



BART Installation and Upgrade Guide

Release 2.5.3

BART Installation and Upgrade Guide

May 28, 2020

Contents

1	Requirements Overview	1
1.1	Supported Platforms and Database Versions	1
1.2	Software Requirements	1
1.3	Limitation	1
2	Installing BART	2
2.1	Using an RPM Package to Install BART on a CentOS or RHEL Host	2
2.2	Installing BART on a Debian or Ubuntu Host	5
2.3	Installing BART on an SLES 12 Host	6
3	Configuring BART	7
3.1	Establishing the BART User Account	7
3.2	Configuring the BART Host	8
3.2.1	Configuration Parameter Reference	9
3.3	Configuring the Database Server	14
3.3.1	Authorizing SSH/SCP Access	14
3.3.2	Setting up a Replication Database User	16
3.3.3	Adding a Database Server to the BART Configuration File	17
3.3.3.1	Database Server Parameter Reference	17
3.3.4	Enabling WAL Archiving	20
3.3.5	Verifying Configuration Settings	25
4	Installation Troubleshooting	26
5	Performing a BART Upgrade	27
5.1	Upgrading from Older Versions of BART (except 2.0)	28
5.2	Upgrading from BART 2.0	29
6	Uninstallation	30
7	Conclusion	31
	Index	32

1.1 Supported Platforms and Database Versions

- To view a complete list of platforms supported by BART, visit the [EnterpriseDB website](#).
- BART supports the following database versions:
 - Advanced Server versions 9.5, 9.6, 10, 11, 12
 - PostgreSQL versions 9.5, 9.6, 10, 11, 12

1.2 Software Requirements

The following components are required for BART installation.

- BART Host Components - Use EnterpriseDB packages to add BART host components; see *Installing BART using an RPM Package* for installation instructions.
- Additional Components - In addition to the BART host components, the following components are required:
 - The *Secure Shell (SSH) server daemon and Secure Copy (SCP) client programs* must be enabled and activated on the BART host as well as on the remote database server hosts on which BART will be managing backup and recovery.
 - BART uses the `pg_basebackup` utility program when taking full backups.

1.3 Limitation

BART supports taking only a full backup of standby servers; it does not support taking incremental or parallel backups of standby servers.

Installing BART

This section will walk you through performing a fresh installation of BART on a host. Installation instructions are organized into the following platform/installer specific sections:

- *Using an RPM Package to Install BART on a CentOS or RHEL Host*
- *Installing BART on a Debian or Ubuntu Host*
- *Installing BART on an SLES 12 Host*

2.1 Using an RPM Package to Install BART on a CentOS or RHEL Host

The following section demonstrates installing BART 2.5.3 on a CentOS 7 host using an RPM package. The tutorial assumes that the user has some knowledge of installation and system administration procedures, and administrative privileges on the host.

1. On a CentOS7 host, the *Extra Packages for Enterprise Linux (EPEL)* package, which contains supporting boost libraries required by BART is installed by default. On a CentOS6/RHEL6 OS, you need to install the EPEL package by assuming the superuser privileges and invoking the following command:

```
yum install -y epel*
```

2. Use either yum or rpm to create the repository configuration file.

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

or

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

3. Visit [this page at the EnterpriseDB website](#) to request credentials to the EDB Yum Repository.

4. Use your choice of editor to open and modify the repository configuration file. The repository configuration file is named `edb.repo` and is located in the `/etc/yum.repos.d` directory.
5. After creating the `edb.repo` file,
 - ensure the value of the `enabled` parameter is `1`.
 - replace the `username` and `password` placeholders in the `baseurl` specification with the name and password of a registered EnterpriseDB user.
 - save the configuration file and exit the editor.

```
[edb]
name=EnterpriseDB RPMs $releasever - $basearch
baseurl=https://<username>:<password>@yum.enterprisedb.com/edb/redhat/
rhel-$releasever-$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/ENTERPRISEDB-GPG-KEY
```

6. Optionally, install the `pg_basebackup` utility program using the server client package. If you do not already have the `pg_basebackup` program installed on the BART host, you can install a limited number of files that include the `pg_basebackup` program by invoking the following command:

```
yum install edb-asxx-server-client
```

In the above command, replace `xx` with the required Advanced Server version. The `pg_basebackup` version must be the same or more recent than the database server to be backed up. For example, `pg_basebackup` version 10 can be used to back up database server version 10, but cannot be used to back up database server version 11.

7. Use the following command to install the BART RPM package:

```
yum install edb-bart
```

8. Repeat the installation process described in this section to install BART on each remote host on which an incremental backup is to be restored.
9. To verify the BART installation, navigate to the `/usr/bart/bin` directory and execute the following command:

```
bart --version
```

The `bart --version` command should return the current BART version. If the `bart --version` command returns an error stating the `PATH` is not available after switching from the root user to another BART user account, adjust the setting of the `PATH` environment variable to include the directory location of the BART `bin` subdirectory in the `~/.bashrc` or `~/.bash_profile` files of the following user accounts:

- The BART user account on the BART host. See *Configuring BART* for details.
- The remote user account on the remote host to which incremental backups are to be restored. For details, see the *EDB Postgres Backup and Recovery User Guide* available at the [EDB website](#).

Upon successful installation, BART is installed in the `BART_HOME` directory:

```
/usr/edb/bart
```

The installation includes the following files:

File Name	Location	Description
bart	<BART_HOME>/bin	BART command line, executable program
bart-scanner	<BART_HOME>/bin	BART WAL scanner program
bart.cfg.sample	<BART_HOME>/etc	Sample BART configuration file
xlogreader_ident.so	<BART_HOME>/lib	Libraries supporting WAL versions
bart_license.txt	<BART_HOME>	License agreement

After BART is installed successfully, you need to *configure the installation*.

