



Backup and Recovery QuickStart Guide

Release 2.5.2

QuickStart Guide

Jan 13, 2020

Contents

1	Installation	2
2	Configuring BART	4
3	Configuring a Database Server	8
3.1	Authorizing SSH/SCP Access	8
3.2	Setting up a Replication Database User	9
3.3	Enabling WAL Archiving	9
3.4	Verifying Configuration Setting	9
4	Taking a Backup	11
5	Conclusion	14
	Index	15

This document provides shortcuts that allow you to install and configure BART and take a full and incremental backup of a database server quickly. Please note that your system may have requirements that are not addressed in this document. For detailed information about BART installation and configuration, see the *BART Installation and Upgrade Guide* available at:

<https://www.enterprisedb.com/edb-docs/>

BART is supported on the following platforms (64 bit only):

- CentOS 6.x or 7.x
- RHEL 6.x or 7.x
- PPC-LE 8 running RHEL or CentOS 7.x
- Ubuntu 18.04 (Bionic)
- Debian 9.x (Stretch)

BART works with the following database versions:

- Advanced Server versions 9.5, 9.6, 10, 11, and 12.
- PostgreSQL versions 9.5, 9.6, 10, 11, and 12.

Before installing BART, ensure that your repository configuration allows access to the EDB repository. To request credentials to the EDB repository, visit:

<https://www.enterprisedb.com/repository-access-request>

1. Then, use `yum` to create the repository configuration file:

```
yum install -y https://yum.enterprisedb.com/edbrepos/edb-repo-latest.noarch.rpm
```

2. Modify the repository configuration file (named `edb.repo`, located in `/etc/yum/repos.d`). Ensure that the value of the `enabled` parameter is `1`, and replace the `username` and `password` placeholders in the `baseurl` specification with the name and password of a registered EnterpriseDB user.
3. Before installing other software, use `yum` to install the EPEL package:

```
yum install epel-release
```

4. Then, use `yum` to install an Advanced Server or PostgreSQL database on any server on which an incremental backup will be restored. The host of the BART server is not required to have an installation of Postgres, but must include a copy of the following:
 - Postgres `libpq` library
 - Postgres `pg_basebackup` utility program
 - Boost Libraries version 1.48 and 1.53 (for RHEL/CentOS 6 and RHEL/CentOS 7 only).
5. Enable and activate Secure Shell (SSH) and Secure Copy (SCP) client programs on the BART host as well as on the remote database server hosts. The BART host and target database server host must accept a passwordless SSH/SCP login connection.

After meeting the prerequisites, you can install the BART RPM package directly from the EnterpriseDB yum repository with the following command:

```
yum install edb-bart
```

Repeat the installation process described in this section to install BART 2.4 on all remote hosts where incremental backups are to be restored.

BART is installed in the `/usr/edb/bart` directory location referred to as `<BART_HOME>`.

Configuring BART

To configure the BART host and each database server that is to be managed by BART, you must:

1. Establish the BART user account and ensure it runs the `bart` and the `bart-scanner` program.
2. Use the `bart.cfg.sample` file to create the `bart.cfg` file. The BART configuration file is located in `<BART_HOME>/etc/bart.cfg`:

```
cp bart.cfg.sample bart.cfg
```

3. Set the environment variable for the BART user account. If the `libpq` library does not reside in the default installation location, you must add the `libpq` library to the `LD_LIBRARY_PATH` environment variable and place the following settings in the BART user account's profile so they take effect upon login:

```
# .bash\_profile
# Get the aliases and functions
if [ -f ~/.bashrc ]; then
. ~/.bashrc
fi

# User specific environment and startup programs

export LD_LIBRARY_PATH=/usr/edb/as11/lib:$LD_LIBRARY_PATH
```

4. Set the following parameters in the `[BART]` section of the BART configuration file:

bart_host (required)

Use this field to specify the IP address of the host where BART is installed, in the form of `<bart_user>@<bart_host_address>`.

backup_path (required)

Use this field to specify the path to the file system parent directory where all BART backups and archived WAL files will be stored. Ensure the BART user account owns the location specified in the `backup_path` parameter.

