*Математические структуры и моделирование*
*2024. № 2 (70). С. 100–104*

УДК 004.056
DOI 10.24147/2222-8772.2024.2.100-104

# ПРОТОКОЛИРОВАНИЕ ДЕЙСТВИЙ ПОЛЬЗОВАТЕЛЕЙ В БАЗЕ ДАННЫХ

**Т.М. Опарина**
старший преподаватель, e-mail: oparinatm@omsu.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

**Аннотация.** Базы данных требуют надёжного и комплексного процесса для протоколирования действий пользователей. Для этого используются журналы сообщений сервера. Однако при протоколировании сообщений администраторы СУБД и баз данных сталкиваются со множеством проблем. Для решения этих проблем в данной статье рассматриваются возможности СУБД PostgreSQL, которая в последнее время становится всё популярнее.

**Ключевые слова:** протоколирование, база данных.

## Введение

Протоколирование в базе данных – это комплекс мероприятий, связанных с мониторингом действий в базе данных. При протоколировании сообщений сервера в базе данных можно столкнуться со следующими проблемами:

1. Недостаточная детализация. Если журнал сообщений не содержит достаточно подробных данных о действиях пользователей или изменениях в базе данных, это может затруднить обнаружение и реагирование на потенциальные угрозы безопасности.

2. Недостаточная защита журнала сообщений от несанкционированного доступа или модификации. Если данные в журнале не защищены должным образом, это может привести к изменению или удалению информации злоумышленниками.

3. При длительной работе с базой данных и увеличением объёма данных в базе данных растёт и размер журналов протоколирования, что, в свою очередь, приводит к проблемам с хранением, обработкой и анализом данных в журнале сообщений, особенно если не предусмотрены эффективные механизмы фильтрации и сжатия информации.

4. При использовании репликации баз данных или распределённых систем возникают сложности с согласованием данных в журналах сообщений. Это может привести к появлению расхождений между данными в системе.

5. В некоторых случаях процесс протоколирования сообщений сервера может быть недостаточно автоматизирован, что усложняет его масштабирование и поддержку. Необходимо разработать эффективные средства автоматизации для обеспечения надёжности и эффективности процесса протоколирования.

В целом видно что протоколирование сообщений сервера в базе данных требует детального анализа и разработки соответствующих правил для обеспечения безопасности и целостности работы системы.

## 1. Обзор требований к протоколированию сообщений сервера в базе данных

Федеральная служба по техническому и экспортному контролю (ФСТЭК) устанавливает ряд требований к процессу протоколирования сообщений в базе данных, чтобы обеспечить защиту данных. Одним из основных требований ФСТЭК к протоколированию сообщений сервера в базе данных является обязательность ведения журнала событий (лог файлов). Этот журнал должен содержать информацию о всех действиях связанных с:

- созданием учётных записей пользователей системы управления базами данных;
- изменением атрибутов учётных записей пользователей системы управления базами данных;
- успешными и неуспешными попытками аутентификации пользователей системы управления базами данных;
- запуском и остановкой системы управления базами данных с указанием причины остановки;
- изменением конфигурации системы управления базами данных; созданием и удалением базы данных, таблицы, за исключением временных таблиц, создаваемых системой управления базами данных в служебных целях;
- подключением, восстановлением базы данных;
- изменением правил разграничения доступа в системе управления базами данных;
- фактами нарушения целостности объектов контроля;
- созданием и изменением процедур (программного кода), хранимых в базах данных и представлений [1].

Другим важным требованием является обеспечение конфиденциальности и целостности протоколируемых данных. Что, в свою очередь, означает, что некоторые

данные в лог-файле должны быть зашифрованы и защищены от несанкционированного доступа, а также должны быть подвержены проверке на целостность для исключения возможности модификации информации. Кроме того, требуется обеспечить возможность протоколирования доступа к лог-файлу для контроля.

## 2. Реализация основных методов протоколирования в PostgreSQL

Одним из важных аспектов работы с PostgreSQL является протоколирование, т. е. запись различных событий и действий, происходящих в базе данных. В базе данных PostgreSQL расширение pg_audit позволяет регистрировать различные события, связанные с безопасностью. Данное расширение работает параллельно со стандартными средствами протоколирования PostgreSQL. Рассмотрим требования к протоколированию, описанные выше, в соответствии с возможностями данной СУБД [2, 3] (см. табл. 1).

Таблица 1. Необходимые параметры протоколирования PostgreSQL для выполнения требований по отслеживанию действий пользователей

| Требования к протоколированию | Основные параметры протоколирования |
|---|---|
| Создание учётных записей пользователей, изменение атрибутов учётных записей | debug_print_parse, debug_print_rewritten, debug_print_plan, log_statement, log_duration |
| Попытки аутентификации пользователей и выход из базы данных | log_connections, log_disconnections |
| Запуск и остановка СУБД | log_checkpoints |
| Изменение конфигурации СУБД | log_checkpoints |
| Правила разграничения доступа | log_statement, log_duration |
| Нарушения целостности объектов | log_statement |
| Создание и изменение процедур | log_statement, log_duration |

Представленные в таблице параметры:

- log_statement,log_duration включают информацию о выполненных SQL командах;
- log_checkpoints включает протоколирование выполнения контрольных точек и точек перезапуска СУБД;
- log_connections – включение протоколирования всех попыток подключения к серверу;

- log_disconnections – включение протоколирования всех попыток отключения от сервера;

- debug_print_parse, debug_print_rewritten, debug_print_plan – параметры включают вывод дерева запроса, плана выполнения запроса.