2.2 Installing BART on a Debian or Ubuntu Host

To install BART on a Debian or Ubuntu host, you must have credentials that allow access to the EnterpriseDB repository. To request credentials for the repository, [visit this page](#).

Perform the following steps to install a Debian package using the EnterpriseDB apt repository.

1. Assume the superuser privileges.

```
sudo su -
```

2. Configure the EnterpriseDB repository; substitute your EnterpriseDB credentials for the username and password placeholders in the following command:

```
sh -c 'echo "deb https://username:password@apt.enterprisedb.com/$(lsb_release -cs)-edb/ $(lsb_release -cs) main" > /etc/apt/sources.list.d/edb-$(lsb_release -cs).list'
```

3. Add support to your system for secure APT repositories.

```
apt-get install apt-transport-https
```

4. Add the EBD signing key; When invoking the command, replace the username and password with the credentials provided by EnterpriseDB.

```
wget -q -O - https://username:password@apt.enterprisedb.com/edb-deb.gpg.  
key | apt-key add -
```

5. Update the repository metadata.

```
apt-get update
```

6. Install the Debian package.

```
apt-get install edb-bart
```

2.3 Installing BART on an SLES 12 Host

This section provides instructions for installing BART on an SLES 12 SP4 host using the zypper package manager.

Note: BART is supported on SLES SP4 and SP5 versions.

1. Assume superuser privileges and use the following command to add the EnterpriseDB repository configuration file to your SLES host:

```
zypper addrepo https://zypp.enterprisedb.com/suse/edb-sles.repo
```

This command creates a repository configuration file named `edb.repo` in the `/etc/zypp/repos.d` directory.

2. Use the zypper utility to install BART.

```
zypper install edb-bart
```


To configure BART, you must identify the BART user account, *configure the BART host*, and *configure the database server* that will be backed up.

3.1 Establishing the BART User Account

The BART user account is an operating system user that will run the BART command line program. The BART user account must:

- own the BART backup catalog.
- be able to run the `bart` program and the `bart-scanner` program.
- be able to establish a SSH/SCP connection to and from each database server managed by BART.

You can optionally use the `enterprisedb` database user as the BART user account for an Advanced Server database and use the `postgres` database user as the BART user account for a PostgreSQL server. If you do not wish to use an existing database user as the BART user account, you must create an operating system user to assume the role.

3.2 Configuring the BART Host

This section describes the configuration steps that must be performed on the BART host; these steps must be performed as a root user.

Step 1. Navigate to the `usr/edb/bart/etc` directory and make a copy of the `bart.cfg.sample` file to create the `bart.cfg` file that will contain the parameter settings.

Step 2. Confirm that the Postgres `pg_basebackup` utility program is installed on the BART host. The `pg_basebackup` utility resides in the `bin` directory under your Postgres installation.

Step 3. Ensure the `LD_LIBRARY_PATH` environment variable includes the location of the `libpq` library. If your `libpq` library does not reside in the default location (`POSTGRES_INSTALL_HOME/lib`), you must add the library path to the `LD_LIBRARY_PATH` environment variable in the BART user account's profile (`bash_profile`) located in `/home/<bart user account>`.

```
# .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
```

Step 4. Create the BART backup catalog and ensure the BART user account holds privileges on the BART backup catalog. In the following example, the BART configuration file specifies `/opt/backup` as the parent directory for the BART backup catalog in the `<backup_path>` parameter:

```
[BART]

bart_host = bartuser@192.168.2.22
backup_path = /opt/backup
pg_basebackup_path = /usr/edb/as11/bin/pg_basebackup
logfile = /tmp/bart.log
scanner_logfile = /tmp/bart_scanner.log
```

In the following example, `bartuser` is the BART user account. The example creates and sets the ownership and permissions on the BART backup catalog:

```
su root
mkdir /opt/backup
chown bartuser /opt/backup
chgrp bartuser /opt/backup
chmod 700 /opt/backup
```

If the subdirectory does not exist, BART creates a subdirectory for each database server listed in the configuration file when you invoke the `bart` command line program.

Step 5. Use your choice of editor to open the BART configuration file (located in the `usr/edb/bart/etc` directory) and edit the configuration as required. You must add the mandatory parameters to the `[BART]` and `[ServerName]` sections (for example, `[EPAS11]`). Default values may be used for optional parameters.

Step 6 Invoke the `CHECK-CONFIG` subcommand, omitting the `-s` option to check the parameter settings in the BART configuration file. The `CHECK-CONFIG` subcommand displays an error message if the required configuration is not properly set.

For information about server section parameters, see *database server parameters*.

3.2.1 Configuration Parameter Reference

In the table that follows:

- Parameters set in the [BART] section are applicable to all BART managed database servers.
- Parameters set in the Server section are applicable only to the specific server; the Server parameter setting overrides the [BART] section setting.

Parameter	Type	Default	Server	[BART]
[BART]	Mandatory	N/A	N/A	yes
<bart_host>	Mandatory	N/A	N/A	Yes
<backup_path>	Mandatory	N/A	N/A	Yes
<pg_basebackup_path>	Mandatory	N/A	N/A	Yes
retention_policy	Optional	<max_number>BACKUPS	Yes	Yes
wal_compression	Optional	Disabled	Yes	Yes
copy_wals_during_restore	Optional	Disabled	Yes	Yes
xlog_method	Optional	fetch	Yes	Yes
logfile	Optional	/tmp/bart.log	N/A	Yes
scanner_logfile	Optional	/tmp/bart_scanner.log	N/A	Yes
<bart_socket_directory>	Optional	/tmp	N/A	Yes
<thread_count>	Optional	1	Yes	Yes
<batch_size>	Optional	49152	Yes	Yes
<scan_interval>	Optional	0	Yes	Yes
<mbm_scan_timeout>	Optional	20 seconds	Yes	Yes
<workers>	Optional	1	Yes	Yes
[Server Name]	Mandatory	N/A	Yes	N/A
<backup_name>	Optional	N/A	Yes	N/A
host	Mandatory	N/A	Yes	N/A
port	Mandatory	5444 for EPAS; 5432 for Postgres	Yes	N/A
user	Mandatory	N/A	Yes	N/A
<archive_path>	Optional	BART backup catalog	Yes	N/A
<archive_command>	Optional	N/A	Yes	N/A
<cluster_owner>	Mandatory	enterprisedb for EPAS postgres for PostgreSQL	Yes	N/A
<remote_host>	Optional	N/A	Yes	N/A
<tablespace_path>	Optional	N/A	Yes	N/A
allow_incremental_backup	Optional	Disabled	Yes	N/A
description	Optional	N/A	Yes	N/A

The following table describes the [BART] host parameters.

