

Backup and Recovery

Contents:

- *Backup Platform Files*
 - *Configuration*
 - *License*
 - *Backup Databases*
 - *Backup commands*
 - *Scheduling*
 - *Recovery*
-

This section provides overview information on the key data and metadata that should be managed by your enterprise backup and recovery policies.

All backups should be performed in accordance with your enterprise's backup and recovery policies.

Backup Platform Files

The following directories on the Trifacta node should be backed up on a regular basis:

Configuration

The following directory hosts key configuration files, including

`trifacta-conf.json`

:

```
/opt/trifacta/conf
```

License

You should backup your license key:

```
/opt/trifacta/license
```

See *License Key*.

Backup Databases

The Trifacta platform utilizes the following PostgreSQL databases as part of normal operations. These databases should be backed up on a regular basis:

Database Name	Databaseld	Description
Trifacta DB	trifacta	Stores users and metadata for flows, including datasets, and recipes.

Jobs DB	trifacta-activiti	Stores and maintains job execution status and details.
Scheduling DB	trifactaschedulingservice	Stores metadata for scheduled jobs.
Time-based Trigger DB	trifactatimebasedtriggerservice	Additional database required for scheduled jobs.

For more information on setting up these databases, see *Set up the Databases*.

Backup commands

The following commands can be used to back up the PostgreSQL databases to a compressed dump file. For more information on command options, see <http://www.postgresql.org/docs/9.3/static/backup.html>.

```
su - postgres
```

Trifacta DB:

```
/usr/pgsql-9.3/bin/pg_dump -Fc trifacta > trif_triDB_bkp_<date>.dump
```

Jobs DB:

```
/usr/pgsql-9.3/bin/pg_dump -Fc trifacta-activiti > trif_actDB_bkp_<date>.dump
```

Scheduling DB:

```
/usr/pgsql-9.3/bin/pg_dump -Fc trifactaschedulingservice > trif_schDB_bkup_<date>.dump
```

Time-Based Trigger DB:

```
/usr/pgsql-9.3/bin/pg_dump -Fc trifactatimebasedtriggerservice > trif_tbtsDB_bkup_<date>.dump
```

Scheduling

You can schedule nightly execution of these backups using a third-party scheduler such as cron.

Recovery

To recover the Trifacta platform based on backups:

NOTE: When the databases are restored, internal identifiers such as job IDs, are reset in an order that may not correspond to the expected order. Consequently, references to specific identifiers may be corrupted. After restoring the databases, you should clear the job logs.

NOTE: If any of the hosts, pathnames, or credentials have changed since the backups were performed, these updates must be applied through

```
trifacta-conf.json
```

or through the Admin Settings page after the restoration is complete.

Steps:

1. Perform a clean install of the Trifacta software provided in your distribution. See *Install*.
2. Apply any patches or maintenance updates that may have been provided to you. See *Maintenance Release Updater*.
3. Restore the `/opt/trifacta/conf` directory from backup.
4. Clear each current database and restore the backup of the version from the preceding release. In some cases, the database may not exist in the previous version.

a. Login:

```
su - postgres
```

b. Trifacta database:

```
psql -c "DROP DATABASE trifacta;"
psql -c "CREATE DATABASE trifacta WITH OWNER trifacta;"
psql --dbname=trifacta < trifacta_backup_<date>.sql
```

c. (Release 3.2 and later) Jobs database:

```
psql -c "DROP DATABASE trifacta-activiti;"
psql -c "CREATE DATABASE trifacta-activiti WITH OWNER trifacta;"
psql --dbname=trifacta-activiti < trif_actDB_backup_<date>.sql
```

d. (Release 4.1 and later) Scheduling database:

```
psql -c "DROP DATABASE trifactaschedulingservice;"
psql -c "CREATE DATABASE trifactaschedulingservice WITH OWNER trifacta;"
psql --dbname=trifactaschedulingservice < trif_schedDB_backup_<date>.sql
```

e. (Release 4.1 and later) Time-based Trigger Service database:

```
psql -c "DROP DATABASE trifactatimebasedtriggerservice;"
psql -c "CREATE DATABASE trifactatimebasedtriggerservice WITH OWNER trifacta;"
psql --dbname=trifactatimebasedtriggerservice < tri_tbtSDB_backup_<date>.sql
```

5. Restart the platform.
6. Login and verify operations. See *Verify Operations*.