СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ В ЗАЩИЩЕННОМ ИСПОЛНЕНИИ НА БАЗЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С ОТКРЫТЫМ ИСХОДНЫМ КОДОМ «СИНЕРГИЯ-БД»

ОПИСАНИЕ ПРИМЕНЕНИЯ

Листов 23

Содержание

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

1.1. Назначение

СУБД «СИНЕРГИЯ-БД» является программным средством общего назначения, предназначенным для создания и управления реляционными базами данных и обеспечивающим многопользовательский доступ к расположенным в них данным с разным уровнем конфиденциальности.

1.2. Основные характеристики

СУБД «СИНЕРГИЯ-БД» состоит из следующих основных компонентов:

- ядро СУБД,
- комплекс средств защиты (КСЗ),
- средства тестирования.

В состав КСЗ входят следующие основные подсистемы:

- средства контроля целостности;
- подсистема аутентификации;
- подсистема управления доступом;
- подсистема управления внешней и оперативной памятью;
- подсистема регистрации событий.

1.3. Возможности

СУБД предоставляет следующие возможности:

- управление данными во внешней памяти;
- управление данными в оперативной памяти;
- выполнение запросов;
- управление транзакциями;
- журнализация изменений, резервное копирование и восстановление базы данных после сбоев, репликация;
- поддержка языков определения и манипулирования данными.

КСЗ обеспечивает реализацию следующих функций по защите информации от НСД:

- дискреционного разграничения доступа;
- мандатное разграничения доступа;
- идентификации и аутентификации;
- очистки памяти;

- регистрации событий;
- сопоставление пользователя с устройством;
- изоляцию модулей;
- надежное восстановление;
- контроля целостности КСЗ.

2. УСЛОВИЯ ПРИМЕНЕНИЯ

2.1. Требования к техническим средствам

Для функционирования СУБД «СИНЕРГИЯ-БД» необходима следующая минимальная конфигурация компьютера:

- аппаратная платформа: процессор с тактовой частотой 2 ГГц,
- объем ОЗУ: 2 Гбайта,
- объем свободного дискового пространства: 40 Гбайт,
- устройство чтения оптических дисков.

3. ОПИСАНИЕ ЗАДАЧИ

Основная задача, решаемая СУБД в процессе своего функционирования — управление реляционными базами данных в соответствии с «Требованиями о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» (приказ ФСТЭК России No 17 от 11.02.2013), «Составом и содержанием организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных» (приказ ФСТЭК России No 21 от 18.02.2013) и методическим документом «Меры защиты информации в государственных информационных системах».

3.1. Классы решаемых задач

Для решения основной задачи функционирования СУБД она разбивается на следующие классы задач:

- дискреционное разграничение доступа;
- мандатное разграничение доступа;
- идентификация и аутентификация;
- очистка памяти;
- регистрация событий;
- контроль целостности КСЗ.

3.1.1. Дискреционное разграничение доступа

Роли

СУБД «СИНЕРГИЯ-БД» использует концепцию ролей для управления разрешениями на доступ к базе данных. Роль можно рассматривать как пользователя базы данных или как группу пользователей, в зависимости от того как роль настроена. Роли могут владеть объектами базы данных (например, таблицами) и выдавать другим ролям разрешения на доступ к этим объектам, управляя тем, кто имеет доступ и к каким объектам. Кроме того, можно предоставить одной роли членство в другой роли, таким образом одна роль может использовать привилегии других ролей.

При начальной инициализации кластера базы данных система всегда содержит одну предопределенную роль. Эта роль является суперпользователем и по умолчанию имеет такое же имя, как и пользователь операционной системы, инициализирующий кластер баз данных. Обычно эта роль называется postgres. Для создания других ролей вначале нужно подключиться с этой ролью.

Каждое подключение к серверу базы данных выполняется под именем конкретной роли и эта роль определяет начальные привилегии доступа для команд выполняемых в этом соединении. Список доступных для подключения ролей, который могут использовать клиенты, определяется настройками аутентификации.

Роль базы данных может иметь атрибуты, определяющие ее полномочия и взаимодействие с системой аутентификации клиентов: право подключения к базе данных, право обходить проверки прав доступа (суперпользователь), право на создание баз данных, право на создание других ролей, право на запуск потоковой репликации.

Групповые роли

Пользователи могут быть сгруппированы для упрощения администрирования: привилегии выдаются или отзываются на всю группу. В СУБД «СИНЕРГИЯ-БД» для этого создается роль, которая представляет группу, а затем членство в этой группе выдается ролям индивидуальных пользователей.

Можно выдавать членство в групповой роли другим групповым ролям.

