Содержание:

Введение

Постоянные предложения приобрести различные (в большинстве своем ведомственные) базы данных свидетельствуют о том, что продажа конфиденциальных сведений о гражданах и юридических лицах стала отдельным видом бизнеса. Если появление очередной опубликованной базы для граждан является просто еще одним малоприятным фактом обнародования сведений об их частной жизни, то на некоторых предприятиях это может отрицательно повлиять на бизнес. Например, для оператора сотовой связи распространение базы биллинга может обернуться существенным оттоком абонентов к более «надежному» оператору-конкуренту. Поэтому оператору подчас экономически более выгодно найти «производителя», подготовившего украденную базу к продаже, и выкупить весь тираж. Но проблема перекрытия возможных утечек при этом остается весьма актуальной.

Основная часть

Защита баз данных является одной из самых сложных задач, стоящих перед подразделениями, отвечающими за обеспечение информационной безопасности. С одной стороны, для работы с базой необходимо предоставлять доступ к данным всем сотрудникам, кто по долгу службы должен осуществлять сбор, обработку, хранение и передачу конфиденциальных данных. С другой стороны, укрупнение баз данных далеко не всегда имеет централизованную архитектуру, в связи с чем действия нарушителей становятся все более изощренными. При этом четкой и ясной методики комплексного решения задачи защиты баз данных, которую можно было бы применять во всех случаях, не существует, в каждой конкретной ситуации приходится находить индивидуальный подход.

Классический взгляд на решение данной задачи включает обследование предприятия с целью выявления таких угроз, как хищения, утрата, уничтожение,

модификация, отказ от подлинности. На втором этапе следует составление математических моделей основных информационных потоков и возможных нарушений, моделирование типовых действий злоумышленников; на третьем - выработка комплексных мер по пресечению и предупреждению возможных угроз с помощью правовых, организационно-административных и технических мер защиты. Однако разнообразие деятельности предприятий, структуры бизнеса, информационных сетей и потоков информации, прикладных систем и способов организации доступа к ним и т. д. не позволяет создать универсальную методику решения.

Долгое время защита баз данных ассоциировалась с защитой локальной сети предприятия от внешних атак хакеров, борьбой с вирусами и т. п. Последние аналитические отчеты консалтинговых компаний выявили другие, более важные направления защиты информационных ресурсов компаний. Исследования убедительно показали, что от утечки информации со стороны персонала и злонамеренных действий «всесильных» администраторов баз данных не спасают ни межсетевые экраны, ни VPN, ни даже «навороченные» системы обнаружения атак и анализа защищенности. Неавторизованный доступ к данным и кража конфиденциальной информации являются главными составляющими потерь предприятий после ущерба, наносимого вирусами.

Один из основных выводов отчета CSI/FBI - значительно возросший ущерб от такой угрозы, как кража конфиденциальных данных. Каждая американская компания в среднем потеряла 355,5 тыс. долл. только из-за утечек конфиденциальных данных за прошедшие 12 месяцев. Средний размер потерь от действий инсайдеров составил 300 тыс. долл. (максимальный - 1,5 млн долл.). Решение вопросов персонифицированного доступа к конфиденциальным данным позволяет выявлять злоумышленника с помощью информации, неопровержимо доказывающей его вину. Это, в свою очередь, невозможно без применения самых современных способов аутентификации и управления доступом.

Сформулировать основные причины несанкционированного доступа к данным и поставленного в ряде случаев на промышленные рельсы сбыта баз данных, содержащих персональные данные абонентов, партнеров или сотрудников и коммерческие тайны компаний.

Итак, имеются следующие исходные данные:

- многие не догадываются о том, что их базы данных крадут;

- кража и причиненный ущерб имеют латентный характер;

- если факт кражи данных установлен, большинство компаний замалчивают причиненный ущерб. Одна из причин этого - отсутствие реальных механизмов сбора доказательной базы по факту кражи конкретным пользователем ресурсов;

- технологии, позволяющие строго персонифицировать действия пользователей и разграничить их права, неизвестны большинству руководителей;

- возможность защиты данных от системных администраторов также малоизвестна, руководители предпочитают считать их наиболее лояльными сотрудниками;

- бюджеты на информационную безопасность, как правило, невелики. Это не позволяет решить проблему комплексно (введение штатных единиц, отвечающих за информационную безопасность.