pg_basebackup_path (required)

Use this field to specify the path to the `pg_basebackup` utility.

xlog_method (optional)

Use this field to specify how the transaction log should be collected during the execution of `pg_basebackup`. The default value is `fetch`; it specifies that the transaction log files will be collected after backup is completed. Set to `stream` to stream the transaction log in parallel with the full backup creation.

retention_policy (optional)

Use this field to specify the retention policy for the backup. This determines when an active backup should be marked as `obsolete`, and hence, be a candidate for deletion. You can specify the retention policy in terms of number of backup or in terms of duration (days, weeks, or months). For example, `max_number BACKUPS` (default setting), `max_number DAYS`, `max_number WEEKS`, or `max_number MONTHS` where `max_number` is a positive integer.

wal_compression (optional)

Use this field to specify if you want to compress the archived Xlog/WAL files in `gzip` format. Set to `enabled` to compress the archived WAL files in `gzip` format when the `MANAGE` subcommand is invoked.

To enable WAL compression, the `gzip` compression program must be present in the BART user account's `PATH`. The WAL compression setting must not be enabled for those database servers where you need to take incremental backups.

copy_wals_during_restore (optional)

Use this field to specify how the archived WAL files are collected when invoking the `RESTORE` operation. Set to `enabled` to copy the archived WAL files from the BART backup catalog to the `<restore_path>/<archived_wals>` directory prior to the database server archive recovery. Set to `disabled` to retrieve the archived WAL files directly from the BART backup catalog during the database server archive recovery. Enabling this option helps you save time during the restore operation.

logfile (optional)

Use this field to specify the path to the BART log file.

scanner_logfile (optional)

Use this field to specify the path to the Xlog/WAL scanner log file. BART does not create a WAL scanner log file if you do not specify the path.

bart_socket_directory (optional) -

Use this field to specify the socket directory path where all BART sockets will be stored. The default directory is `/tmp`.

While specifying the `bart_socket_directory` path, you must ensure that the directory exists and the BART user has the required directory access permissions.

thread_count (optional)

Use this field to specify the number of worker threads to copy blocks or data files from the database server to the BART backup catalog.

Specify a thread count of 1 if you want to take the backup using the `pg_basebackup` utility.

batch_size (optional)

Use this field to specify the number of blocks of memory used for copying modified blocks from the database server to the BART backup catalog when the `BACKUP` subcommand is invoked for incremental backups. The maximum permitted value is 131072 ($131072 * 8192 = 1$ GB). The minimum permitted value is 1 ($1 * 8192 = 8192$ bytes).

scan_interval (optional)

Use this field to specify the number of seconds after which the WAL scanner should scan the new WAL files. The default value is 0, which means no brute-force scanning will be started.

mbm_scan_timeout (optional)

Use this field to specify the number of seconds to wait for MBM files before timing out; the default value is 20 seconds. The value must be greater than 0. The `mbm_scan_timeout` parameter is applicable only for incremental backup.

workers (optional)

Use this field to specify the number of parallel worker processes required to stream the modified blocks of an incremental backup to the restore host. The default value is 1.

5. Invoke the `CHECK-CONFIG` subcommand omitting the `-s` option to check the parameter settings in the BART configuration file including `bart_host`, `backup_path`, and `pg_basebackup_path`.
6. Set the following parameters for each database server in the server section of the BART configuration file. The parameter setting in the server section overrides the setting in the global [BART] section for that particular database server.

[ServerName] (required)

Specify a database server name.

backup_name (optional)

Specify user-friendly name for the backups of the database server.

host (required)

Specify the IP address of the database server to be configured for backup.

port (optional)

Specify the port number identifying the database server instance to be backed up. The default is port 5444.

user (required)

Specify the replication database user name used by BART to establish the connection to the database server for full backups.

archive_path (optional) – Specify the path where archived WAL files will be stored. The default location of the archived WAL files is the BART backup catalog (`backup_path/server_name/archived_wals`).

archive_command (optional)

Use this field to specify the desired format of the archive command string to be used in the `bart.cfg` file.

cluster_owner (required)

Specify the Linux operating system user account that owns the database cluster. This is typically `enterprisedb` for Advanced Server clusters installed in compatible mode, or `postgres` for PostgreSQL or Advanced Server clusters installed in the PostgreSQL compatible mode.

remote_host (optional)

Use this field to specify the IP address of the remote server to which a backup is to be restored.