Parameters/Placeholder	Description
[BART] (mandatory)	Identifies the global section of the configuration file (it must be named BART).
bart_host (mandatory)	Specify the bart user name and the IP address of the bart host on which the BART utility resides, in the form of <bart_user>@<bart_host_address>.
backup_path (mandatory)	Specify the path to the file system parent directory where all BART backups are stored.

Continued on next page

Table 3.1 – continued from previous page

Parameters/Placeholder	Description
<code>pg_basebackup_path</code> (mandatory)	Specify the path to the <code>pg_basebackup</code> program that you installed on the BART host. For information about <code>pg_basebackup</code> version-specific restrictions, see the EDB Postgres Backup and Recovery User Guide .
<code>wal_compression</code> (optional)	Set this parameter to <code>enabled</code> to compress the archived WAL files in <code>gzip</code> format in the BART backup catalog when the <code>MANAGE</code> subcommand is invoked. By default it is set to <code>disabled</code> . The <code>gzip</code> compression program must be in the BART user account's <code>PATH</code> and the WAL compression setting must not be enabled for those database servers where you need to take incremental backups.
<code>copy_wals_during_restore</code> (optional)	Set this parameter to <code>enabled</code> to copy the archived WAL files from the BART backup catalog to the <code>restore_path/archived_wals</code> directory prior to the database server archive recovery. Enabling this option helps you save time during the restore operation. Set this parameter to <code>disabled</code> (default) to retrieve the archived WAL files directly from the BART backup catalog during the database server archive recovery. During the restore operation, recovery settings will be saved in the <code>postgresql.auto.conf</code> file. The <code>restore_command</code> in the <code>postgresql.auto.conf</code> file will be determined by the value specified in the <code>copy_wals_during_restore</code> parameter. If the <code>RESTORE</code> subcommand is invoked with the <code>-c</code> option, the archived WAL files are copied from the BART backup catalog to the <code>restore_path/archived_wals</code> directory, thus overriding any setting of the <code>copy_wals_during_restore</code> parameter. If the <code>RESTORE</code> subcommand is invoked without the <code>-c</code> option, the value specified by the <code>copy_wals_during_restore</code> parameter is used.
<code>xlog_method</code> (optional)	Specify how the transaction log is collected during the execution of <code>pg_basebackup</code> through the <code>BACKUP</code> subcommand. Set <code>xlog_method</code> to <code>fetch</code> (default) to collect the transaction log files after the backup is completed. Set to <code>stream</code> to stream the transaction log in parallel with the full backup creation.

Continued on next page

Table 3.1 – continued from previous page

Parameters/Placeholder	Description
retention_policy (optional)	<p>Set this parameter to determine when an active backup should be marked as <code>obsolete</code> when the <code>MANAGE</code> subcommand is used.</p> <p>You can specify the retention policy either in terms of number of backups or duration (days, weeks, or months). <code><max_number> BACKUPS</code> (default), <code><max_number> DAYS</code>, <code><max_number> WEEKS</code>, or <code><max_number> MONTHS</code> where <code><max_number></code> is a positive integer.</p> <p>For information about managing backups using a retention policy, see the EDB Postgres Backup and Recovery User Guide.</p>
logfile (optional)	<p>Use this parameter to specify the path to the BART log file. The default log file location is <code>/tmp/bart.log</code>.</p> <p>The log file will be created the first time you invoke the <code>bart</code> command line program using the sample configuration file value. To change the default setting, you must delete the <code>bart.log</code> file from the <code>/tmp</code> directory and create a new log file in another directory so that a new log file will be created and owned by the new BART user account.</p> <p>If no path to a log file is specified, BART does not create a log file.</p>
scanner_logfile (optional)	<p>Use this parameter to specify the path to the XLOG/WAL scanner log file. The default location is <code>/tmp/bart_scanner.log</code>.</p> <p>The scanner log file will be created the first time you invoke the <code>bart_scanner</code> program using the sample configuration file value. To change the default setting, you must delete the <code>bart_scanner.log</code> file from the <code>/tmp</code> directory and create a new log file in another directory so that a new log file will be created and owned by the new BART user account.</p> <p>If no path to a log file is specified, BART does not create a WAL scanner log file.</p>
<bart_socket_directory> (optional)	<p>Specify the socket directory path where all BART sockets will be stored. The default directory is <code>/tmp</code>. While specifying the <code>bart_socket_directory</code> path, you must ensure that the directory exists and the BART user has the required access permissions to the directory.</p>