При удалении групповой роли любое членство в этой роли будет автоматически отозвано (в остальном на членов групповой роли это никак не повлияет). Однако любые объекты, владельцем которых является групповая роль, предварительно должны быть удалены или переданы другим владельцам. Также любые права, выданные групповой роли, должны быть отозваны.

Привилегии

Когда в базе данных создается объект, ему назначается владелец. Владельцем обычно становится роль, с которой был выполнен оператор создания. Для большинства типов объектов в исходном состоянии только владелец (или суперпользователь) имеет полный доступ к объекту, в том числе неотъемлемое право изменять, удалять объект, а также выдавать или отзывать привилегии.

Объекту можно назначить нового владельца. Суперпользователь может делать это без ограничений, а обычный пользователь, только если он является одновременно текущим владельцем объекта (или членом роли владельца) и членом новой роли.

Чтобы разрешить использовать объект другим ролям, нужно дать им соответствующие привилегии. Можно дать привилегию доступа к объекту «с правом передачи», что позволит получившему такую привилегию назначать ее другим. Если такое право передачи впоследствии будет отозвано, то все, кто получил данное право доступа (непосредственно или по цепочке передачи), потеряют его.

Единицей разграничения доступа к данным обычно является таблица, но можно разграничить доступ и на уровне столбцов. Пользователь может выполнять операции над столбцом, если у него есть либо привилегия на этот конкретный столбец, либо на всю таблицу.

Существует несколько типов привилегий: SELECT, INSERT, UPDATE, DELETE, TRUNCATE, REFERENCES, TRIGGER, CREATE, CONNECT, TEMPORARY, EXECUTE и USAGE. Набор прав, применимых к определенному объекту, зависит от типа объекта.

Политики защиты строк

В дополнение к описанной системе безопасности, для таблиц можно определить политики защиты строк, ограничивающие набор строк, которые могут быть возвращены при обычных

запросах или добавлены командами, изменяющими данные. По умолчанию, таблицы не имеют политик, в них видны и могут быть добавлены любые строки, так что доступ к ним регулируется обычным образом. Подход с ограничивающими политиками также называется защитой на уровне строк (RLS, Row Level Security).

Когда для таблицы включается защита строк, все обычные запросы к таблице (за исключением запросов ее владельца) на выборку или добавление строк обрабатываются с учетом политик. Если политики для таблицы не определены, применяется политика запрета по умолчанию, так что никакие строки в этой таблице нельзя получить или добавить. Права, действующие на уровне всей таблицы, в частности TRUNCATE и REFERENCES, защита строк не ограничивает.

Чтобы определить, какие строки в таблице с защитой на уровне строк будут видны, и какие могут быть добавлены, задается выражение, возвращающее логический результат. Это выражение будет вычисляться для каждой строки перед другими условиями или функциями, задействованными в запросе. Одним исключением из этого правила являются герметичные функции, которые гарантированно не допускают утечки информации. Чтобы независимо управлять набором строк, которые можно получить, и набором строк, которые можно добавить, для политики можно задать два выражения. Заданное выражение обрабатывается в составе запроса с правами исполняющего его пользователя, но в этом выражении могут применяться функции, определяющие контекст безопасности.

Неотъемлемое право включать или отключать защиту строк, а также определять политики для таблицы, имеет только владелец объекта.

Когда к определенному запросу применяются несколько политик, они объединяются логическим сложением, подобному тому, как некоторая роль получает права всех ролей, в которые она включена.

На проверки ссылочной целостности, например, на ограничения уникальности, первичные и внешние ключи, защита строк не распространяется, чтобы не нарушалась целостность данных. Поэтому схемы и политики защиты на уровне строк необходимо тщательно прорабатывать, чтобы перекрыть каналы неявной утечки информации через эти проверки.

Средства управления дискреционными ПРД

Для управления дискреционными ПРД используются несколько команд SQL.

Команда GRANT определяет права доступа. Команда GRANT имеет две основные разновидности: первая назначает права для доступа к объектам баз данных (таблицам, колонкам, представлениям, сторонним таблицам, последовательностям, базам данных, оберткам сторонних данных, сторонним серверам, функциям, процедурным языкам, схемам или табличным пространствам), а вторая назначает одни роли членами других.

GRANT для объектов баз данных дает одной или нескольким ролям определенные права для доступа к объекту базы данных. Эти права добавляются к списку имеющихся, если роль уже наделена какими-то правами. Также можно дать роли некоторое право для всех объектов одного типа в одной или нескольких схемах.

GRANT для ролей включает роль в члены одной или нескольких других ролей. Членство в ролях играет важную роль, так как права, данные роли, распространяются и на всех ее членов.

