Средства Мониторинга Безопасности Баз Данных в Распределенных Компьютерных Системах на Основе Сенсоров

В.Е. Мухин, Я.И. Корнага

Национальный технический университет Украины "Киевский политехнический институт", пр. Победы, 37, Киев, Украина

v_mukhin@mail.ru, slovyan_k@ukr.net

Аннотация. *В работе рассмотрена общая архитектура средств мониторинга безопасности баз данных в распределенных компьютерных системах на основе сенсоров. Предложен специальный механизм сенсоров для системы мониторинга безопасности, который выполняет анализ активных в данный момент процессов в базе данных и позволяет выявить подозрительные события. В результате разработан специализированный механизм поддержки мониторинга безопасности для комплексного мониторинга всех действий с базой данных, что обеспечивает дополнительную защиту от несанкционированных действий, в том числе в режиме реального времени.*

Ключевые слова

Базы данных, распределенные компьютерные системы, мониторинг безопасности, сенсор

1 Введение

Локально-ориентированные подходы к мониторингу безопасности баз данных (БД) в распределенных компьютерных системах имеют существенный недостаток: они снижают общую производительность обработки данных в БД ввиду того, что локально-ориентированные средства либо реализуют собственный механизм аудита, либо используют API-функции ядра СУБД (Системы Управления Базой Данных) для взаимодействия с базой данных, что требует значительных временных затрат [1, 2, 3].

Таким образом, несмотря на то, что локально-ориентированный подход является достаточно эффективным для реализации основных требований к мониторингу действий легальных пользователей, а также для выявления и нейтрализации вторжений, на практике сетевой подход получает все большее распространение, несмотря на свои определенные недостатки.

2 Общая архитектура средств мониторинга безопасности баз данных на основе сенсоров

Предлагается реализовать средства мониторинга на основе прямого доступа к памяти, выделяемой операционной системой для СУБД, в частности, общей кэш-памяти, например SGA (System Global Area) в СУБД Oracle, или процедурный кэш в MS SQL [4,5].

Архитектура предлагаемого средства включают следующие компоненты: специальный сканер отпечатков пальцев, программу-агент, размещенную на сервере базы данных, которая проводит мониторинг всех действий, а также сервер на основе JavaEE, выполняющий сбор данных со всех сенсоров (рис. 1).