Continued on next page

Table 3.1 – continued from previous page

Parameters/Placeholder	Description
<thread_count> (optional)	<p>Specify the number of worker threads for copying blocks (for incremental backups) or data files (for full backup) from the database server to the <code>archive_path</code> when the <code>BACKUP</code> subcommand is invoked. The default value is 1.</p> <p>The same set of worker threads are used for the compression operation when taking full backups in order to provide parallel, compressed backups when the <code>BACKUP</code> subcommand is specified with the <code>-z</code> or <code>-c</code> options. The compression operation does not apply to incremental backups. See <i>thread count</i> for more information.</p>
<batch_size> (optional)	<p>Specify the number of blocks of memory used for copying modified blocks from the database server to the <code>archive_path</code> when the <code>BACKUP</code> subcommand is invoked for incremental backups.</p> <p>The default value is 49152 blocks; each block is 8192 bytes. The maximum permitted value is 131072 blocks and the minimum permitted value is 1 block. Reduce the <code><batch_size></code> setting if the server runs out of memory while executing the <code>pg_read_binary_file()</code>.</p>
<scan_interval> (optional)	<p>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.</p>
<mbm_scan_timeout> (optional)	<p>Specify the number of seconds to wait for MBM files before timing out; This parameter is applicable only for incremental backup.</p> <p>The default value is 20 seconds. The <code>mbm_scan_timeout</code> parameter value must be greater than 0. If the value is 0 or negative, then an error will be displayed during an incremental backup.</p>
<workers> (optional)	<p>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.</p>

Thread Count

If the `BACKUP` subcommand is invoked with the `--thread-count` option, then the number of worker threads specified by this option overrides any setting of the `thread_count` parameter in the BART configuration file. If the `BACKUP` subcommand is invoked without the `--thread-count` option, then the following determines the number of worker threads used:

- The setting of the `thread_count` parameter in the server section of the BART configuration file overrides the setting of `thread_count` in the global section for that particular database server.
- If omitted in the server section, the setting of `thread_count` in the global section is used.
- If the `thread_count` parameter is not specified in either section, the default is 1.
- When taking a full backup, if the `thread_count` in effect is only 1, then the `pg_basebackup` utility is used to take the full backup unless the `--no-pg_basebackup` option is specified with the `BACKUP` subcommand.

`<thread_count>` will not be effective if the backup is taken on a standby server.

If parallel backup is run with `N` number of worker threads, then it will initiate `N + 1` concurrent connections with the server.

3.3 Configuring the Database Server

This section describes the procedure for enabling BART backup and recovery management for a database server. To configure the database server, you need to:

- Authorize SSH/SCP access without a password prompt.
- *Create and configure a replication database user.*
- *Update the BART configuration file (server section).*
- *Enable WAL archiving of the server.*
- *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.

3.3.1 Authorizing SSH/SCP Access

BART uses the Secure Shell (`ssh`) and Secure Copy (`scp`) Linux utility programs to copy the backup and WAL files from the BART managed database servers to the BART host as well as to restore backups.

- The client/server `ssh` and `scp` commands must not prompt for a password when establishing a connection with the target server (the server to which a passwordless connection is being made).
- A passwordless connection uses *authorized public keys* (public key of a client user account) to authenticate with the target server.
- You must add the public key of each client user account to the target user account's authorized public keys list on the target server.

BART Connections that Require Authentication without a Password

For BART usage, there are two scenarios that require a passwordless SSH/SCP connection:

- When connecting from each BART managed database server (SSH/SCP client) to the BART host (target SSH/SCP server) to support WAL archiving as implemented by the `archive_command` parameter.
 - In this case, the database server user account should generate the public key file (`id_rsa.pub`) with the `ssh-keygen -t rsa` command on the database server host.
 - The public key file name should be appended to the `~/.ssh/authorized_keys` file on the BART host. The `authorized_keys` file is in the BART user account's home directory.
- When connecting from the BART host (SSH/SCP client) to each BART managed database server (target SSH/SCP server) for taking incremental backups and for supporting restoration of the full backup, the archived WAL files, and the modified blocks, which occurs when the `BART RESTORE` subcommand is given.
 - In this case, the BART user account should generate the public key file (`id_rsa.pub`) with the `ssh-keygen -t rsa` command on the BART host.
 - The public key file name should be appended to the `~/.ssh/authorized_keys` file on the database server host. The `authorized_keys` file is in the home directory of the user account that owns the directory where the database backup is to be restored.
- If backups are to be taken from a given database server host, but restored to a different database server host, the passwordless SSH/SCP connections must be configured from the BART host to the database server host from

which the backup is to be taken as well as from the BART host to the database server host to which the backup is to be restored.

See the [EDB Postgres Backup and Recovery Reference Guide](#) to view examples of creating a passwordless connection.

Enabling Public Key Authentication

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

1. In the SSH server daemon configuration file (`sshd_config`) located in the `/etc/ssh`, set the `PubkeyAuthentication` parameter to `yes`.
2. Reload the configuration file:

```
service sshd reload
```

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

Creating a Passwordless Connection

The following general instructions will walk you through generating a client's public key file, creating the target server's authorized public keys file, and creating a passwordless connection.

Step 1. On the client system, log in as the user account that will be initiating the SSH or SCP connection.

Step 2. Navigate to the user account's home directory and check for an existing `.ssh` subdirectory. If the `.ssh` directory does not exist, create one and assign the required privileges to the user.

Step 3. Generate the public key file with the following command. Accept all prompted defaults and do not specify a passphrase when prompted for one.

```
ssh-keygen -t rsa
```

The public key file named `id_rsa.pub` is created in the `.ssh` subdirectory.

Step 4. While logged into the client where you just generated the public key file, use SCP to make a temporary copy of it on the target server:

```
scp ~/.ssh/id_rsa.pub <target_user>@<host_address>:tmp.pub
```

Step 5. Navigate into the target user account's home directory and check for an existing `.ssh` subdirectory. If it does not exist, create one and assign the required privileges to the user.