Команда REVOKE отзывает права доступа, то есть лишает одну или несколько ролей прав, назначенных ранее.

Команда CREATE ROLE создает роль в базе данных.

Команда ALTER ROLE изменяет роль в базе данных.

Команда DROP ROLE удаляет роль в базе данных.

Команда SET ROLE устанавливает идентификатор текущего пользователя в рамках сеанса. После выполнения этой команды права доступа для команд SQL проверяются так, как если бы сеанс изначально был установлен с указанным именем роли.

Команда ALTER DEFAULT PRIVILEGES определяет права доступа по умолчанию. Она позволяет задавать права, применяемые к объектам, которые будут создаваться в будущем.

Команда SET SESSION AUTHORIZATION устанавливает идентификатор пользователя сеанса и идентификатор текущего пользователя в рамках сеанса. С помощью этой команды, можно, например, временно переключиться на непривилегированного пользователя, сохранив возможность затем стать суперпользователем.

Команда DROP OWNED удаляет все объекты базы данных, принадлежащие роли. Кроме того, роль лишается всех прав, которые она имела для объектов текущей базы данных и общих объектов.

Команда REASSIGN OWNED изменяет владельца объектов базы данных, принадлежащих заданной роли.

Команда CREATE POLICY создает политику для таблицы. Политика дает разрешение на выборку, добавление, изменение или удаление строк, удовлетворяющих соответствующему выражению политики.

Команда ALTER POLICY изменяет определение политики.

Команда DROP POLICY удаляет политику из таблицы.

Команда ALTER TABLE изменяет определение таблицы. В частности, эта команда управляют применением политик защиты строк, принадлежащих таблице, в случаях, когда пользователь является или не является владельцем таблицы.

3.1.2. Мандатное разграничение доступа

Основой мандатного механизма разграничения доступа является управление доступом к защищаемым ресурсам БД на основе иерархических и неиерархических меток доступа. Это позволяет реализовать многоуровневую защиту с обеспечением разграниченияdоступа пользователей к защищаемым ресурсам БД и управление потоками информации.

В качестве иерархических и неиерархических меток доступа при использовании СУБД используются метки конфиденциальности или метки безопасности операционной системы.

СУБД «СИНЕРГИЯ-БД» не имеет собственного механизма назначения, хранения и модификации меток пользователей и использует для этого механизмы ОС.

В СУБД «СИНЕРГИЯ-БД» минимальной структурой, защищяемой мандатными метками конфиденциальности являются строки таблиц. Метки конфиденциальности строк хранятся в самих таблицах в скрытом системном столбце maclabel. Помимо строк таблиц, метки конфиденциальности могут быть установлены для баз данных, табличных пространств, схем, таблиц, представлений, материализованных представлений, последовательностей, функций, процедурных языков.

В части реализации мандатного разграничения доступа в дополнение к мандатной метке конфидициальности вводится понятие объектов-контейнеров. Под объектами-контейнерами понимаются такие объекты, которые могут содержать другие объекты. Для определения способа доступа к объектам внутри контейнера используется мандатный признак CCR (Container Clearance Required). Если признак установлен для объекта-контейнера, то доступ к контейнеру и его содержимому определяется мандатной меткой конфиденциальности контейнера. Если признак CCR для объекта-контейнера не установлен, то доступ к объектам внутри контейнера осуществялется без учета метки конфиденциальности самого контейнера.

Таблица. Объекты-контейнеры и содержимое контейнеров

Контейнер	Содержимое
База данных	Схема, процедурный язык
Табличное пространство	Таблица, материализованное представление
Схема	Таблица, представление, материализованное представление, функция, последовательность
Таблица	Строки таблицы, наследующие от неё таблицы
Процедурный язык	Функция

Используемая модель доступа накладывает ограничение на возможные значения мандатной метки конфиденциальности объекта: метка объекта не может превышать метку контейнера, в котором он содержится. Таким образом при назначения меток, сначала должны быть последовательно заданы максимальные метки соответствующих контейнеров.

Субъекты доступа

Субъектом доступа в СУБД «СИНЕРГИЯ-БД» явдяется сеанс пользователя для работы с базой данных.

В операционной системе каждый пользователь может иметь множество меток конфиденциальности. Диапазон возможных значений меток определяется заданными для пользователя минимальной и максимальной метками. В момент подключения к СУБД у пользователя установлена текущая метка конфиденциальности из диапазона возможных значений.

Текущая метка конфиденциальности ОС будет использоваться для каждого подключения пользователя к базе данных. Для получения текущей метки сеанса пользователя следует использовать функцию getusermaclabel().