Specify the value in the form of `<remote_user>@<remote_host_address>`
`<remote_user>` is the user account on the target database server host that accepts a passwordless SSH/SCP login connection and owns the directory where the backup is to be restored.
`<remote_host_address>` is the IP address of the remote host where you want to restore the backup.

tablespace_path (optional)

Use this field to specify the path to which tablespaces are to be restored. Specify this parameter value in the `OID = tablespace_path; OID = tablespace_path ...` format. If the backup is to be restored to a remote host (specified by the `remote_host` parameter), then the tablespace must exist on the remote host.

allow_incremental_backups (optional)

Set this parameter to `enabled` to permit incremental backups.

description (optional)

Use this field to specify the database server description.

For detailed information about configuring the following optional parameters in the server section, refer to the *BART Installation and Upgrade Guide* available at:

<https://www.enterprisedb.com/edb-docs/>

- `Retention_policy`
- `xlog_method`
- `wal_compression`
- `copy_wals_during_restore`
- `thread_count`
- `batch_size`
- `scan_interval`
- `mbm_scan_timeout`
- `workers`

Configuring a Database Server

To configure a database server, you must:

1. Authorize SSH/SCP access to the server.
2. Create and configure a replication database user.
3. Enable WAL archiving of the server.
4. Verify the server configuration settings.

Note: You must authorize SSH/SCP access and set up a replication database user before restarting the database server with WAL archiving enabled.

3.1 Authorizing SSH/SCP Access

The following example enables SSH/SCP access on a CentOS 6.x host; similar (platform-specific) steps will apply to other platforms/versions.

1. First, enable public key authentication:
 - (a) In the SSH server daemon configuration file `/etc/ssh/sshd_config`, ensure that the following parameter is set to `yes` and is not commented out:

```
PubkeyAuthentication yes
```
 - (b) Reload the configuration file using the `service sshd reload`, `service sshd stop`, `service sshd start`, or `service sshd restart` command.

Note: If you get any SSH or SCP errors, examine the log file (`/var/log/secure`).

2. Then, execute the following command to create a passwordless connection:

```
ssh-copy-id target_user@host_address
```

For more information about how to generate an authorized public key, see the *Authorized Public Keys Generation* section of the *BART Installation and Upgrade Guide* available at:

<https://www.enterprisedb.com/edb-docs>

3.2 Setting up a Replication Database User

To set up a replication database user:

1. Choose a database user to serve as the *replication database user* (a superuser) for each Postgres database server to be managed by BART.
2. Modify the `pg_hba.conf` file to allow the replication database user to access the `template1` database. Include the replication database user in the `pg_hba.conf` file as a `replication` connection if `pg_basebackup` is to be used for taking any backups.
3. Specify the replication database user for the database server in the BART configuration file in the `user` parameter.

3.3 Enabling WAL Archiving

To enable WAL archiving, set the following parameters in the `postgresql.conf` file for any database server for which BART is to perform a backup.

- Set `wal_level` to `archive` for Postgres 9.5 or to `replica` for Postgres 9.6 or later.
- Set `archive_mode` to `on`.
- Set the PostgreSQL `archive_command` parameter to copy the WAL files to the `archive_path`. The `archive_command` configuration parameter mentioned here is located in the `postgresql.conf` file; the PostgreSQL `archive_command` parameter is used in a different manner than the BART `archive_command` parameter previously mentioned in this guide.
- Set `max_wal_senders` to a value high enough to leave at least one session available for the backup. If the `xlog_method=stream` parameter setting is to be used by this server, the `max_wal_senders` setting must account for an additional session for transaction log streaming.

Note: Run the `INIT` subcommand with the `-o` option to override any existing `archive_command` setting in the `postgresql.conf` or the `postgresql.auto.conf` file.