Step 6. Append the temporary, client's public key file, `tmp.pub`, to the `authorized_keys` file. If an `authorized_keys` file does not exist, create a new file, but do not completely replace any existing `authorized_keys` file.

```
cat tmp.pub >> ~/.ssh/authorized_keys
```

Make sure the `authorized_keys` file is only accessible by the file owner and not by groups or other users. If the `authorized_keys` file does not have the required permission setting or it was newly created, change the file permissions as follows:

```
chmod 600 ~/.ssh/authorized_keys
```

Step 7. Delete the temporary public key file:

```
rm tmp.pub
```

Now, when logged into the client system as `user` there should be no prompt for a password when commands such as the following is given:

```
ssh target_user@host_address
```

3.3.2 Setting up a Replication Database User

For each database server that is to be managed by BART, a database user must be chosen to serve as the *replication database user*. The replication database user sets the Postgres `archive_command` configuration parameter when the `INIT` subcommand is invoked and creates backups when the `BACKUP` subcommand is invoked. The replication database user must be a superuser.

When executed with the PSQL client, the following PostgreSQL command creates a superuser to be the replication database user:

```
CREATE ROLE repuser WITH LOGIN SUPERUSER PASSWORD 'password';
```

The `pg_hba.conf` file must minimally permit the replication database user to have access to the database.

In the following example, the `pg_hba.conf` file permits the `repuser` (replication database user) to have access to the `template1` database. The IP address from which `repuser` has access to `template1` database is the location of the BART host:

For `pg_basebackup` only: If `pg_basebackup` is to be used for taking any backups (such as for standby servers), the replication database user must also be included in the `pg_hba.conf` file as a replication database connection as shown by the last entry in the following example.

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all md5
# IPv4 local connections:
host template1 repuser 192.168.2.22/32 md5
host all enterisedb 127.0.0.1/32 md5
# IPv6 local connections:
host all all ::1/128 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
host replication repuser 192.168.2.22/32 md5
```

The replication database user must be specified for the `user` parameter in the BART configuration file for the database server as shown in the following example:

```
[ACCTG]
host = 192.168.2.24
port = 5444
user = repuser
cluster_owner = enterisedb
remote_host = enterisedb@192.168.2.24
description = "Accounting"
```

There must be no password prompt when connecting to the database server with the replication database user. There are several ways to permit this; one recommended method is to use a `.pgpass` file located in the BART user account's home directory.

For example, if `bartuser` is the BART user account, then the `.pgpass` file located in the `/home/bartuser` directory must contain the following entry:

```
192.168.2.24:5444::repuser:password
```

When `bartuser` invokes a BART backup, the password for the replication database user, `repuser`, is obtained from the `.pgpass` file of `bartuser` to connect to the database server running at `192.168.2.24` on port `5444`.

The `.pgpass` file must contain an entry for each BART managed database server and its corresponding replication database user and password.

3.3.3 Adding a Database Server to the BART Configuration File

To manage the backup and recovery of a database server, you must add entries to the server section of the BART configuration file (located in `<BART_HOME>/etc/bart.cfg`). Settings in the server section will override the settings in the `[BART]` section for that particular database server. If omitted, default values will be used.

For each cluster serviced by BART, the following parameters are required:

```
[HR]

host = 192.168.2.24
port = 5432
user = postgres
cluster_owner = postgres
```

Note: The port parameter setting is required only if the database server listens on a port other than the default (for example if Postgres listens on a port other than 5432).

3.3.3.1 Database Server Parameter Reference

Set the following parameters 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. If omitted, the default value will be used.

The following table describes the database server parameters.

Parameters/Placeholder	Description
<code>[ServerName]</code> (mandatory)	Specify the server name that you want to backup using BART. It is not case-sensitive when referenced with BART subcommand options. A lowercase conversion of this name is used to create a subdirectory in the BART backup catalog for storing the backups and WAL files for this database server (for eg., <code>epas12</code>).
<code><backup_name></code> (optional)	Specify a <i>template</i> for user-defined, friendly names that will be assigned to the backups of the database server. The maximum permitted length of backup name is 49 characters. The <code><backup_name></code> parameter can be overridden by the <code>--backup-name</code> option of the <code>BACKUP</code> subcommand. If this parameter is omitted from the BART configuration file, and the <code>--backup-name</code> option with a user-defined name is not specified with the <code>BACKUP</code> subcommand, then the backup can only be referenced in BART subcommands by the BART assigned, integer backup identifier.

Continued on next page

Table 3.2 – continued from previous page

Parameters/Placeholder	Description
host (mandatory)	Specify the IP address of the database server to be configured for backup.
port (mandatory)	Specify the port number identifying the database server instance (that is, the relevant database cluster) to be backed up. The default port number for EPAS is 5444 and for Postgres it is 5432. The port parameter setting is only required if the database server listens on a port other than the default value.
User (mandatory)	Specify the replication database user name used by BART to establish the connection to the database server for full backups. See <i>Setting up a Replication Database User</i> for more information.
<archive_path> (optional)	Specify the path where archived WAL files will be stored. The default location is the BART backup catalog (<backup_path>/<server_name>/archived_wals).
<archive_command> (optional)	<p>When the INIT subcommand is used, the content and variables specified in the BART <archive_command> result in the archive command string to be generated into the Postgres archive_command parameter in the postgresql.auto.conf file.</p> <p>To configure the BART <archive_command> parameter, enclose the command string within single quotes (''). If you do not specify the <archive_command> parameter in the configuration file, the default setting is taken as 'scp %p %h:%a/%f'. See <i>Archive Command Auto Configuration</i> for information about variables.</p> <p>The BART <archive_command> parameter in the BART configuration file, and the Postgres <archive_command> parameter in the postgresql.conf file (or the postgresql.auto.conf file) refer to two different parameters that are to be set in different manners.</p>
<cluster_owner> (required)	Specify the Linux operating system user account that owns the database cluster. This is typically enterprisedb for Advanced Server database clusters installed in the Oracle compatible mode, or postgres for Advanced Server database clusters installed in the PostgreSQL compatible mode and PostgreSQL database clusters.

Continued on next page

Table 3.2 – continued from previous page

Parameters/Placeholder	Description
<remote_host> (optional)	Specify the IP address of the remote server to which a backup is to be restored. Specify this parameter 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. For restoring a backup to a remote host or for restoring any backup where <remote_user> and the BART user account are not the same users, either this parameter must be set or it may be specified with the -r option with the BART RESTORE subcommand.
<tablespace_path> (optional)	Specify path to which tablespaces are to be restored in the format OID = <tablespace_path>; If the backup is to be restored to a remote host specified by the <remote_host> parameter, then the tablespace paths must exist on the remote host.
allow_incremental_backups (optional)	Set this parameter to enabled to enable use of the WAL scanner and permit taking incremental backups when the BACKUP subcommand is invoked with the --parent option. Set it to disabled (default) to disallow incremental backups and thus permit only full backups. For information about using the BACKUP subcommand and running the WAL scanner, please see the EDB Postgres Backup and Recovery User Guide available at this page .
Description (optional)	Specify the description that will be used to identify the database server.

For information regarding how to configure the following parameters, see *configuring the BART host*.

- retention_policy
- xlog_method
- wal_compression
- copy_wals_during_restore.
- thread_count.
- batch_size.
- scan_interval.
- mbm_scan_timeout.
- workers

Backup Name Template

- The template is an alphanumeric string that may include the following variables that will be replaced with the timestamp values when the backup is taken:
 - %year to be replaced by 4-digit year

- %month to be replaced by 2-digit month
 - %day to be replaced by 2-digit day
 - %hour to be replaced by 2-digit hour
 - %minute to be replaced by 2-digit minute
 - %second to be replaced by 2-digit second
- To include a percent sign (%) as a character in the backup name, specify %% in the template.
 - Do not enclose the template string in quotes even if you want the template to include space characters, otherwise the enclosing quotes are stored as part of the backup name. However, when referenced with the `-i` option by BART subcommands use of space characters in the backup name requires enclosing the backup name in quotes.

The following example shows the configuration settings of three database servers:

