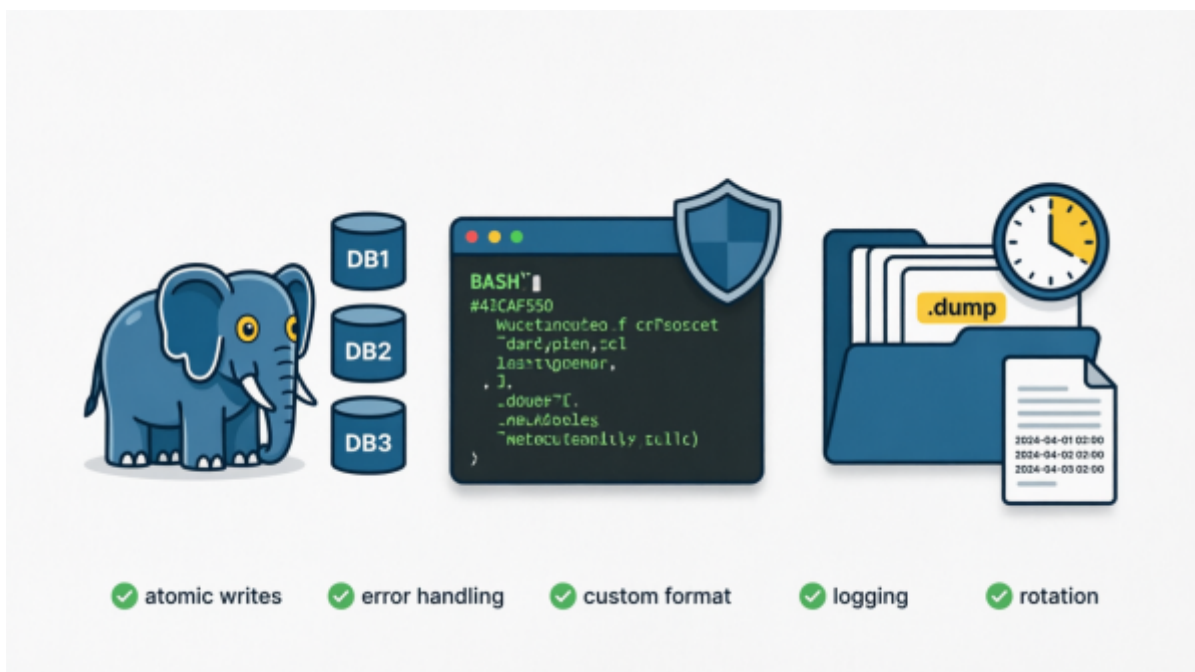


Скрипт архивирования всех баз PostgreSQL на Linux (Версия 2.0).

Экспорт в PDF 

Дата создания: 2026/07/01 08:43 (C) mihanik



Дано

- Операционная система Linux с установленным PostgreSQL.
- В СУБД создано несколько баз.
- Имеется текстовый файл со списком баз, которые нужно архивировать.

Надо

- Создать архивную копию баз из списка.
- Бэкап каждой базы должен храниться в отдельном архивном файле.
- Во время архивации старые архивные копии должны перезаписываться.

Нет создания директории	<code>\mkdir -p «\$BackupDir»`</code>	Скрипт не падает, если папка ещё не создана
Нет проверки списка БД	Проверка существования <code>\LIST_FILE`</code>	Ясная ошибка вместо молчаливого провала

□ Ключевые архитектурные изменения

1. Формат бэкапа

Было: `pg_dump | gzip` → `.sql.gz`

Стало: `pg_dump -Fc -Z 6` → `.dump`

Это даёт:

- Встроенное сжатие (не нужен `gzip`)
- Возможность параллельного восстановления (`pg_restore -j 4`)
- Возможность восстановить одну таблицу из дампа: `pg_restore -t users ...`
- Меньше нагрузка на CPU при дампе

2. Список баз

Было: Автоматический, все БД кроме шаблонов

Стало: Управляемый файл `/opt/scripts/pg_backup_list.lst` с комментариями

3. Аутентификация

- **Было:** Подключение через unix-сокет → ошибка `Peer authentication failed`
- **Стало:** `-h localhost` → TCP-подключение с проверкой пароля

Файл со списком баз

Создайте файл со списком баз для бэкапа: `/opt/scripts/pg_backup_list.lst`

[pg_backup_list.lst](#)

```
# Список баз для бэкапа
my_db1
my_db2
# analytics - временно отключена
```

Сам скрипт

[pg_backup.sh](#)


```
total=$((total + 1))
log "[${total}] Обработка базы: $dbname"

temp_file="$BackupDir/${dbname}.dump.tmp"
final_file="$BackupDir/${dbname}.dump"

# -Fc  – кастомный формат (сжатый, бинарный)
# -Z   – уровень сжатия (0-9)
if pg_dump -h localhost -U postgres -Fc -Z "$DUMP_COMPRESSION"
"$dbname" > "$temp_file"; then
    mv "$temp_file" "$final_file"
    log "Успешно создан архив: $final_file"
    success=$((success + 1))
else
    log "ОШИБКА при создании архива $dbname! Старый бэкап сохранен."
    rm -f "$temp_file"
    failed=$((failed + 1))
fi
done < "$LIST_FILE"

# === Ротация: удаляем бэкапы старше 7 дней ===
# Закомментирована

#find "$BackupDir" -name "*.dump" -type f -mtime +7 -delete
#log "Ротация завершена: удалены файлы старше 7 дней."

# === Итог ===

unset PGPASSWORD
log "=== Завершение: всего=${total} успешно=${success} ошибок=${failed} ==="

if [ "$failed" -gt 0 ]; then
    exit 1
fi
```

Источники

[Наверх](#)



В моей WIKI постоянно ведётся какая-то работа со статьями.
Если у вас возникли вопросы или замечания,
можете их отправлять на почту support@mihanik.net