Если пользователь был аутентифицирован и в ОС, и в СУБД, то для сеанса работы в БД используется метка конфиденциальности полученная из ОС. Это внутренние пользователи.

Если пользователь был аутентифицирован в ОС, но не был аутентифицирован в СУБД, то он подключается к базе данных под именем nobody и его сеанс получает нулевую метку конфиденциальности. Это внешние пользователи.

Для администратора базы данных и администратора информационной безопасности существуют специальные пользователи в СУБД, которые после подключения игнорируют мандатные ПРД для выполнения своих задач. В качестве администратора базы данных СУБД «СИНЕРГИЯ-БД» использует пользователя (роль базы данных) с именем postgres. В качестве администратора информационной безопасности используется пользователь с именем dbsa. Эти два специальных пользователя, а также пользователь nobody, автоматически создаются при инициализации кластера базы данных командой initdb.

Пользователи не прошедшие аутентификацию и в ОС, и в СУБД, а также не являющиеся администраторами – в базу данных не допускаются.

Таким образом, для субъектов доступа (сеансов работы с базы данных) СУБД не хранит метки конфиденциальности. СУБД использует метку конфиденциальности полученную из ОС для реализации мандатных ПРД.

Мандатный доступ к табличным данным

Так как мандатный контроль доступа может быть определен только для видов доступа на чтение и на запись информации, все множество операций с данными в защищаемых объектах приводится к ним следующим образом:

– INSERT, COPY … FROM …, TRUNCATE — доступ на запись;

– UPDATE, DELETE — последовательное выполнение доступа на чтение и запись информации;

– SELECT, COPY … TO ... — доступ на чтение.

Применение мандатных ПРД осуществляется на уровне доступа к объектам БД и на уровне доступа непосредственно к данным (на уровне строк таблиц).

Проверка мандатных прав доступа к объектам осуществляется одновременно с проверкой дискреционных прав доступа к ним. Таким образом, доступ предоставляется только при одновременном санкционировании дискреционными и мандатными ПРД.

Все записи, помещаемые в таблицы, для которых установлена защита на уровне строк, наследуют текущую метку пользователя. Обновляемые записи меняют свою метку на текущую метку конфиденциальности сеанса пользователя. Доступ к существующим записям и возможность их обновления и удаления определяются установленными мандатными правилами.

Для администратора базы данных и администратора информационной безопасности игнорируются правила мандатного разграничения доступа. Это позволяет производить регламентные работы с БД (например, создание резервных копий и последующее восстановление из них).

Мандатный доступ к объектам БД

Помимо изменения собственно данных (в таблицах), при работе с СУБД возможны операции изменения схемы и размещения объектов в базе данных: перенос объектов между схемами, табличными пространствами, модификация структуры объектов, удаление объектов и т.д. К таким операциям с метаданными также применяются мандатные ПРД.

Доступ к метаданным, которые в СУБД «СИНЕРГИЯ-БД» называются объектами системного каталога, реализуется при помощи набора команд SQL. Таким образом можно считать, что мандатное разграничения доступа применяется ко всем объектам БД, включая объекты системного каталога. Как и для обычных таблиц метки системных объектов располагаются в скрытом системном столбце maclabel в соответствующих таблицах системного каталога.

Для реализации мандатных ПРД, все операции изменения объектов БД (по аналогии с операциями доступа к данным) должны быть сведены к видам доступа чтение и запись. Считается что операторам:

– CREATE, ADD — требуется доступ на запись;

– ALTER, DROP — требуется последовательное выполнение доступа на чтение и запись информации;

– использование или обращение к объекту в других SQL-командах — требуется доступ на чтение.

Проверка мандатных прав доступа к метаданным осуществляется одновременно с проверкой дискреционных прав доступа к ним. Таким образом, доступ предоставляется только приодновременном санкционировании дискреционными и мандатными ПРД.

Средства управления мандатными ПРД

Для управления мандатными ПРД к объектам СУБД «СИНЕРГИЯ-БД» можно использовать:

– psql — консольная утилита (текстовый терминал) для администрирования и работы с СУБД;

– pgAmin 3 — графическая утилита администрирования.

При создании защищаемых объектов БД, мандатная метка создаваемого объекта устанавливается равной текущей мандатной метке сеанса пользователя. Если создаваемый объект является контейнером для других объектов, то для него устанавливается признак CCR.

Пользователи могут изменять мандатную метку объекта в сторону повышения. Изменять мандатную метку объекта в сторону понижения могут только администратор БД и администратор информационной безопасности.