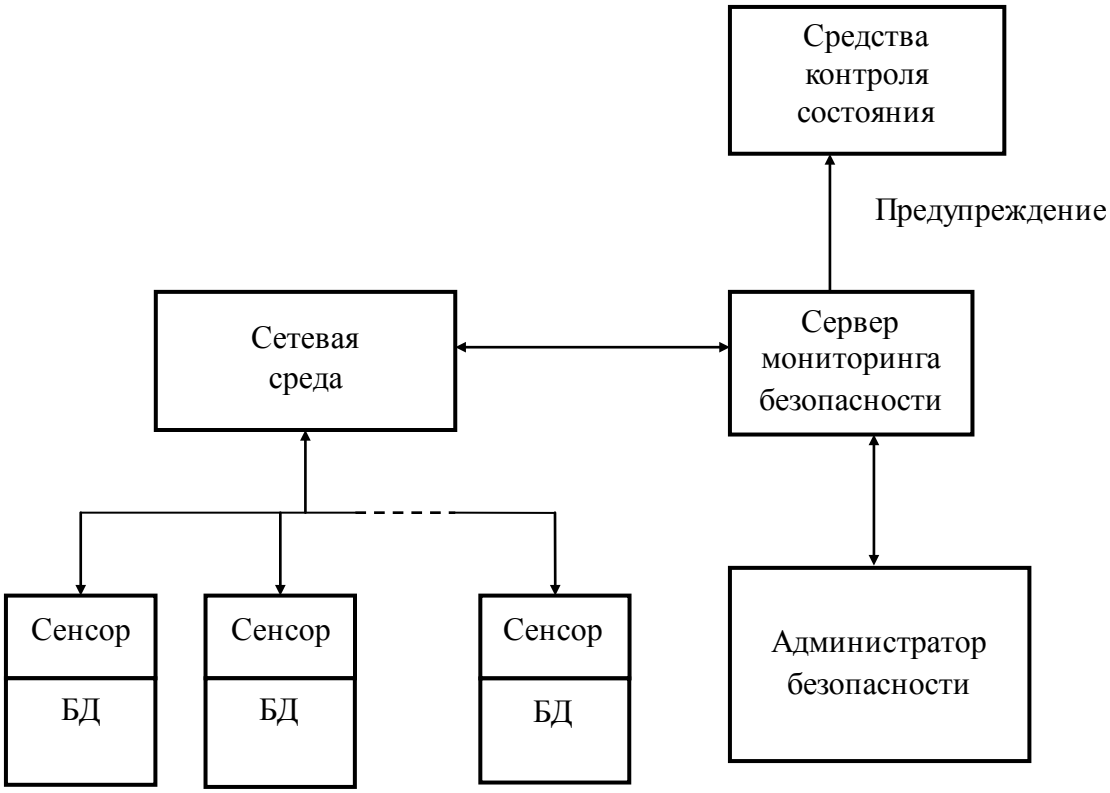

Рис. 1. Архитектура средств мониторинга безопасности баз данных на основе сенсоров

3 Архитектура сенсора для предложенной системы мониторинга безопасности

Сенсор в предложенной системе мониторинга безопасности (СМБ) представляет собой отдельный механизм, функционирующий на сервере базы данных, при этом сенсор достаточно надежно защищен от внешних воздействий. (рис. 2).

Сенсор автоматически идентифицирует все события, происходящие в распределенной компьютерной системе. В процессе работы сенсор прикрепляется к временно-выделяемой памяти (SGA в случае Oracle) и циклически запускает опрос по мониторингу событий путем выборки данных из памяти с определенной частотой. На каждом цикле сенсор выполняет анализ активных в данный момент процессов в каждой сессии в базе данных и определяет с помощью предопределенных и установленных администратором правил подозрительные события. Сообщения об подозрительных событиях отправляются на сервер базы данных для их дальнейшего анализа и генерации соответствующих сообщений.

Сенсор также может принудительно досрочно прекращать сессии работы в случае определенных нарушений со стороны пользователей. С другой стороны, для его работы требуется незначительные вычислительные ресурсы системы и он практически не влияет на операции ввода/вывода. Упреждающая функция сенсора может быть реализована с использованием DDL-триггеров, которые выборочно задерживают выполнение команд DDL и DCL на короткие промежутки времени (несколько миллисекунд), что позволяет сенсору своевременно реагировать на опасные действия.

Выделенный сервер в предложенной системе мониторинга безопасности может управлять несколькими сенсорами из различных баз данных, а также он поддерживает функцию масштабирования числа сенсоров. Сервер достаточно просто интегрируется в структуру системы безопасности, что позволяет повысить эффективность общего управления безопасностью компьютерной системы. (рис. 3).

