

Your remote PostgreSQL DBA team

data egret



From Backup to Integrity Leveraging WAL-G for PostgreSQL

Marat Bogatyrev
Senior PostgreSQL DBA, Data Egret

dataegret.com



More about checksum
validation using WAL-G
[in this post.](#)

2 The Importance of Checksum Validation

- **Why is it important?**
 - Checksums help detect corruption due to hardware issues
- **Reminder: Checksums are disabled by default in PostgreSQL***
 - Cluster must be offline during the enabling process, which can take time
- **The tool [pg_checksums](#) allows you to**
 - Enable, disable, or verify checksums (on an offline cluster!)
 - Ensures data integrity and aids early corruption detection

ENABLE CHECKSUM

```
root@pgwal:~# pg_checksums -e -P -D /data/pg_data/postgres
2110/2110 MB (100%) computed
Checksum operation completed
Files scanned: 3860
Blocks scanned: 263931
Files written: 0
Blocks written: 0
pg_checksums: syncing data directory
pg_checksums: updating control file
```

CHECK CHECKSUM (db must be offline)

```
root@pgwal:~# /pg_checksums -c -D /data/pg_data/postgres
pg_checksums: error: could not read block 0 in file "/data/pg_data/postgres/base/16390/16563": read 12 of 8192
```

* Enable data checksums by default in PostgreSQL 18



Checking Backups with WAL-G

VERIFY

```
root@test27-db-01]# usr/local/bin/wal-g backup-push -f -s -v /data/pg_data/postgres*
INFO: 2024/10/08 12:25:52.588586 Selecting the latest backup as the base for the current delta backup...
INFO: 2024/10/08 12:25:52.602764 Doing full backup.
INFO: 2024/10/08 12:25:52.610169 Calling pg_start_backup()
WARNING: 2024/10/08 12:25:52.630169 Corruption found in /data/pg_data/postgres/base/16493/16529/[967], expected 7a4c, found c88b
INFO: 2024/10/08 12:25:52.845646 Starting a new tar bundle
INFO: 2024/10/08 12:25:52.845704 Walking ...
INFO: 2024/10/08 12:25:52.846017 Starting part 1 ...
INFO: 2024/10/08 12:25:52.846118 Starting part 2 ...
INFO: 2024/10/08 12:25:53.081292 Packing ...
INFO: 2024/10/08 12:25:53.081896 Finished writing part 2.
INFO: 2024/10/08 12:26:25.994541 Finished writing part 1.
INFO: 2024/10/08 12:26:26.448139 Starting part 3 ...
INFO: 2024/10/08 12:26:26.448194 /global/pg_control
INFO: 2024/10/08 12:26:26.448730 Finished writing part 3.
INFO: 2024/10/08 12:26:26.449340 Calling pg_stop_backup()
INFO: 2024/10/08 12:27:49.604398 Starting part 4 ...
INFO: 2024/10/08 12:27:49.604576 backup_label
INFO: 2024/10/08 12:27:49.604591 tablespace_map
INFO: 2024/10/08 12:27:49.609611 Finished writing part 4.
INFO: 2024/10/08 12:27:52.810995 Wrote backup with name base_00000001000000000000000B to storage default
```

WAL-VERIFY

```
root@pgwalg:~# wal-g wal-verify integrity
INFO: 2024/10/18 11:10:04.083785 Current WAL segment: 000000010000000500000033
INFO: 2024/10/18 11:10:04.084713 Building check runner: integrity
INFO: 2024/10/18 11:10:04.085560 Detected earliest available backup: base_0000000010000000400000058
INFO: 2024/10/18 11:10:04.085576 Running the check: integrity
[wal-verify] integrity check status: FAILURE
[wal-verify] integrity check details:
```

TLI	START	END	SEGMENTS COUNT	STATUS
1	0000000100000000400000058	00000001000000004000000FB	164	FOUND
1	00000001000000004000000FC	00000001000000004000000FD	2	MISSING LOST
1	00000001000000004000000FE	0000000100000000500000032	53	FOUND

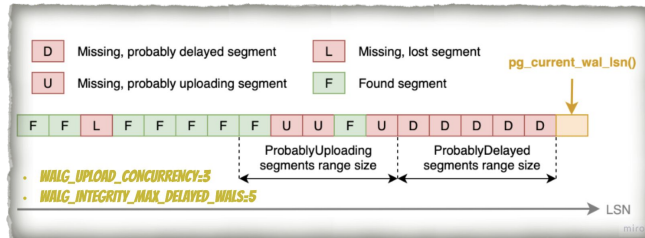
WAL-G OPTIONS

verify:

Validate all data page checksums during backup creation (*analyze and alert WARNING/ERROR messages*)

wal-verify integrity:

Verify the consistency of WAL archives



4 Bug Fix in Checksum Verification

Corrupting the data file

1. Copy the first 121,900 bytes.
2. Add 123 bytes of random data.
3. Copy the rest of the file.
4. Replace the original file.

```
1 dd if=base/16390/16568 of=tempfile bs=121900 count=1
2 dd if=/dev/urandom bs=123 count=1 >> tempfile
3 dd if=base/16390/16568 bs=121900 skip=1 oflag=seek_bytes >> tempfile
4 mv tempfile base/16390/16568
```

CURRENT

```
[root@test27-db-01]# usr/local/bin/wal-g backup-push -f -s -v /data/pg_data/postgres"
INFO: 2024/10/08 12:25:52.588586 Selecting the latest backup as the base for the current delta backup...
INFO: 2024/10/08 12:25:52.602764 Doing full backup.
INFO: 2024/10/08 12:25:52.610169 Calling pg_start_backup()
INFO: 2024/10/08 12:25:52.845646 Starting a new tar bundle
INFO: 2024/10/08 12:25:52.845704 Walking ...
INFO: 2024/10/08 12:25:52.846017 Starting part 1 ...
INFO: 2024/10/08 12:25:52.846118 Starting part 2 ...
INFO: 2024/10/08 12:25:53.081292 Packing ...
INFO: 2024/10/08 12:25:53.081896 Finished writing part 2.
INFO: 2024/10/08 12:26:25.994541 Finished writing part 1.
INFO: 2024/10/08 12:26:26.448139 Starting part 3 ...
INFO: 2024/10/08 12:26:26.448194 /global/pg_control
INFO: 2024/10/08 12:26:26.448730 Finished writing part 3.
INFO: 2024/10/08 12:26:26.449340 Calling pg_stop_backup()
INFO: 2024/10/08 12:27:49.604398 Starting part 4 ...
INFO: 2024/10/08 12:27:49.604576 backup_label
INFO: 2024/10/08 12:27:49.604591 tablespace_map
INFO: 2024/10/08 12:27:49.609611 Finished writing part 4.
INFO: 2024/10/08 12:27:52.810995 Wrote backup with name base_00000001000000000000000B to storage default
```

 **Data file** 
is corrupt
NO ERRORS

Fix
Next Release

```
[root@test27-db-01]# usr/local/bin/wal-g backup-push -f -s -v /data/pg_data/postgres"
INFO: 2024/10/08 12:25:52.588586 Selecting the latest backup as the base for the current delta backup...
INFO: 2024/10/08 12:25:52.602764 Doing full backup.
INFO: 2024/10/08 12:25:52.610169 Calling pg_start_backup()
WARNING: 2024/10/08 12:25:52.910169 /data/pg_data/postgres/base/16390/16568 invalid file size 13434892. File copied without validation. The file may be corrupted.
INFO: 2024/10/08 12:25:52.845646 Starting a new tar bundle
INFO: 2024/10/08 12:25:52.845704 Walking ...
INFO: 2024/10/08 12:25:52.846017 Starting part 1 ...
INFO: 2024/10/08 12:25:52.846118 Starting part 2 ...
INFO: 2024/10/08 12:25:53.081292 Packing ...
INFO: 2024/10/08 12:25:53.081896 Finished writing part 2.
INFO: 2024/10/08 12:26:25.994541 Finished writing part 1.
INFO: 2024/10/08 12:26:26.448139 Starting part 3 ...
INFO: 2024/10/08 12:26:26.448194 /global/pg_control
INFO: 2024/10/08 12:26:26.448730 Finished writing part 3.
INFO: 2024/10/08 12:26:26.449340 Calling pg_stop_backup()
INFO: 2024/10/08 12:27:49.604398 Starting part 4 ...
INFO: 2024/10/08 12:27:49.604576 backup_label
INFO: 2024/10/08 12:27:49.604591 tablespace_map
INFO: 2024/10/08 12:27:49.609611 Finished writing part 4.
INFO: 2024/10/08 12:27:52.810995 Wrote backup with name base_00000001000000000000000B to storage default
```

 **Data file** 
is corrupt
and we see
WARNING



5 Conclusions

MUST HAVE

- 1 Enable checksum validation on DB
- 2 Control your backup process and alert
- 3 Implement a dedicated system for restoring backups and performing checks
- 4 Always use the latest version of WAL-G