3.1.3. Идентификация и аутентификация

При подключении к серверу базы данных, клиентское приложение указывает имя пользователя СУБД «СИНЕРГИЯ-БД», так же как и при обычном входе пользователя на

компьютер с ОС Unix. При работе в среде SQL по имени пользователя определяется, какие у него есть права доступа к объектам базы данных.

СУБД «СИНЕРГИЯ-БД» предлагает несколько различных методов аутентификации клиентов, включая GSSAPI, PAM. Метод аутентификации конкретного клиентского соединения может основываться на адресе компьютера клиента, имени базы данных, имени пользователя.

При аутентификации запроса на подключение имена пользователей базы данных СУБД «СИНЕРГИЯ-БД» сопоставляются с именами пользователей операционной системы. В зависимости от результата этого сопоставления спользователи СУБД получают права:

— Аутентифицированные и ОС и СУБД — права в соответствии с меткой, переданной ОС. Это внутренние пользователи СУБД.

— Аутентифицированные только ОС — получают минимальные права, соответствующие минимальной метке. Это внешние пользователи СУБД.

— Аутентифицированные только СУБД — это администраторы информационной безопасности или администраторы базы данных.

Пользователи не прошедшие аутентификацию в СУБД не допускаются.

Файл pg_hba.conf

Аутентификация клиентов управляется конфигурационным файлом, который называется pg_hba.conf и обычно расположен в каталоге с данными кластера БД. (HBA расшифровывается как host-based authentication — аутентификации по имени узла.) Файл pg_hba.conf с настройками по умолчанию создается командой initdb при инициализации кластера.

Обычный формат файла pg_hba.conf представляет собой набор записей, по одной на строке. Пустые строки игнорируются, как и любой текст комментария после знака #. Записи не продолжаются на следующей строке. Записи состоят из некоторого количества полей, разделенных между собой пробелами или символами табуляции. В полях могут быть использованы пробелы, если они взяты в кавычки. Если в кавычки берется какое-либо зарезервированное слово в поле базы данных, пользователя или адресации (например, all или replication), то слово теряет свое особое значение и просто обозначает базу данных, пользователя или сервер с данным именем.

Каждая запись обозначает тип соединения, диапазон клиентского IP адреса (если он соотносится с типом соединения), имя базы данных, имя пользователя, и способ аутентификации, который будет использован для соединения в соответствии с этими параметрами. Первая запись с соответствующим типом соединения, адресом клиента, указанной базой данных и именем пользователя применяется для аутентификации. Если выбрана запись и аутентификация не прошла, последующие записи не рассматриваются. Если же ни одна из записей не подошла, в доступе будет отказано.

Поскольку записи файла pg_hba.conf рассматриваются последовательно для каждого подключения, порядок записей имеет значение.

Файл pg_hba.conf прочитывается во время запуска и в момент получения основным

процессом сервера сигнала SIGHUP. Если файл изменяется во время работы системы, необходимо послать сигнал процессу postmaster (используя pg_ctl reload или kill -HUP), чтобы он прочел обновленный файл.

Файл сопоставления имен пользователей

Когда используется внешняя система аутентификации, имя пользователя операционной системы, инициировавшего подключение, может не совпадать с именем пользователя базы данных, под которым он хочет подключиться. В этом случае может быть составлен файл сопоставления имен пользователя, чтобы соотнести имя пользователя операционной системы и пользователя базы данных. Чтобы использовать функцию сопоставления имен пользователя, надо указать параметр map=*map-name* в поле auth-options файла pg_hba.conf. Эта опция поддерживается для всех методов аутентификации, получающих внешние имена пользователей. Для различных подключений могут использоваться разные сопоставления. Чтобы указать, какое сопоставление использовать для каждого конкретного подключения, имя нужного сопоставления должно быть указано в параметре *map-name*.

Сопоставления имен пользователя определяются в файле сопоставления, который по умолчанию называется pg_ident.conf и хранится в каталоге данных кластера (другое расположение файла сопоставления имен может быть указано в параметра сервера ident_file).

Управление паролями

Одним из встроенных способов аутентификации является введения клиентом пароля. База данных паролей СУБД отделена от паролей пользователей операционной системы. Пароль для каждого пользователя базы данных хранится в системном каталоге pg_authid. Управлять паролями можно с помощью команд SQL CREATE ROLE и ALTER ROLE. Если для пользователя не было установлено пароля, аутентификация данным методом для такого пользователя всегда будет завершаться неудачно.

Для дополнительных возможностей управления паролями в СУБД «СИНЕРГИЯ-БД» используется модуль passwordcheck. Модуль passwordcheck проверяет пароли пользователей, задаваемые командами CREATE ROLE и ALTER ROLE. Если пароль признаётся слишком слабым, он не принимается и команда завершается ошибкой.