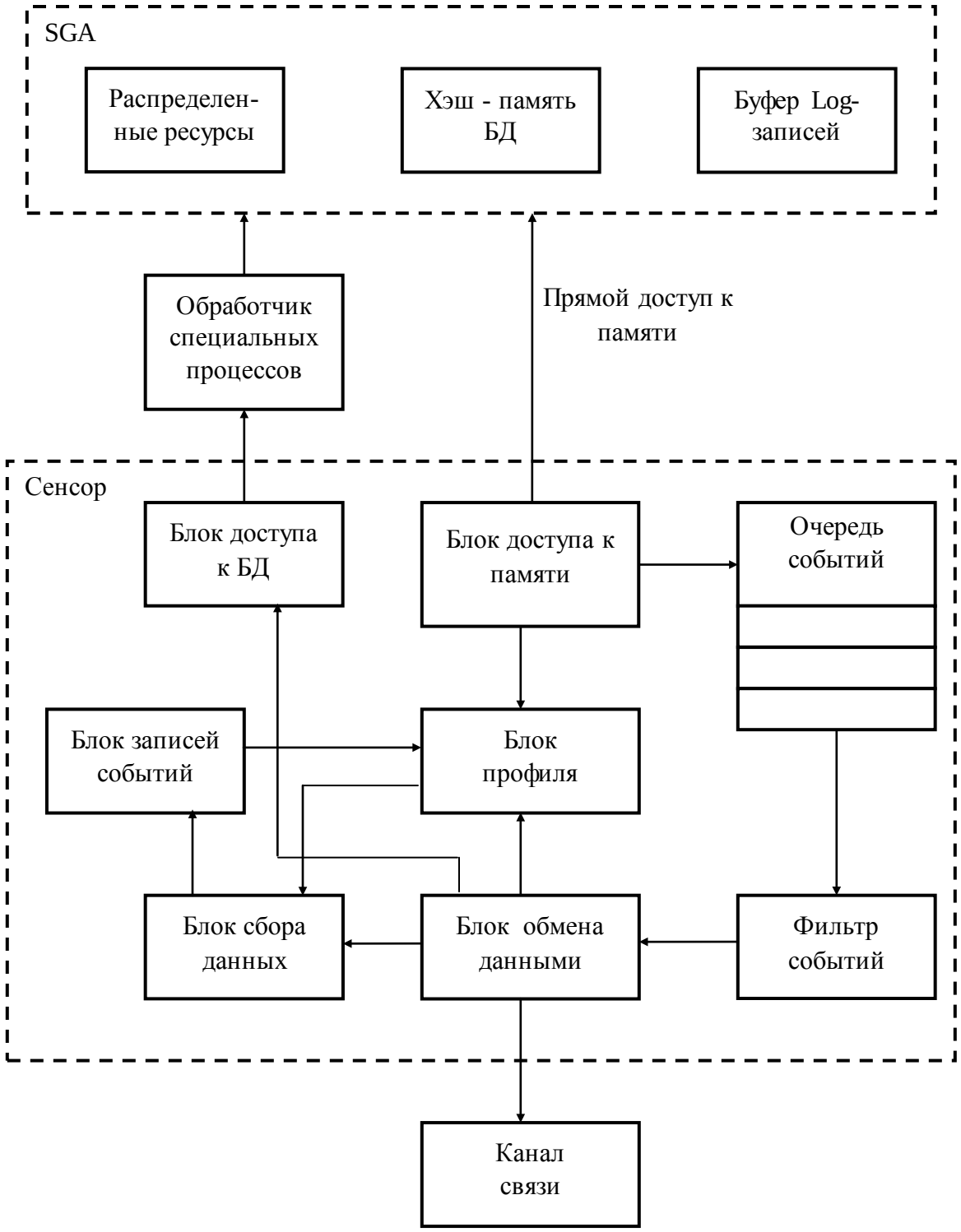

Рис. 2. Архитектура сенсора системы мониторинга безопасности

Структура предлагаемой системы также обеспечивает распределение прав и полномочий между субъектами, что является одним из основных требований к системе безопасности. Администратор безопасности данной системы является лицом, определяющим правила политики безопасности и получающим сообщения об инцидентах от разных источников, в частности от администратора компьютерной системы и администратора базы данных. Предложенная система позволяет, с одной стороны, обеспечить требуемый уровень защиты компьютерной системы, а с другой — непрерывное выполнение вычислительных операций системой.

