

# Cluster File Encryption in Postgres

BRUCE MOMJIAN



This presentation explains the design of cluster file encryption in Postgres.

*<https://momjian.us/presentations>*



*Creative Commons Attribution License*

*Last updated: June 2024*

# What Is Cluster File Encryption?

Cluster File Encryption (CFE) is a Postgres feature currently under development that will encrypt all user data stored in the file system

- Uses two-levels of encryption
- Cluster-level key (key encryption key) is stored externally
- Data encryption keys are encrypted with the cluster key
- Encrypts only files containing user data, not all files
- Uses AES128, AES192, or AES256
- Heap/index file encryption overhead is 2-4%; WAL encryption will cause additional overhead

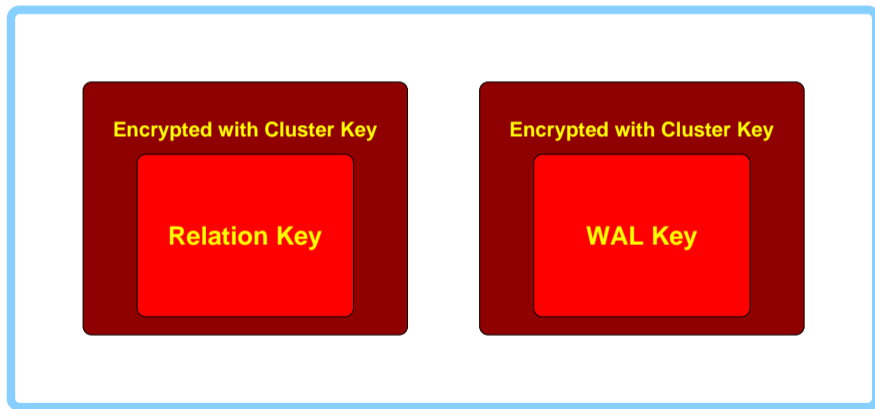
[https://github.com/postgres/postgres/compare/master..bmomjian:\\_cfe-01-doc.patch](https://github.com/postgres/postgres/compare/master..bmomjian:_cfe-01-doc.patch)

# Protections

- Prevents users with read access on the directories from being able to access the user data stored in those files
- Provides data-at-rest security, including physical backups
- Does not protect against unauthorized file system writes
- Does not protect against users who have read access to database process memory

# Key Storage

**\$PGDATA/pg\_cryptkeys**

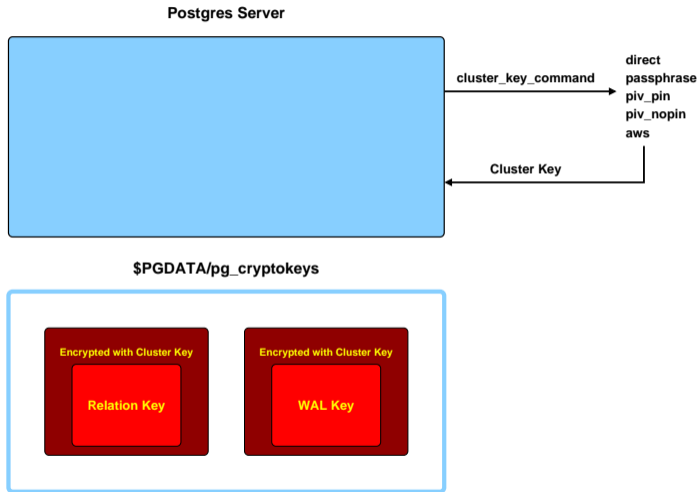


# Cluster Key Retrieval

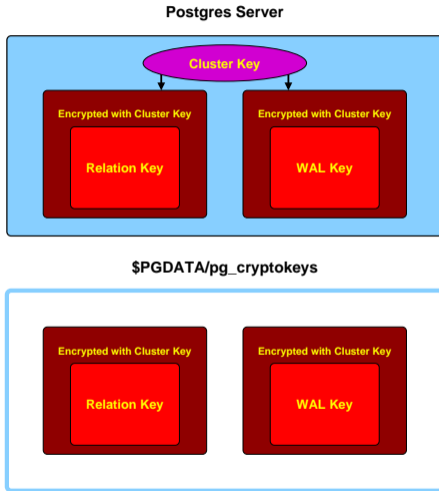
The cluster key, which unlocks the data keys stored in the file system, can be retrieved from:

- User's terminal
- Cryptographic hardware
- External key store

# Key Retrieval

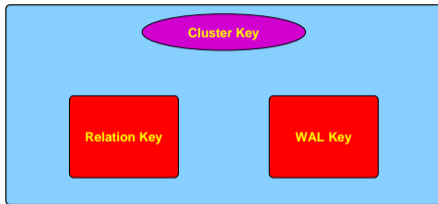


# Using the Cluster Key

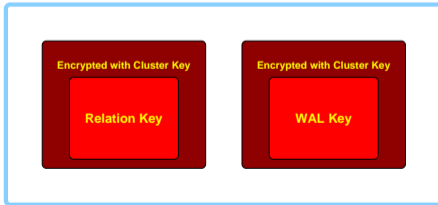


# Unlocking Keys

Postgres Server



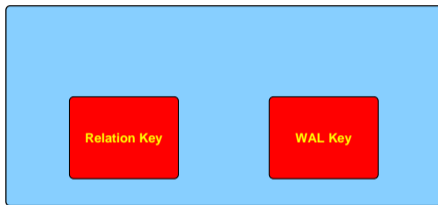
\$PGDATA/pg\_cryptkeys



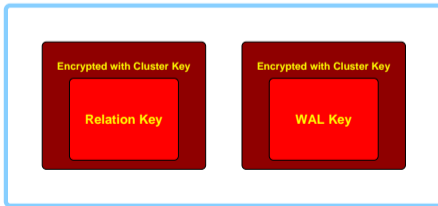


# Erasing the Cluster Key

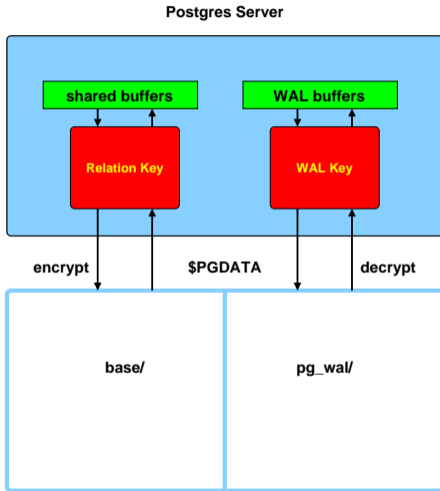
Postgres Server



\$PGDATA/pg\_cryptokeys



# Encryption/Decryption



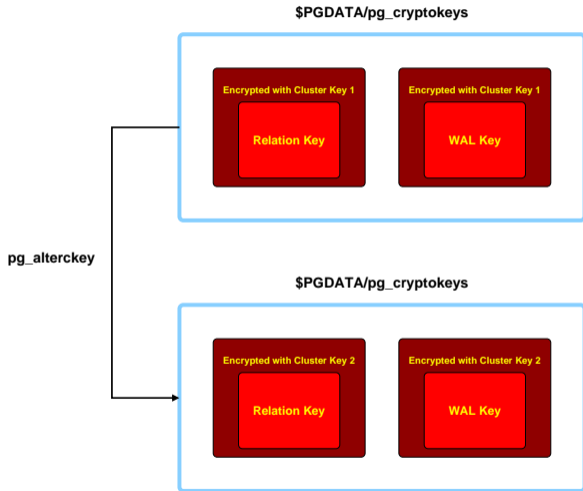
# Key Rotation

Cluster file encryption allows cluster key and data key rotation

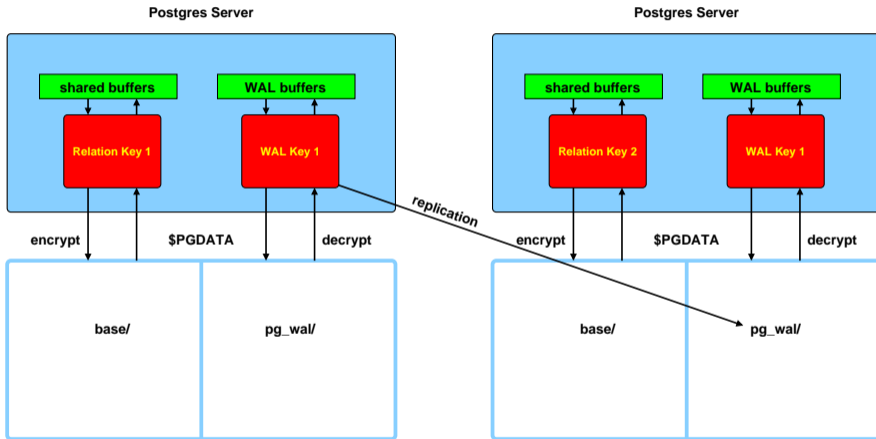
- Cluster key rotation is accomplished by running `pg_alterkey`, [https://github.com/postgres/postgres/compare/bmomjian:cfe-07-bin..bmomjian:\\_cfe-08-pg\\_alterkey.patch](https://github.com/postgres/postgres/compare/bmomjian:cfe-07-bin..bmomjian:_cfe-08-pg_alterkey.patch)
- Data key rotation is accomplished by creating a standby with a different data key and switching to it, then changing the WAL key

Encryption can be added to an existing cluster by creating and switching to a standby server that has encryption enabled.

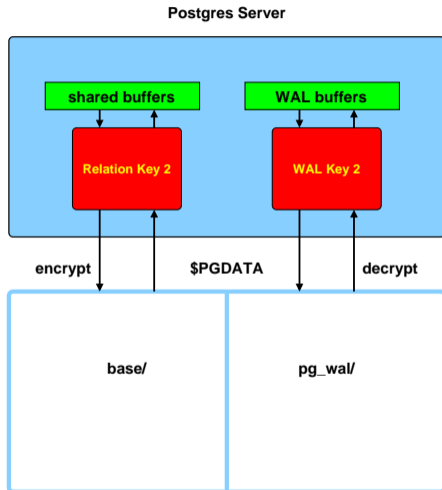
# Cluster Key Rotation



# Relation Key Rotation Using Replication



# WAL Key Rotation



# Current Status

- Dedicated feature page, [https://wiki.postgresql.org/wiki/Transparent\\_Data\\_Encryption](https://wiki.postgresql.org/wiki/Transparent_Data_Encryption)
- Several pending patches, [https://wiki.postgresql.org/wiki/Transparent\\_Data\\_Encryption#Patches](https://wiki.postgresql.org/wiki/Transparent_Data_Encryption#Patches)
- Testing possible
- Patch application is planned for Postgres 16
- Postgres 16 release planned for September/October 2023

# Conclusion



<https://momjian.us/presentations>

<https://www.flickr.com/photos/08dreizehn/>