```
[ACCTG]

host = 127.0.0.1
port = 5444
user = enterprisedb
cluster_owner = enterprisedb
backup_name = acctg_%year-%month-%dayT%hour:%minute:%second
archive_command = 'cp %p %a/%f'
allow_incremental_backups = enabled
retention_policy = 8 BACKUPS
description = "Accounting"

[MKTG]

host = 192.168.2.24
port = 5444
user = repuser
cluster_owner = enterprisedb
remote_host = enterprisedb@192.168.2.24
allow_incremental_backups = enabled
description = "Marketing"

[HR]

host = 127.0.0.1
port = 5432
user = postgres
cluster_owner = postgres
retention_policy = 4 DAYS
description = "Human Resources"
```

3.3.4 Enabling WAL Archiving

WAL archiving must be enabled for the database server for which BART is to perform backup and recovery management.

- The WAL Archiving Configuration section describes the manual WAL archiving configuration process.
- The Archive Command Auto Configuration section describes an automated WAL archiving process.

WAL Archiving Configuration

Set the following configuration parameters in the `postgresql.conf` file to enable WAL archiving

- 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`.
- 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 database server, the `max_wal_senders` setting must account for an additional session for the transaction log streaming (the setting must be a minimum of 2). See *Configuring the BART host* for information about the `xlog_method` parameter.

For detailed information about WAL archiving, see the [PostgreSQL Core Documentation](#).

The `ARCHIVE PATH` field displayed by the `BART SHOW-SERVERS` subcommand displays the full directory path where the WAL files should be copied as specified in the `archive_command` configuration parameter in the `postgresql.conf` file:

```
-bash-4.1$ bart SHOW-SERVERS -s acctg
SERVER NAME : acctg
HOST NAME   : 192.168.2.24
USER NAME   : repuser
PORT        : 5444
REMOTE HOST :
RETENTION POLICY : none
DISK UTILIZATION : 0.00 bytes
NUMBER OF ARCHIVES : 0
ARCHIVE PATH : /opt/backup/acctg/archived_wals
ARCHIVE COMMAND : (disabled)
XLOG METHOD   : fetch
WAL COMPRESSION : disabled
TABLESPACE PATH(s) :
INCREMENTAL BACKUP : DISABLED
DESCRIPTION  : "Accounting"
```

The parameter settings in the following example will copy the WAL files to a directory named `/opt/backup/acctg/archived_wals` on the BART host located at `192.168.2.22` as the `bartuser` user account. Using the `bartuser` account ensures that the operation will have sufficient permissions to copy to the BART backup catalog owned by `bartuser`.

```
wal_level = archive          # minimal, archive, or hot_standby
                             # (change requires restart)
...
archive_mode = on           # allows archiving to be done
                             # (change requires restart)
archive_command = 'scp %p bartuser@192.168.2.22:/opt/backup/acctg/archived_wals/%f'
                             # command to use to archive a logfile segment
                             # placeholders: %p = path of file to archive
                             # %f = file name only
...
max_wal_senders = 1         # max number of walsender processes
                             # (change requires restart)
```

The database server must be restarted in order to initiate WAL archiving, but do not do so until you have verified that the full path of the BART backup catalog has been created by some prior BART subcommand or the archive operation will fail.

Start the WAL scanner by executing the following command:

```
./bart-scanner
```

Archive Command Auto Configuration

To enable WAL archiving:

- In the `postgresql.conf` file, set the `wal_level` to `archive` for Postgres 9.5 or to `replica` for Postgres 9.6 or later, set the `archive_mode` to `on`, and 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 database server as determined in the BART configuration file, the `max_wal_senders` setting must account for an additional session for the transaction log streaming (that is, the setting must be a minimum of 2). See [Configuring the BART host](#) for information on the `xlog_method` parameter.
- Configure the Postgres `archive_command` parameter automatically with the `INIT` subcommand and restart the database server when you are ready to initiate WAL archiving. The `INIT` subcommand invokes the Postgres `ALTER SYSTEM` command to set the Postgres `archive_command` configuration parameter in the `postgresql.auto.conf` file located in the managed database server's `POSTGRES_INSTALL_HOME` data directory. For additional information about the `INIT` subcommand, see the [EDB Postgres Backup and Recovery User Guide](#).

The archive command string that the `INIT` subcommand generates into the `postgresql.auto.conf` file is determined by the parameter setting of the BART `archive_command` parameter in the server section of the BART configuration file. If the BART `archive_command` parameter is not set in the server section for a given database server, the command string that is configured uses the following default format:

