#!/bin/bash

LINODE\_API\_TOKEN= #include your own API token

VOLUME\_ID= # Input your volume ID (eg: linode-cli volumes list)

DATE\_TIME="`date +%Y%m%d%H%M%S`";

#volume group to be choose

volume\_group\_label="backup" #input unique label for your respective backup

linode\_cli() {

curl -H "Authorization: Bearer $LINODE\_API\_TOKEN" -X POST -H "Content-Type: application/json" \

-d "{\"label\": \"$volume\_group\_label-$DATE\_TIME\"}" \

"https://api.linode.com/v4/volumes/$VOLUME\_ID/clone"

}

response\_2=$(linode\_cli)

echo $response\_2

response=$(curl -H "Authorization: Bearer $LINODE\_API\_TOKEN" "https://api.linode.com/v4/volumes")

# Array to store the latest items

latest\_items=()

# Array to store the oldest items

oldest\_item=()

# Iterate over the JSON array using a process substitution for the while loop

while IFS= read -r item; do

decoded=$(echo "$item" | base64 --decode)

id=$(echo "$decoded" | jq -r '.id')

label=$(echo "$decoded" | jq -r '.label')

# Add the item to the latest\_items array

latest\_items+=($id)

# Remove the oldest items if the array length exceeds 2

if [ "${#latest\_items[@]}" -gt 2 ]; then

oldest\_item=${latest\_items[0]}

oldest\_items+=("$oldest\_item")

unset 'latest\_items[0]'

latest\_items=("${latest\_items[@]}")

fi

done < <(echo "$response" | jq -r --arg volume\_label "$volume\_group\_label" '.data[] | select(.linode\_id == null) | select(.label | contains($volume\_label)) | @base64')

if [ ${#oldest\_items[@]} -eq 0 ]; then

echo "Don't have any oldest volume"

else

#Delete Linode Oldest Volume

for item in "${oldest\_items[@]}"; do

delete\_response=$(curl -H "Authorization: Bearer $LINODE\_API\_TOKEN" -X DELETE "https://api.linode.com/v4/volumes/$item")

# Check if the volume deletion was successful

if [ $? -eq 0 ]; then

echo "Volume with ID $item deleted successfully"

else

echo "Failed to delete volume with ID $item"

fi

done

fi