Чтобы задействовать этот модуль, нужно добавить строку '$libdir/passwordcheck' в переменную shared_preload_libraries в postgresql.conf, а затем перезапустите сервер.

Для управления сложностью паролей модуль passwordcheck позволяет использовать следующие параметры конфигурации в файле postgresql.conf:

password_min_unique_chars

Минимальное количество неповторяющихся символов в пароле.
По умолчанию: 8

password_min_pass_len

Минимальное количество символов в пароле.
По умолчанию: 8

password_with_nonletters

Требовать наличие в пароле символов, отличных от букв.
По умолчанию: on

Все утилиты, поставляемые с СУБД «СИНЕРГИЯ-БД», при вводе пароля исключают отображение вводимых символов.

Управление учетными записями

Для управления блокированием учетных записей пользователей в СУБД «СИНЕРГИЯ-БД» используются следующие параметры конфигурации в файле postgresql.conf:

special_blocking_policy

Включает или отключает подсистему блокирования учетных записей.
По умолчанию: off

special_blocking_interval

Задает интервал времени, на котором измеряется превышение количества неуспешных попыток подключения.
По умолчанию: 10min

special_blocking_counts_wrong

Задает максимальное количество неуспешных попыток подключения за интервал времени special_blocking_interval.
По умолчанию: 5

special_blocking_idle_interval

Задает максимальный интервал времени с последнего успешного подключения. При достижении этого интервала, учетная запись пользователя автоматически блокируется.
По умолчанию: 30d

Для управления количеством одновременных сеансов под одной учетной записью пользователя используется опция CONNECTION LIMIT команд CREATE ROLE, ALTER ROLE. Например, чтобы установить не более 3-х одновременных подключений пользователю, нужно выполнить команду:

ALTER ROLE имя_пользователя CONNECTION LIMIT 3;

3.1.4. Очистка памяти

Очистка файлов во внешней памяти

При выполнении ряда операторов происходит удаление файлов, при котором дисковое пространство возвращается операционной системе. Это, в первую очередь, команды удаления:

— DROP TABLE

— DROP TEMPORARY TABLE

— DROP MATERIALIZED VIEW

— DROP INDEX

— TRUNCATE

— DROP DATABASE

— DROP SCHEMA

Но также и команды, вызывающие пересоздание объектов:

— VACUUM FULL

— REINDEX

— ALTER TABLE ADD COLUMN (с указанием значения по умолчанию)

— ALTER TABLE ALTER COLUMN TYPE

При установленном конфигурационном параметре wipe_file_on_delete удаляемые файлы предварительно заполняются нулевыми байтами.

Обратите внимание, что команда ALTER TABLE DROP COLUMN не приводит к пересозданию файла. Вся информация, содержавшаяся в удаленном столбце, остается внутри страниц (хотя к ней и невозможно обратиться средствами SQL). Если необходимо физически очистить файл от этой информации, используйте команду VACUUM FULL после удаления столбца.

Очистка страниц

В соответствии с принципом многоверсионности, при удалении строк (операцией DELETE) информация в странице не удаляется, но помечается как удаленная. Обновление (операцией UPDATE) строк работает как удаление строки и вставка новой; поэтому предыдущее значение также не удаляется из страницы.

Если на строки таблицы есть ссылки из индексов, то и в индексных страницах сохраняются ссылки на удаленные, но еще не освобожденные версии строк.

Эти данные необходимы для корректной работы механизма многоверсионности и не могут быть удалены из страницы до тех пор, пока сохраненная версия строки доступна хотя бы из одного активного снимка данных. Когда версия строки не доступна ни из одного снимка, эта версия может быть удалена процессом очистки (VACUUM). При этом одновременно удаляются и ссылки на версию строки из индексов.

При этом удаление не означает физическую очистку: при обычной работе соответствующее место на странице помечается как свободное и впоследствии может быть использовано для размещения какой-либо другой строки.

При установленном конфигурационном параметре wipe_heaptuple_on_delete процесс очистки не только помечает область страницы как свободную, но и заполняет ее нулевыми байтами.

Очистка оперативной памяти

При работе сервера постоянно происходит выделение и освобождение областей оперативной памяти. СУБД использует собственную систему распределения памяти, основанную

на контекстах. Память выделяется в одном из вложенных друг в друга контекстов, а при уничтожении контекста освобождается вся память, выделенная как в этом контексте, так и во всех вложенных. Такой поход позволяет существенно облегчить задачу устранения утечек памяти. Тем не менее, в конечном итоге выделение и освобождение памяти происходит с помощью средств ОС.