Основные требования по безопасности данных, предъявляемые к БД и СУБД, во многом совпадают с требованиями, предъявляемыми к безопасности данных в компьютерных системах - контроль доступа, криптозащита, проверка целостности, протоколирование и т.д.

Под управлением целостностью в БД понимается защита данных в БД от неверных (в отличие от несанкционированных) изменений и разрушений. Поддержание целостности БД состоит в том, чтобы обеспечить в каждый момент времени корректность (правильность) как самих значений всех элементов данных, так и взаимосвязей между элементами данных в БД . С поддержанием целостности связаны следующие основные требования.

Обеспечение достоверности. В каждый элемент данных информация заносится точно в соответствии с описанием этого элемента .Должны быть предусмотрены механизмы обеспечения устойчивости элементов данных и их логических взаимосвязей к ошибкам или неквалифицированным действиям пользователей.

Управление параллелизмом. Нарушение целостности БД может возникнуть при одновременном выполнении операций над данными, каждая из которых в отдельности не нарушает целостности БД . Поэтому должны быть предусмотрены механизмы управления данными, обеспечивающие поддержание целостности БД при одновременном выполнении нескольких операций.

Восстановление. Хранимые в БД данные должны быть устойчивы по отношению к неблагоприятным физическим воздействиям (аппаратные ошибки, сбои питания и т .п .) и ошибкам в программном обеспечении . Поэтому должны быть предусмотрены механизмы восстановления за предельно короткое время того состояния БД, которое было перед появлением неисправности.

Вопросы управления доступом и поддержания целостности БД тесно соприкасаются между собой, и во многих случаях для их решения используются одни и те же механизмы. Различие между этими аспектами обеспечения безопасности данных в БД состоит в том, что управление доступом связано с предотвращением преднамеренного разрушения БД, а управление целостностью - с предотвращением непреднамеренного внесения ошибки.

Большинство систем БД представляют собой средство единого централизованного хранения данных. Это значительно сокращает избыточность данных, упрощает доступ к данным и позволяет более эффективно защищать данные . Однако, в технологии БД возникает ряд проблем, связанных, например, с тем, что различные пользователи должны иметь доступ к одним данным и не иметь доступа к другим . Поэтому, не используя специальные средства и методы, обеспечить надежное разделение доступа в БД практически невозможно.

Большинство современных СУБД имеют встроенные средства, позволяющие администратору системы определять права пользователей по доступу к различным частям БД, вплоть до конкретного элемента . При этом имеется возможность не только предоставить доступ тому или иному пользователю, но и указать разрешенный тип доступа: что именно может делать конкретный пользователь с конкретными данными (читать, модифицировать, удалять и т . п .), вплоть до реорганизации всей БД Таблицы (списки) управления доступом широко используются в компьютерных системах, например, в ОС для управления доступом к файлам .Особенность использования этого средства для защиты БД состоит в том, что в качестве объектов защиты выступают не только отдельные файлы (области в сетевых БД, отношения в реляционных БД), но и другие структурные элементы БД: элемент, поле, запись, набор данных.

Нарушение целостности данных может быть вызвано рядом причин:

сбои оборудования, физические воздействия или стихийные бедствия;

ошибки санкционированных пользователей или умышленные действия несанкционированных пользователей;

программные ошибки СУБД или ОС;

ошибки в прикладных программах;

совместное выполнение конфликтных запросов пользователей и др.

Нарушение целостности данных возможно и в хорошо отлаженных системах . Поэтому важно не только не допустить нарушения целостности, но и своевременно обнаружить факт нарушения целостности и оперативно восстановить целостность после нарушения.