1. After verifying that the full path of the BART backup catalog has been created, restart the database server to initiate WAL archiving.
2. Start the WAL scanner by executing the following command:

```
./bart-scanner
```

3.4 Verifying Configuration Setting

- Use the `CHECK-CONFIG` subcommand with the `-s` option to verify the parameter settings in the database server configuration for which the `-s` option is specified.

```
bart CHECK-CONFIG [ -s <server_name> ]
```

In addition, the following `postgresql.conf` parameters for the database server must be properly set and activated for certain processes:

- The `cluster_owner` parameter must be set to the user account owning the database cluster directory.
- A passwordless SSH/SCP connection must be set between the BART user and the user account specified by the `cluster_owner` parameter.
- The BART user parameter must specify a database superuser.
- The `pg_hba.conf` file must contain a replication entry for the database superuser specified by the BART user parameter.
- The `archive_mode` parameter in the `postgresql.conf` file must be enabled.
- The `archive_command` parameter in the `postgresql.auto.conf` or the `postgresql.conf` file must be set.
- The `allow_incremental_backups` parameter in the BART configuration file must be enabled for database servers for which incremental backups are to be taken.
- Archiving of WAL files to the `archive_path` must be in process.
- The WAL scanner program must be running.

Taking a Backup

This section provides information about creating a full or incremental backup of a database server. For detailed information about taking a full backup, incremental backup, point-in-time recovery and restore process, see the *BART User Guide* available at:

<https://www.enterprisedb.com/edb-docs>

The syntax of the `BACKUP` subcommand is:

```
bart BACKUP -s { server_name | all }
[ -F { p | t } ]
[ -z ] [ -c compression_level ]
[ --parent { backup_id | backup_name } ]
[ --backup-name backup_name ]
[ --thread-count number_of_threads ]
[ { --with-pg_basebackup | --no-pg_basebackup } ]
[ --check ]
```

Note: While a `BACKUP` subcommand is in progress, no other processes may run in parallel.

Along with the `BACKUP` subcommand, you can:

- Specify the `-s` option and replace the *server_name* with the server name to be backed up (it must be configured in the BART configuration file). Specify `all` to take a backup of all servers. This argument is mandatory.

The backup is saved in the `backup_path/server_name/backup_id` directory.

Specify the following options only if required. If you do not specify any of the following options, the backup is created using the default settings.

- Specify the `-F p` option to create a backup in the plain text format and `-F t` to create a backup in tar format (default). If the transaction log streaming method is used, then the `-F p` option must be specified.
- Specify the `-z` option to use `gzip` compression on the tar file output using the default compression level. This option is applicable only for the tar format.

- Specify the `-c` option to apply the `gzip` compression level on the `tar` file output, and replace `compression_level` with the digit 1 through 9, with 9 being the best compression (applicable only for the `tar` format).
- If you want to take an incremental backup, specify the option `--parent` and replace `backup_id` with the backup identifier of a parent backup or replace `backup_name` with the parent backup name. Incremental backup can only be taken in the plain text format (`-F p`). Specify the option `--check` before taking an incremental backup to verify if the required MBM files are present in the BART backup catalog. The `--parent` option must be specified when the `--check` option is used.
- Specify the option `--backup-name` and replace `backup_name` with the user-friendly name assigned to the backup.
- Specify the option `-thread count` and replace `number_of_threads` with the number of worker threads to run in parallel to copy blocks for incremental backups
- Specify the option `--with-pg_basebackup` to use `pg_basebackup` to take a full backup. The number of thread counts in effect is ignored as given by the `thread_count` parameter in the BART configuration file.

Note: If the thread count in effect is greater than 1, then the `pg_basebackup` utility is not used to take the full backup unless the `--with-pg_basebackup` option is specified with the `BACKUP` subcommand.

- Specify the option `--no pg_basebackup` to not use `pg_basebackup` to take a full backup.