А также параметры, которые являются важными для обеспечения безопасности и целостности данных: log_destination (для включения журнала протоколирования), log_error_verbosity (управляет количеством детальной информации, записываемой в журнал сервера).

Эти параметры задаются в конфигурационном файле postgresql.conf или в командной строке при запуске сервера (ALTER SYSTEM SET).

Результат работы протоколирования в СУБД PostgreSQL представлен на рис. 1.

```
2024-03-04 00:54:03.804 MSK [9160] ВАЖНО:  пользователь "postgres" не прошёл проверку подлинности (по паролю)
2024-03-04 00:54:03.804 MSK [9160] ПОДРОБНОСТИ:  Подключение соответствует строке 117 в "C:/Program Files/PostgreSQL/16/data/pg_hba.conf":
"host    all             all             ::1/128                 scram-sha-256"
2024-03-04 00:55:30.821 MSK [5968] СООБЩЕНИЕ:  начата контрольная точка: time
2024-03-04 00:55:40.951 MSK [5968] СООБЩЕНИЕ:  контрольная точка завершена: записано буферов: 94 (0.6%); добавлено файлов WAL 0, удалено: 0,
переработано: 0; запись=10.108 сек., синхр.=0.019 сек., всего=10.131 сек.; синхронизировано_файлов=51, самая_долгая_синхр.=0.001 сек.,
средняя=0.001 сек.; расстояние=393 kB, ожидалось=393 kB; lsn=0/16373B8, lsn redo=0/1637380

2024-03-04 01:21:22.064 MSK [4708] ОШИБКА:  выполнение оператора отменено по запросу пользователя
2024-03-04 01:21:22.103 MSK [5968] СООБЩЕНИЕ:  выключение
2024-03-04 01:21:22.114 MSK [5968] СООБЩЕНИЕ:  начата контрольная точка: shutdown immediate
2024-03-04 01:21:22.135 MSK [5968] СООБЩЕНИЕ:  контрольная точка завершена: записано буферов: 9 (0.1%); добавлено файлов WAL 0, удалено: 0,
переработано: 0; запись=0.001 сек., синхр.=0.003 сек., всего=0.033 сек.; синхронизировано_файлов=7, самая_долгая_синхр.=0.001 сек.,
средняя=0.001 сек.; расстояние=0 kB, ожидалось=354 kB; lsn=0/1637468, lsn redo=0/1637468
```

Рис. 1. Журнал протоколирования

## Заключение

Таким образом, журналы протоколирования в PostgreSQL позволяют отслеживать изменения, обнаруживать ошибки, предотвращать угрозы безопасности и восстанавливать данные в случае сбоев, что является критически важным для защиты конфиденциальной информации. Важно понимать, что требования к безопасности постоянно меняются и развиваются, поэтому необходимо постоянно совершенствовать процессы протоколирования. Перспективы развития систем протоколирования включают в себя использование новых технологий для автоматизации процессов протоколирования, оптимизацию параметров записи данных, улучшение анализа и мониторинга журналов. Дальнейшие исследования в этой области могут быть направлены на анализ эффективности новых методов протоколирования, разработку инновационных инструментов для анализа журналов, изучение влияния протоколирования на производительность системы и разработку рекомендаций по оптимизации процесса протоколирования. Эффективное протоколирование является ключевым компонентом обеспечения безопасности данных.

## Литература

1. Требования по безопасности информации. Утверждены приказом ФСТЭК России от 14 апреля 2023 г. № 64. URL: https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/trebovaniya-po-bezopasnosti-informatsii-utverzhdeny-prikazom-fstek-rossii-ot-14-aprelya-2023-g-n-64 (дата обращения: 24.04.2024).
2. Документация PostgresSQL URL. https://postgrespro.ru/docs/postgresql/14/runtime-config-logging\#RUNTIME-CONFIG-LOGGING-WHAT (дата обращения: 24.04.2024)
3. Лесовский А.В. Мониторинг PostgreSQL. М.: Бумба, 2024. 247 с.

**LOGGING OF USER ACTIONS IN THEE DATABASE**

**T.M. Oparina**
Assistant Professor, e-mail: oparinatm@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

**Abstract.** Databases require a reliable and comprehensive process for logging user actions. Server message logs are used for this purpose. However, DBMS and database administrators face many problems when logging messages. To solve these problems, this article discusses the capabilities of the PostgreSQL database management system, which has recently become more popular.

**Keywords:** logging, database.

*Дата поступления в редакцию: 28.04.2024*