# info.SetPassword – Technical Documentation

Version: 2.0  
Date: 2025-08-01  
Page: 1 of 2

## Summary

This function updates the password of an existing PostgreSQL user after validating the input. It prevents updates to superuser accounts and logs all actions into the audit schema for compliance.

## Function Logic

1. Checks if username is NULL, empty, or unsafe ('postgres')  
2. Ensures password is not NULL, empty, and >= 12 characters  
3. Confirms the role exists in `pg\_roles`  
4. Executes `ALTER ROLE ... WITH PASSWORD ...`  
5. Inserts an audit row into `info.password\_change\_audit` with timestamp, caller, and target username

## Audit Schema

- Table: `info.password\_change\_audit`  
- Fields: audit\_id (PK), changed\_at, changed\_by, changed\_user, note  
- Created if missing during initial install  
- Permissions split across reader/writer/admin roles

## Security Notes

- Prevents superuser update (hard-coded check for 'postgres')  
- Can only update existing roles  
- Requires EXECUTE on function and ALTER privilege on target role

## Limitations

- Does not support role creation  
- Does not enforce strong password complexity beyond length  
- No password history check or rotation policy included

## References

- https://www.postgresql.org/docs/current/sql-alterrole.html  
- https://www.postgresql.org/docs/current/catalog-pg-authid.html

## TL;DR

Safely updates a user's password. Logs who changed it, when, and for whom. Rejects risky behavior like touching 'postgres'.