Neo4j - Backup/Restore Best Practice



Agenda

- Backup - Overview
- Backup - Frequency
- Backup - Storage Considerations
- Backup - Best Practice and Memory Recommendations
- Backup - Command
- Backup - New features in 4.2
- Restore
- Restore - Command
- Consistency Check
- Offline Backup - neo4j-admin dump and load

Backup - Overview

Documentation Link: http://neo4j.com/docs/operations-manual/current/backup/perform-backup/

- Online Backup only available in Neo4j Enterprise (server and embedded)
- Backups are the most important part of operations and must be taken on regular basis.
- Online backup (enabled by default: dbms.backup.enabled)
- Databases to backup
  - System database is critical and should be backed up regularly
  - Rest of user databases will also need to be backed up
- Full backup:
  - Specify an empty target directory, or one that does not already contain a store file.
  - By default, automatically runs consistency check against the backed-up store.
- Incremental backup:
  - If the target directory contains a store, it will automatically capture an incremental backup.
  - Relies on historical transaction logs, so if those are already purged, it will take a new full backup. If you have problems with logs not being available, you can increase the dbms.tx_log.rotation.retention_policy setting in neo4j.conf.

Databases to Backup

- Database Backup strategy should include backup process for every database
- Each database needs to be backed up individually.
- Default there are 2 databases - neo4j and system
  - Neo4j - this is default user database
  - System - database contains configurations i.e. operational states of database, security configurations etc.
- It is very important to backup all the databases including the “System” database

Backup Frequency

Backup frequency depends on data and load process.

- Data load is more real time
  - Daily Full Backups and Incrementals hourly.
- Data is static and updated via nightly batches
  - Weekly Full Backups and Daily Incremental

Backup - Storage Considerations

- Backup can be taken and and stored locally
  - Depending on the size of the data store, backups can consume equal amount of space hence recommended to move outside of Neo4j instance for archivals.
  - On AWS -
    - *Backups taken on Neo4j instance (EC2) can be taken locally on EC2 instance but then recommended to be stored by copying them to Amazon S3 directly or by using EBS snapshot.*
- Backups can be taken from a remote server if available.
  - Preferred approach as it minimizes resource impact on the neo4j instance

Backup - Best Practice and Memory Configuration

- Backups uses memory and I/O, and can impact performance on the instance when run.
  - Run Backup from a remote server if available
  - If remote server is not available and neo4j is setup as a cluster with Read Replica, it is recommended to run the backup process from Read Replica.
  - If Read Replicas are not available, schedule the backup from Core Follower when there is not much activity on the system.
- It is always recommended to configure the memory parameters as it helps with backup performance
  - Configure heap size for the backup using environment variable ***HEAP_SIZE (e.g. export HEAP_SIZE=2G)***
  - Configure pagecache for the backup by specifying ***--pagecache*** in the backup command.

Backup - Best Practice and Memory Configuration

- Based on Backup frequency make sure to configure transaction log rotation appropriately `***dbms.tx_log.rotation.retention_policy=7 days***`
- Neo4j backup process does not backup configuration files (neo4j.conf) or SSL/TLS certificates. Hence make sure to have a separate process to back the files.
- Do not perform SCP, rsync and snapshots of the graph data store while the database is online.
    - Perform an online backup first and then perform scp, rsync or snapshot of the disk image(containing the backed up files).

Perform a Backup

```
neo4j-admin backup --backup-dir=<path>
                   [--verbose=<true/false>]
                   [--from=<host:port>]
                   [--database=<database>]
                   [--fallback-to-full=<true/false>]
                   [--pagecache=<size>]
                   [--check-consistency=<true/false>]
                   [--check-graph=<true/false>]
                   [--check-indexes=<true/false>]
                   [--check-label-scan-store=<true/false>]
                   [--check-property-owners=<true/false>]
                   [--report-dir=<path>]
                   [--additional-config=<path>]
                   [--include-metadata=<all/users/roles>]
```

Backup - New features in 4.2

- --include-metadata
  - Include metadata in the backup. Metadata contains security settings related to the database. Can't be used for backing system database.
  - **roles** - commands to create the roles and privileges (for both database and graph) that affect the use of the database
  - **users** - commands to create the users that can use the database and their role assignments
  - **all** - include roles and users

- --database
  - **Name** can contain `*` or `?` for name globbing. If a backup of the specified database exists in the target directory, then an incremental backup will be attempted.

Backup Demo

./bin/neo4j-admin backup --backup-dir=./backup/ --database=*

Restore

- Database can be restored from a backup copy using the ***neo4j-admin restore*** command as given :
  - https://neo4j.com/docs/operations-manual/current/backup/restoring/
- Database can be restored on a single instance of the database.
- Database can be restored on a cluster
  - Database must be restored on each of the core instances of the cluster

Perform a Restore

- `neo4j-admin restore --from=<path,path>`
  `[--verbose=<true/false>]`
  `[--database=<database>]`
  `[--force]`
  `[--move]`
  `[--to-data-directory=<path>]`
  `[--to-data-tx-directory=<path>]`

Restore Demo

./bin/neo4j-admin backup --backup-dir=./backup/ --database=*

Consistency Check

- The Consistency Check is run automatically during the backup
- For very large stores, or stores with very large indexes, this can be quite slow. Hence recommended to skip consistency check during backup and then schedule it separately using the ***neo4j-admin check-consistency*** command.
- This must be done on an OFFLINE copy of the data store. Consistency check can be run on the backed up copy of the data store files.
- Make sure to perform consistency check on regular (Weekly) basis as part of the operation process.

Offline Backup - neo4j-admin dump and load

- Offline backup can be taken using the `neo4j-admin dump` command
- Database must be offline for the dump command
  - Neo4j service can be running but the database has to be stopped
- Syntax: `neo4j-admin dump --database=<database> --to=<destination-path>`
- Dump command creates a “.dump” compressed file with the data store

Offline - neo4j-admin dump and load

- Offline backup (dump files) can be restored using `neo4j-admin load` command
- Load command decompresses and restores all the data store files.
- Syntax: `neo4j-admin load --from=<archive-path> --database=<database> [--force]`
- Loading a new database
  - Perform neo4j-admin load command to restore the database
  - Then login to system database
  - Create the new database that was loaded.



neo4j