/fstab` entry is present
- Check `node-agent-provision.sh` logs for success
- Check newly created disk info:

```
Java
gcloud compute instances describe INSTANCE_NAME \
  --zone=ZONE \
  --format="value(disks[0].source)"
```

```
gcloud compute disks describe DISK_NAME --zone=ZONE
```

Troubleshooting Matrix

Issue	Possible Causes	Solution
Node fails to start in YBA	- Node agent issues- SSH access problems- Permission issues	- Check node agent logs- Verify SSH access- Check user permissions
Data disk not mounting	- Incorrect UUID- fstab issues- Permission problems	- Verify disk UUID- Check /etc/fstab- Verify mount permissions
SSH access issues	- Instance status- Firewall rules- SSH key configuration	- Check instance status- Verify firewall rules- Check SSH key configuration