При обычной работе освобождающаяся часть ОЗУ возвращается операционной системе и может быть выделена другому процессу.

При установленном параметре wipe_memctx_on_free освобождающаяся часть оперативной памяти, принадлежащая контексту, предварительно заполняется нулевыми байтами.

В настоящее время вся память выделяется в рамках контекстов. Однако рекомендуется также установить параметр wipe_mem_on_free, чтобы обнулять и освобождаемую память, не входящую ни в какой контекст.

Очистка журнала упреждающей записи

Журнал упреждающей записи (WAL) — это стандартный метод обеспечения целостности данных. Основная идея WAL состоит в том, что изменения в файлах с данными (где находятся таблицы и индексы) должны записываться только после того, как эти изменения были занесены в журнал, т. е. после того, как записи журнала, описывающие данные изменения, будут сохранены на постоянное устройство хранения. Если следовать этой процедуре, то записывать страницы данных на диск после подтверждения каждой транзакции нет необходимости, потому что мы в случае сбоя есть возможность восстановить базу данных с помощью журнала: любые изменения, которые не были применены к страницам с данными, могут быть воссозданы из записей журнала.

Результатом использования WAL является значительное уменьшение количества запросов записи на диск, потому что для гарантии, что транзакция подтверждена, в записи на диск нуждается только файл журнала, а не каждый файл данных, измененный в результате транзакции. Файл журнала записывается последовательно; таким образом, затраты на синхронизацию журнала намного меньше, чем затраты на запись страниц с данными. Это особенно справедливо для серверов, которые обрабатывают много небольших транзакций.

Контрольные точки — это точки в последовательности транзакций, в которых гарантируется, что файлы с данными и индексами были обновлены всей информацией, записанной перед контрольной точкой. Во время контрольной точки все страницы данных, находящиеся в памяти, сохраняются на диск, а в файл журнала записывается специальная запись контрольной точки. В случае сбоя процедура восстановления ищет последнюю запись контрольной точки, чтобы определить эту точку в журнале, от которой процедура должна начать операцию воспроизведения изменений. Любые изменения файлов данных перед этой точкой гарантированно находятся уже на диске. Таким образом, после контрольной точки, сегменты журнала, которые предшествуют записи воспроизведения, больше не нужны и могут быть удалены или перезаписаны.

Сервер базы данных постоянно хранит определенное количество сегментов, которое определяется параметром min_wal_size. Сегменты находятся в каталоге $PGDATA/pg_xlog. В случаях большой нагрузки количество сегментов может увеличиваться и достигать max_wal_size или несколько больше. Пока размер сегментов превышает порог, заданный min_wal_size, при

освобождении сегменты удаляются; в противном случае сегменты будут циклически перезаписываться.

WAL также делает возможным поддержку резервного копирования и восстановления на определенный момент времени, а также репликацию данных между серверами. В зависимости от того, для какой цели используется журнал, он может содержать все изменения или только часть (что определяется параметром wal_level). Таким образом, журнал может содержать информацию, подлежащую защите.

При включенном параметре wipe_xlog_on_free, сегмент WAL будет заполнен нулевыми байтами перед тем, как файл будет удален или перезаписан.

3.1.5. Регистрация событий

Для регистрации событий безопасности в СУБД «СИНЕРГИЯ-БД» используется модуль pgaudit. Он предоставляет возможность детального журналирования различных событий безопасности.

Модуль pgaudit работает параллельно со стандартными средствами журналирования PostgreSQL (logging collector) и не зависит от них. Журнал регистрации событий безопасности для модуля pgaudit формируется отдельно от журнала сервера.

При старте СУБД «СИНЕРГИЯ-БД» запускаются два специальных фоновых процесса. Один процесс (pgaudit configuration worker), считывает из конфигурационного файла pgaudit.conf информацию о событиях безопасности, которые необходимо регистрировать. В дальнейшем именно этот процесс принимает решение о том, нужно ли заносить в журнал произошедшее событие. Второй процесс (pgaudit logging worker) непосредственно выполняет запись события безопасности.

Конфигурационный файл pgaudit.conf располагается в директории с данными (PGDATA) и представляет собой текстовый файл, который может редактироваться средствами ОС. Для изменения файла средствами SQL, модуль pgaudit предлагает несколько функций и представление pgaudit_settings для просмотра содержимого.