```
'scp %p %h:%a/%f'
```

The following table describes these variables:

Variable	Description
<code>%p</code>	The path of the file to archive used by the Postgres archiving process.
<code>%h</code>	Will be replaced by the <code><bart_user>@<bart_host_address></code> as specified in the <code><bart_host></code> parameter setting.
<code>%a</code>	Will be replaced by the BART <code>archived_wals</code> directory as specified in the archive path parameter setting. If the <code><archive_path></code> is not specified, then the default directory is <code><backup_path>/<server_name>/archived_wals</code> . <code><server_name></code> is the lowercase conversion of the database server name.
<code>%f</code>	The archived file name used by the Postgres archiving process.

The placeholders `%h` and `%a` are replaced by the `INIT` subcommand when creating the archive command string. The placeholders `%p` and `%f` are not replaced by the `INIT` subcommand, but are kept as given to be used by the Postgres archiving process.

For example, to use the default archive command format, the BART configuration file contains the following settings where the BART `archive_command` parameter is omitted from the server section for ACCTG:

```
[BART]

bart_host= bartuser@192.168.2.22
backup_path = /opt/backup
pg_basebackup_path = /usr/edb/as11/bin/pg_basebackup
logfile = /tmp/bart.log
scanner_logfile = /tmp/bart_scanner.log
```



```
[ACCTG]

host = 127.0.0.1
port = 5444
user = repuser
cluster_owner = enterprisedb
description = "Accounting"
```

The INIT subcommand is invoked by BART user account `bartuser` as follows:

```
[bartuser@localhost ~]$ bart INIT -s acctg -o
INFO: setting archive_command for server 'acctg'
WARNING: archive_command is set. server restart is required
```

If the BART backup catalog directory is not already complete, it will be completed.

The resulting archive command string in the `postgresql.auto.conf` file located in the managed database server's `POSTGRES_INSTALL_HOME/data` directory appears as follows:

```
# Do not edit this file manually!
# It will be overwritten by ALTER SYSTEM command.
archive_command = 'scp %p
bartuser@192.168.2.22:/opt/backup/acctg/archived_wals/%f'
```

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. In addition, the `-o` option must be used to generate the command string if the `archive_mode` is set to `off` even if there are no existing settings of the `archive_command` in the `postgresql.conf` or `postgresql.auto.conf` files.

In this example, the following BART configuration file is used with an explicit setting of the BART `archive_command` parameter:

```
[BART]

bart_host= enterprisedb@192.168.2.22
backup_path = /opt/backup
pg_basebackup_path = /usr/edb/as11/bin/pg_basebackup
logfile = /tmp/bart.log
scanner_logfile = /tmp/bart_scanner.log

[ACCTG]

host = 127.0.0.1
port = 5444
user = repuser
cluster_owner = enterprisedb
archive_command = 'cp %p %a/%f'
description = "Accounting"
```

The INIT subcommand is invoked by BART user account `enterprisedb` as follows:

```
-bash-4.1$ bart INIT -s acctg -o
INFO: setting archive_command for server 'acctg'
WARNING: archive_command is set. server restart is required
```

The resulting Postgres `archive_command` parameter in the `postgresql.auto.conf` file appears as follows:

```
# Do not edit this file manually!  
# It will be overwritten by ALTER SYSTEM command.  
archive_command = 'cp %p /opt/backup/acctg/archived_wals/%f'
```

When the database server has been restarted, the ARCHIVE COMMAND field of the SHOW-SERVERS subcommand displays the active Postgres archive command as shown by the following example:

```
-bash-4.1$ bart SHOW-SERVERS -s acctg  
SERVER NAME : acctg  
HOST NAME : 127.0.0.1  
USER NAME : repuser  
PORT : 5444  
REMOTE HOST :  
RETENTION POLICY : none  
DISK UTILIZATION : 48.00 MB  
NUMBER OF ARCHIVES : 0  
ARCHIVE PATH : /opt/backup/acctg/archived_wals  
ARCHIVE SCOMMAND : `cp %p /opt/backup/acctg/archived_wals/%f`  
XLOG METHOD : fetch  
WAL COMPRESSION : disabled  
TABLESPACE PATH(s) :  
INCREMENTAL BACKUP : DISABLED  
DESCRIPTION : "Accounting"
```

3.3.5 Verifying Configuration Settings

To verify the parameter settings of the database server specified, execute the `CHECK-CONFIG` subcommand with the `-s` option:

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

The `CHECK-CONFIG` subcommand confirms the following:

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

After configuring the BART host and the database server(s), you can start using BART. For information about using BART, see the [EDB Postgres Backup and Recovery User Guide](#).

Installation Troubleshooting

This section provides a workaround for the following installation issue that you may encounter:

```
existing lock /var/run/yum.pid:  another copy is running as pid 3104.  
Another app is currently holding the yum lock; waiting for it to exit
```

To fix this issue, execute the following command:

```
rm -f /var/run/yum.pid
```

Performing a BART Upgrade

This section outlines the process of upgrading BART from an existing version to the latest version.

- *Upgrading from BART 2.0* describes the upgrade process from BART 2.0 to the latest version.
- *Upgrading from Older Versions of BART (except 2.0)* describes the upgrade process from previous BART versions (except 2.0) to the latest version.

Upgrade Restrictions

The following restrictions apply with regard to previous BART versions.

- You can take incremental backups using the latest version only when the parent backup (full or incremental backup) has also been taken with the latest version.
- Using the latest version, you can restore incremental backups that are taken only with the latest version of BART. However, using the latest version you can restore full backups that were taken with older versions.

5.1 Upgrading from Older Versions of BART (except 2.0)

Perform the following steps to upgrade from older versions of BART (except 2.0) to the latest version:

Step 1: Assume the identity of the BART user account and invoke the following command to stop the BART WAL scanner program (`bart-scanner`):

```
bart-scanner STOP
```

Step 2: As the `root` user, upgrade to the latest BART version with the `yum upgrade` command.

- To upgrade the BART RPM package directly from the *EDB Yum Repository* website, specify only the package name:

```
yum upgrade edb-bart
```

You can also use a downloaded RPM package file to upgrade. To use a downloaded BART RPM package file to upgrade, use the `yum` command, specifying the complete RPM package file name:

```
yum upgrade edb-bart-2.5.3 rhel7.x86_64.rpm
```

Step 3: Repeat the process described in this section to upgrade to BART 2.5.3 on each remote hosts where an incremental backup will be restored.

For additional information about restoration of incremental backups on remote hosts, see the *EDB Postgres Backup and Recovery User Guide* available at [this page](#).

Step 4: If the `bart --version` command returns an error stating the `PATH` is not available after switching from `root` user to another BART user account, adjust the setting of the `PATH` environment variable to include the location of the BART 2.5.3 executable (the `bin` subdirectory) in the `~/.bashrc` or `~/.bash_profile` files of the following user accounts:

- The BART user account on the BART host.
- The remote user account on the remote host to which incremental backups are to be restored. For details, see the *EDB Postgres Backup and Recovery User Guide* available at [this page](#).

The `PATH` setting should be the same as set for BART 2.5.3 since all versions use `/usr/edb/bart/bin`.

Note: After upgrading to BART 2.5.3, you must take a new full backup of your system before performing an incremental backup.

5.2 Upgrading from BART 2.0

Perform the following steps to upgrade BART 2.0 to the latest version of BART:

Step 1: Install the latest version of BART. For information about how to install, see *Using an RPM Package to Install BART*.

Step 2: Save a copy of your BART 2.0 configuration file. The default location of the BART 2.0 configuration file is `/usr/edb/bart2.0/etc/bart.cfg`.

Step 3: Invoke the following command to remove BART 2.0:

```
yum remove edb-bart20
```

Step 4: Place the BART 2.0 configuration file (`bart.cfg`) that you saved in Step 2 in the newly created `/usr/edb/bart/etc` directory. You can use many of the same configuration parameters for BART 2.5.3, but note that you must use a new directory for the BART backup catalog. A new set of full backups and incremental backups taken using BART 2.5.3 must be stored in a new BART backup catalog.

To specify an alternative configuration file name or location, use the `-c` option with BART subcommands. For more information about the `-c` option, see the EDB Postgres Backup and Recovery User Guide [available at this page](#).

Note: The `bart.cfg` configuration file is only required on the BART 2.5.3 host from which you will invoke BART subcommands. BART does not require the `bart.cfg` file on hosts on which an incremental backup will be restored.

Step 5: Adjust the setting of the `PATH` environment variable to include the location of the BART 2.5.3 executable (the `bin` subdirectory) in the `~/.bashrc` or `~/.bash_profile` files for the following user accounts:

- The BART user account on the BART host.
- The user account on the remote host to which incremental backups will be restored. For details, see the *EDB Postgres Backup and Recovery User Guide* [available at this page](#).

Step 6: Perform the BART 2.5.3 installation and BART 2.0 removal process on each remote host on which an incremental backup was restored using BART 2.0.

Note: After upgrading to BART 2.5.3, you must take a new full backup of your system before performing an incremental backup.

To uninstall BART, assume the identity of the `root` user and invoke the following command:

```
yum remove edb-bart
```

Uninstalling BART does not delete the backup files and archived WAL files that reside in the BART backup catalog. To permanently delete the backup files and archived WAL files in the BART backup catalog (`/opt/backup`), use one of the following commands:

- `rm -rf /opt/backup`
- `BART DELETE` subcommand

For information about the `BART DELETE` subcommand, refer the EDB Postgres Backup and Recovery User Guide [available at this page](#).

EDB Postgres Backup and Recovery Installation and Upgrade Guide

Copyright © 2014 - 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

- EnterpriseDB and Postgres Enterprise Manager are registered trademarks of EnterpriseDB Corporation. EDB and EDB Postgres are trademarks of EnterpriseDB Corporation. Oracle is a registered trademark of Oracle, Inc. Other trademarks may be trademarks of their respective owners.
- 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.

A

- Adding a Database Server to the BART Configuration File, 17
- Archive_Command_auto_configuration, 22
- Authorizing SSH/SCP Access, 14

B

- BART host components, 1

C

- Conclusion, 31
- Configuring BART, 7
- Configuring the BART Host, 8
- Configuring the Database Server, 14

E

- Enabling WAL Archiving, 20
- Establishing the BART User Account, 7

I

- Installation Troubleshooting, 26
- Installing BART, 2
- Installing BART on a Debian or Ubuntu Host, 5
- Installing BART on an SLES 12 Host, 6

L

- Limitation, 1

P

- Performing a BART Upgrade, 27
- Prerequisites, 1

R

- Requirements Overview, 1

S

- Setting up a Replication Database User, 16
- Software Requirements, 1

- Supported Platforms and Database Versions, 1

U

- Uninstallation, 30
- Upgrading from BART 2.0, 29
- Upgrading from Older Versions of BART (except 2.0), 28
- Using an RPM Package to Install BART on a CentOS or RHEL Host, 2

V

- Verifying Configuration Settings, 25