А. М. Бондарь
Московский инженерно-физический институт (государственный университет)

СРЕДСТВА ШИФРОВАНИЯ ДАННЫХ В СУБД Oracle

Статья посвящена предоставляемым СУБД Oracle 9i/10g возможностям обеспечения конфиденциальности данных, обрабатываемых и хранимых в БД. Рассматриваются ключевые аспекты следующих функциональностей СУБД Oracle, решающих эту задачу: прозрачное шифрование данных, выборочное шифрование данных и обеспечение конфиденциальности данных, передаваемых по сети. Приводятся вариант защиты автоматизированной системы на базе СУБД Oracle с использованием рассмотренных технологий и рекомендации по их применению.

Задача обеспечения конфиденциальности данных имеет место быть практически в любой автоматизированной системе (АС). Система управления базами данных (СУБД) Oracle предоставляет разнообразные средства защиты хранимой в базе данных (БД) информации. Например, использование функционала выборочного шифрования данных позволяет защищать обрабатываемую и хранимую в СУБД информацию, в то время как помимо этого имеется возможность обеспечения конфиденциальности данных, передаваемых по сети между клиентом и сервером СУБД. В 10-й версии СУБД Oracle появилась возможность обеспечивать конфиденциальность данных и на уровне операционной системы (ОС) благодаря новой функциональности «прозрачное шифрование данных». Таким образом, в зависимости от требований к АС можно построить достаточно сложную подсистему обеспечения конфиденциальности информации, не прибегая к использованию дополнительных внешних средств (т. е. используя только предоставляемый СУБД Oracle набор функциональностей).

Прозрачное шифрование данных

Прозрачное шифрование данных (ПШД) позволяет защищать данные путем шифрования непосредственно файлов БД (или их частей) на уровне ОС, причем секретные ключи шифрования хранятся вне БД, во внешнем защищенном модуле.

Работа ПШД

При использовании ПШД можно защитить целые таблицы данных или их отдельные столбцы. При этом авторизованные пользователи БД будут иметь прямой доступ к этим данным, как если бы они и не были зашифрованы. Дело в том, что процесс шифрования/расшифрования данных проходит «прозрачно» (незаметно) для авторизованного пользователя БД.

Таблица БД может содержать один или несколько зашифрованных столбцов данных, при этом у каждой такой таблицы будет отдельный ключ шифрования (для всех столбцов). Подобные ключи хранятся только в зашифрованном на мастер-ключе виде в словарной таблице БД. Сам мастер-ключ хранится в защищенном внешнем модуле вне БД — в «*электронном бумажнике*» (wallet) и доступен

только для администратора по безопасности. Внешний модуль *электронный бумажник* также отвечает за генерацию секретных ключей и шифрование/расшифрование данных.

Настройка и использование ПШД

Первым шагом при настройке ПШД является конфигурирование *электронного бумажника*, который может быть разделяемым для всех экземпляров БД в системе или индивидуальным для каждого из них. В обязанности администратора по безопасности помимо этого входит задание мастер-ключа, который будет использоваться в ПШД. Это реализуется командой:

ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY *password*,

где *password* — это пароль к внешнему защищенному модулю, определенный в конфигурационном файле СУБД sqlnet.ora.

Как только мастер-ключ задан, при последующих перезапусках сервера СУБД потребуется только открывать *электронный бумажник* для использования (команда ALTER SYSTEM SET WALLET OPEN IDENTIFIED BY *password*). Для того чтобы запретить функциональность ПШД на работающем сервере СУБД, т. е. сделать недоступными для пользователей все зашифрованные данные (так как процедуры шифрования/расшифрования «на лету» перестанут работать), достаточно закрыть *электронный бумажник* (команда ALTER SYSTEM SET WALLET CLOSE). Администратор по безопасности может сделать неактивным мастер-ключ в случае его компрометации или истечения срока действия в соответствии с принятыми в АС политиками. Тогда ранее зашифрованные данные не будут перезашифрованы, а только вновь создаваемые либо изменяемые данные будут шифроваться на новом мастер-ключе, таким образом, *электронный бумажник* поддерживает версионность мастер-ключей в системе.