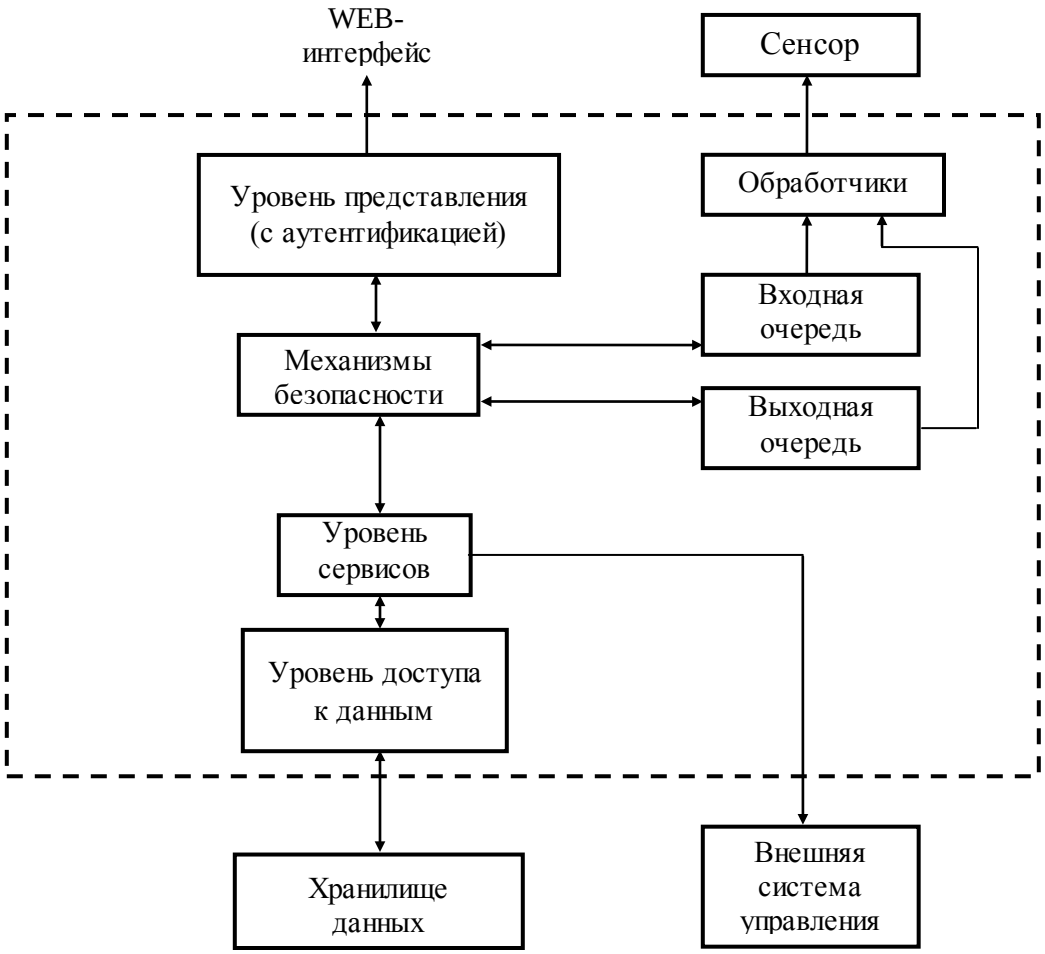

Рис. 3. Архитектура сервера мониторинга безопасности

4 Заключение

В работе предложен специализированный механизм реализации мониторинга безопасности баз данных, с помощью которого выполняется комплексный мониторинг всех действий с базой данных и обеспечивается защиту от несанкционированных действий легальных пользователей.

Также данный механизм позволяет выполнить детальный мониторинг транзакций, запросов, объектов и сохраненных процедур базы данных с уведомлениями об инцидентах в режиме реального времени и предупреждением вторжений.

Кроме того, предложенный механизм позволяет провести отслеживание новых обнаруженных уязвимых мест базы данных и оперативно устранить эти уязвимости, что важно в практических приложениях.

Литература

[1] Скотт В. Эмблер, Прамодкумар Дж. Садаладж, Refactoring Databases: Evolutionary Database Design. Москва: Вильямс, 321, 2007.

[2] А. И. Баранчиков, П. А. Баранчиков, А. Н. Пылькин, Алгоритмы и модели ограничения доступа к записям баз данных, Санк-Петербург: Горячая Линия – Телеком, 182, 2011.

[3] С. Н. Смирнов, Безопасность систем баз данных, Москва: Гелиос АРВ, 352, 2007.

[4] Томас Кайт, Oracle для профессионалов. Архитектура, методики программирования и особенности версий 9i, 10g и 11g, Москва: Вильямс, 848, 2011.

[5] Б. П. Арсеньев, С. А Яковлев, Интеграция распределенных баз данных, Москва: Лань, 464, 2001.