Поддержание целостности на основе приведенных выше ограничений целостности представляет собой достаточно сложную проблему в системе БД даже с одним пользователем . В системах, ориентированных на многопользовательский режим работы, возникает целый ряд новых проблем, связанных с параллельным выполнением конфликтующих запросов пользователей . Прежде , чем рассмотреть механизмы защиты БД от ошибок, возникающих в случае конфликта пользовательских запросов, раскроем ряд понятий, связанных с управлением параллелизмом.

Важнейшим средством механизма защиты целостности БД выступает объединение совокупности операций, в результате которых БД из одного целостного состояния переходит в другое целостное состояние, в один логический элемент работы, называемый транзакцией. Суть механизма транзакций состоит в том, что до завершения транзакции все манипуляции с данными проводятся вне БД, а занесение реальных изменений в БД производится лишь после нормального завершения транзакции.

С точки зрения безопасности данных такой механизм отображения изменений в БД очень существенен . Если транзакция была прервана, то специальные встроенные средства СУБД осуществляют так называемый откат - возврат БД в состояние, предшествующее началу выполнения транзакции (на самом деле откат обычно заключается просто в невыполнении изменений, обусловленных ходом транзакции, в физической БД) . Если выполнение одной транзакции не нарушает целостности БД, то в результате одновременного выполнения нескольких транзакций целостность БД может быть нарушена . Чтобы избежать подобного рода ошибок, СУБД должна поддерживать механизмы, обеспечивающие захват транзакциями модифицируемых элементов данных до момента завершения модификации так называемые блокировки . При этом гарантируется, что никто не получит доступа к модифицируемому элементу данных, пока транзакция не освободит его .

Применение механизма блокировок приводит к новым проблемам управления параллелизмом, в частности, к возникновению ситуаций клинча двух транзакций. Причем, если некоторая транзакция пытается блокировать объект, который уже блокирован другой транзакцией, то ей придется ждать, пока не будет снята блокировка объекта транзакцией, установившей эту блокировку . Иными словами, блокировку объекта может выполнять только одна транзакция

Восстановление данных - процесс получения доступа к файлам, записанным на том или ином носителе информации, которые стали недоступными вследствие программного сбоя, выхода носителя из строя или ошибочных действий пользователя. Возможность восстановления данных при помощи специальных программ существует в том случае, если они не были перезаписаны другой информацией. Также во многом успех зависит от сохранности структуры файловой системы и работоспособности носителя вообще.

Как известно, данные на любом современном носителе информации на самом низком уровне хранятся в виде битовых последовательности нулей и единичек. То есть, в виде намагниченных/заряженных секторов (1) или их отсутствия (0).

Однако, Windows и прочие операционные системы для ускорения и упрощения доступа к данным работают на более высоких уровнях с использованием различных файловых систем. Файловая система представляет собой программную прослойку для эффективного взаимодействия ОС с информацией на физическом носителе. Она состоит из двух частей: системной области и области данных. Системная область хранит в себе загрузочный сектор (отвечает за возможность загрузки с носителя и его корректное распознавание), а также ряд секторов, хранящих индексные таблицы файлов и иную служебную информацию.

Вся информация физически хранится в области данных, однако, ведомости о файлах находятся в системной области. Механизм такой организации работы выглядит следующим образом: при подключении носителя к компьютеру система не сканирует весь диск на наличие файлов, а быстро считывает данные о них из системной области. Так же ОС взаимодействует с носителем, например, при удалении данных: физически файлы не уничтожаются, а удаляются лишь ссылки на них в файловой таблице. Это даёт системе основания считать "освободившиеся" кластеры носителя пустыми и пригодными для дальнейшей перезаписи. Таким образом, первый случай, когда восстановление данных возможно - исчезновение ссылки на файл в файловой таблице при условии, что файл не был перезаписан иными данными. Второй распространённый случай - форматирование носителя.

Существует три типа форматирования:

- Быстрое форматирование - стирается только файловая таблица, но не затрагивается область данных. При таком форматировании шансы на восстановления весьма высоки (при условии, что на отформатированную флешку ничего больше не записывалось).