The following example creates a full backup in the default `tar` format with `gzip` compression. Note that checksums are generated for the full backup and user-defined tablespaces for the `tar` format backup.

```
[edb@localhost bin]$ ./bart BACKUP -s hr -z
INFO: DebugTarget - getVar(checkDiskSpace.bytesAvailable)
INFO: new backup identifier generated 1567591909098
INFO: creating 5 harvester threads
NOTICE: all required WAL segments have been archived
INFO: backup completed successfully
INFO:
BART VERSION: 2.5
BACKUP DETAILS:
BACKUP STATUS: active
BACKUP IDENTIFIER: 1567591909098
BACKUP NAME: none
BACKUP PARENT: none
BACKUP LOCATION: /home/edb/bkup_new/hr/1567591909098
BACKUP SIZE: 13.91 MB
BACKUP FORMAT: tar.gz
BACKUP TIMEZONE: America/New_York
XLOG METHOD: fetch
BACKUP CHECKSUM(s): 0
TABLESPACE(s): 3
Oid      Name      Location
16387   test1    /home/edb/tbl1
16388   test2    /home/edb/tbl2
16389   test3    /home/edb/tbl3

START WAL LOCATION: 0000000100000000000000025
STOP WAL LOCATION: 0000000100000000000000026
BACKUP METHOD: streamed
BACKUP FROM: master
START TIME: 2019-09-04 06:11:49 EDT
```

```
STOP TIME: 2019-09-04 06:11:53 EDT
TOTAL DURATION: 4 sec(s)
```

The following example shows an incremental backup taken by specifying the `--parent` option. The option `-F p` must be specified as well for plain text format.

```
[edb@localhost bin]$ ./bart BACKUP -s hr -F p --parent hr_full_1 --backup-name
hr_incr_1
INFO: DebugTarget - getVar(checkDiskSpace.bytesAvailable)
INFO: checking /home/edb/bkup_new/hr/archived_wals for MBM files from 0/20000028 to
0/22000000
INFO: new backup identifier generated 1566899827751
INFO: creating 5 harvester threads
NOTICE: all required WAL segments have been archived
INFO: backup completed successfully
INFO:
BART VERSION: 2.5
BACKUP DETAILS:
BACKUP STATUS: active
BACKUP IDENTIFIER: 1566899827751
BACKUP NAME: hr_incr_1
BACKUP PARENT: 1566899819709
BACKUP LOCATION: /home/edb/bkup_new/hr/1566899827751
BACKUP SIZE: 7.19 MB
BACKUP FORMAT: plain
BACKUP TIMEZONE: America/New_York
XLOG METHOD: fetch
BACKUP CHECKSUM(s): 0
TABLESPACE(s): 0
START WAL LOCATION: 0000000100000000000000022
STOP WAL LOCATION: 0000000100000000000000023
BACKUP METHOD: streamed
BACKUP FROM: master
START TIME: 2019-08-27 05:57:07 EDT
STOP TIME: 2019-08-27 05:57:08 EDT
TOTAL DURATION: 1 sec(s)
```

EDB Postgres Backup and Recovery QuickStart Guide

Copyright © 2007 - 2020 EnterpriseDB Corporation.

All rights reserved.

EnterpriseDB® Corporation

34 Crosby Drive, Suite 201, Bedford, MA 01730, USA

T +1 781 357 3390 F +1 978 467 1307 E

info@enterprisedb.com

www.enterprisedb.com

- EDB designs, establishes coding best practices, reviews, and verifies input validation for the logon UI for EDB Postgres product where present. EDB follows the same approach for additional input components, however the nature of the product may require that it accepts freeform SQL, WMI or other strings to be entered and submitted by trusted users for which limited validation is possible. In such cases it is not possible to prevent users from entering incorrect or otherwise dangerous inputs.
- EDB reserves the right to add features to products that accept freeform SQL, WMI or other potentially dangerous inputs from authenticated, trusted users in the future, but will ensure all such features are designed and tested to ensure they provide the minimum possible risk, and where possible, require superuser or equivalent privileges.
- EDB does not warrant that we can or will anticipate all potential threats and therefore our process cannot fully guarantee that all potential vulnerabilities have been addressed or considered.

C

Conclusion, 14