Все регистрируемые события относятся к следующим классам:

– команды DDL для создания, изменения и удаления объектов СУБД (базы данных, табличные пространства, схемы, таблицы, представления, последовательности, языки, функции);

– команды управления доступом к объектам (GRANT, REVOKE, MAC LABEL, MAC CCR, CHMAC);

– команды DML для доступа к объектам (INSERT, UPDATE, DELETE, SELECT, TRUNCATE для таблиц и/или представлений, EXECUTE для функций).

– события подключения и отключения к базе данных;

– все команды, выполняемые конкретным пользователем.

События безопасности могут регистрироваться как в централизованной системе протоколирования ОС (syslog), так и в файлах ОС. Возможна одновременная запись как в syslog,

так и в файлы.

Запись в файлы выполняется в формате csv. Каждое событие записывается отдельной строкой и состоит из следующих полей:

- дата и время события
- имя пользователя
- имя базы данных
- идентификатор серверного процесса (PID)
- уровень важности сообщения: INFO или ERROR
- порядковый номер команды в сессии
- номер вложенной команды для сложных команд (CREATE TABLE … AS SELECT ...)
- название оператора
- тип объекта
- название объекта
- результат выполнения оператора: SUCCESS или FAILURE
- текст SQL-команды
- параметры команды (например, для PREPARE)

Для файлов, в которые ведется запись регистрируемых событий, можно настроить директорию расположения и организовать ротацию. Переключение на запись в новый файл может происходить либо по истечении указанного времени, либо при превышении заданного размера файла журнала. Настройка ротации журнальных файлов позволяет организовать схему работы с очисткой событий безопасности по истечении заданного времени.

Пользователь СУБД «СИНЕРГИЯ-БД» с атрибутом SUPERUSER выдает доступ к модулю pgaudit и файлам журнала только пользователю с ролью администратора информационной безопасности.

3.1.6. Контроль целостности КСЗ

Механизм контроля целостности состоит из двух частей:

1. Утилита вычисления и проверки контрольной суммы;
2. Встроенная в СУБД возможность проверять контрольную сумму при запуске сервера.

К контролируемым объектам относятся: неизменяемые файлы, дополнительные файлы и таблицы системного каталога.

Неизменяемые файлы — исполняемые программы, библиотеки и прочие файлы, которые что ни при каких обстоятельствах не должны изменяться.

Дополнительные файлы — контролируемые файлы, которые могут изменяться администратором (например, конфигурационные).

Таблицы системного каталога — контролируемая выборка данных, относящаяся к КСЗ.

Настройки хранятся в конфигурационных файлах. Для неизменяемых файлов используется один конфигурационный файл; для дополнительных файлов и таблиц системного каталога может быть создано несколько конфигураций — по одной для каждого кластера, обслуживаемого установленной СУБД.

Контрольные суммы могут проверяться автоматически при запуске сервера СУБД, а также по требованию с помощью утилиты pg_integrity_check.

Для организации периодического контроля можно воспользоваться демоном cron.

4. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

4.1. Входные данные

Главное назначение СУБД «СИНЕРГИЯ-БД» - это обработка и хранение данных в реляционных базах данных. Поэтому к основным входным данным относятся загружаемые в СУБД табличные строки.

Помимо этого к входным данными для СУБД «СИНЕРГИЯ-БД» можно отнести:

- загружаемые в таблицы данные;
- обращение субъектов доступа (ролей) к защищаемым объектам доступа — базам данных и их элементам, таким, как таблицы, столбцы и строки, функции и т. п.;
- атрибуты ролей, определяющие их полномочия и взаимодействие с системой аутентификации клиентов;
- привилегии ролей, определяющие дискреционные правила разграничения доступа к объектам баз данных;
- политики защиты строк, определяющие правила доступа к строкам таблиц;
- метки конфиденциальности защищаемых объектов БД;
- мандатный признак CCR для объектов-контейнеров.

4.2. Выходные данные

Выходными данными для СУБД является результат использования субъектом доступа защищаемых объектов, предоставленных ему в соответствии с установленными ПРД.

К таким результатам могут относиться: подключение к базе данных; получение результата запроса к объектам БД; добавление, изменение или удаление данных в БД.

5. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

СУБД — система управления базами данных

БД — база данных

КСЗ — комплекс средств защиты

НСД — несанкционированный доступ

ОЗУ — оперативное запоминающее устройство

ОС — операционная система

ПРД — правила разграничения доступа

SQL — Structured Query Language (язык структурированных запросов)

WAL — Write-Ahead Log (журнал упреждающей записи)

Лист регистрации изменений									
Номера листов					Всего листов в документе	Номер документа	Входящий № сопроводит. документа и дата	Подпись	Дата
Изм.	Изме-нённых	Заме-нённых	новых	Аннулиро-ванных					