- Полное форматирование - стирается и системная область, и область данных. Этот тип форматирования предусматривает полную очистку носителя, однако, для ускорения процесса стирается область данных не полностью, а фрагментами. Это даёт (пусть и небольшой) шанс на восстановление нужных файлов.

- Низкоуровневое форматирование - все секторы носителя информации заполняются нулями. После такого форматирования восстановить что-либо практически нереально, поскольку все данные уничтожаются полностью. В Windows штатно отсутствует возможность низкоуровневого форматирования, поэтому даже после полной очистки диска её средствами восстановление данных теоретически возможно! Аналогично можно попробовать восстановить информацию при сбоях файловых систем, которыми часто "грешат" флешки. При таких сбоях обычно частично или полностью уничтожается системная область и флешка требует форматирования:

Как уже отмечалось, возникновение сбоев в аппаратном или программном обеспечении может вызвать необходимость восстановления и быстрого возвращения в состояние, по возможности близкое к тому, которое было перед возникновением сбоя (ошибки). К числу причин, вызывающих необходимость восстановления, зачастую относится и возникновение тупиковой ситуации.

Можно выделить три основных уровня восстановления:

Оперативное восстановление, которое характеризуется возможностью восстановления на уровне отдельных транзакций при ненормальном окончании ситуации манипулирования данными (например, при ошибке в программе).

Промежуточное восстановление .Если возникают аномалии в работе системы (системно-программные ошибки, сбои программного обеспечения, не связанные с разрушением БД), то требуется восстановить состояние всех выполняемых на момент возникновения сбоя транзакций.

Длительное восстановление. При разрушении БД в результате дефекта на диске восстановление осуществляется с помощью копии БД. Затем воспроизводят результаты выполненных с момента снятия копии транзакций и возвращают систему в состояние на момент разрушения

Прекращение выполнения транзакции вследствие появления сбоя нарушает целостность БД . Если результаты такого выполнения транзакции потеряны, то имеется возможность их воспроизведения на момент возникновения сбоя . Таким образом, понятие транзакции играет важную роль при восстановлении. Для восстановления целостности БД транзакции должны удовлетворять следующим требованиям:

необходимо, чтобы транзакция или выполнялась полностью, или не выполнялась совсем;

необходимо, чтобы транзакция допускала возможность возврата в первоначальное состояние, причем, для обеспечения независимого возврата транзакции в начальное состояние монопольную блокировку необходимо осуществлять до момента завершения изменения всех объектов;

необходимо иметь возможность воспроизведения процесса выполнения транзакции, причем, для обеспечения этого требования, совместную блокировку необходимо осуществлять до момента завершения просмотра данных всеми транзакциями.

В процессе выполнения любой транзакции наступает момент ее завершения . При этом все вычисления, сделанные транзакцией в ее рабочей области, должны быть закончены, копия результатов ее выполнения должна быть записана в системный журнал . Подобные действия называют операцией фиксации. При появлении сбоя целесообразнее осуществлять возврат не в начало транзакции, а в некоторое промежуточное положение. Точку, куда происходит такой возврат, называют точкой фиксации (контрольной точкой). Пользователь может установить в процессе выполнения транзакции произвольное количество таких точек . Если в ходе выполнения транзакции достигается точка фиксации, то СУБД автоматически осуществляет указанную выше операцию.

Заключение

Основным средством, используемым при восстановлении, является системный журнал, в котором регистрируются все изменения, вносимые в БД каждой транзакцией . Возврат транзакции в начальное состояние состоит в аннулировании всех изменений, которые осуществлены в процессе выполнения транзакции . Такую операцию называют откатом . Для воспроизведения результатов выполнения транзакции можно, используя системный журнал, восстановить значения проведенных изменений в порядке их возникновения, либо выполнить транзакцию повторно . Воспроизведение результатов выполнения транзакции с использованием системного журнала называется раскруткой. Раскрутка является достаточно сложной, но необходимой операцией механизмов восстановления современных БД