По умолчанию ПШД использует симметричную криптографию, основанную на алгоритме AES (Advanced Encryption Standard — американский стандарт шифрования данных, шифр Rijndael) с ключом 192 бита, с добавлением *соли* (добавки к открытому тексту в виде случайной строки), при этом мастер-ключ генерируется случайным образом в *электронном бумажнике*. Однако имеется возможность использовать в ПШД ассиметричную криптографию (для этого требуется наличие сертификата стандарта X.509 версии 3), но это сильно повлияет на производительность операций шифрования/расшифрования.

Когда мастер-ключ установлен, можно приступить к шифрованию данных. Необходимость шифрования отдельных столбцов таблицы может быть задана при ее создании (опционально в команде CREATE TABLE) либо после путем изменения ее столбцов, но в любом случае используется опция ENCRYPT. Например, команда: ALTER TABLE employee ADD (ssn VARCHAR2(11) ENCRYPT) добавит к таблице employee столбец ssn, зашифрованный на алгоритме AES с ключом 192 бита с добавлением *соли*. Если планируется в дальнейшем индексировать этот столбец, то надо исключить добавление *соли* при шифровании параметром NO SALT в выражении ENCRYPT. Если необходимо, наоборот, отключить использование ПШД для определенного столбца, то для этих целей используется выражение DECRYPT аналогичным образом. Применяемый по умолчанию алгоритм шифрования также можно переопределить при задании ПШД с помощью параметра USING «*алгоритм*» в выражении ENCRYPT, например, выражение ENCRYPT USING '3DES168' означает, что будет использоваться алгоритм Triple DES с ключом 168 бит.

Каждая таблица при использовании ПШД может иметь, по крайней мере, один секретный ключ. Эти ключи, как и используемые алгоритмы шифрования, можно изменить в любой момент с помощью команды REKEY (ALTER TABLE employee REKEY — обновить секретный ключ).

Информацию о таблицах и их столбцах, где используется ПШД, можно почерпнуть из системного представления DBA_ENCRYPTED_COLUMNS.

Преимущества использования ПШД:

· отсутствие необходимости администрирования ключей шифрования непосредственно пользователями;

· обеспечение конфиденциальности данных БД с минимумом усилий;

· защищенное хранение мастер-ключа шифрования во внешнем модуле вне БД;

· абсолютно «прозрачный» доступ к зашифрованным данным со стороны аутентифицированных пользователей (не требующий дополнительных действий по расшифрованию данных).

К недостаткам данной функциональности можно отнести следующее:

· ряд ограничений в использовании (невозможность поиска по диапазону значений по шифрованному столбцу, невозможность использования утилит IMP/EXP с шифрованными данными и т. д.);

· дополнительная нагрузка на СУБД при работе с зашифрованными данными.

В случае если вышеописанные недостатки критичны, разумно отказаться от использования данной функциональности в пользу пакета DBMS_CRYPTO, предоставляющего функциональность выборочного шифрования данных.

Обеспечение конфиденциальности и целостности данных при передаче по сети

Сетевой трафик между сервером СУБД Oracle и клиентскими приложениями может быть защищен. Дополнительная опция СУБД Oracle Advanced Security позволяет настраивать шифрование и контроль целостности передаваемых между клиентом и сервером СУБД данных, используя такие алгоритмы шифрования, как DES, Triple DES, AES, RC4, и алгоритмы хеширования MD5 и SHA-1. Гибкие настройки делают возможным задание длины ключа для используемых алгоритмов.

Особенностью опции Oracle Advanced Security при защите сетевого трафика службы Oracle Net Service является использование алгоритма Диффи—Хеллмана для генерации части разового сессионного ключа. Другая часть сессионного ключа создается в ходе аутентификации клиента на сервере и является разделяемым секретом. Таким образом, сильный сессионный ключ состоит из двух частей: сгенерированной в ходе сессионного взаимодействия, меняющейся каждый раз, и полученной в результате аутентификации, известной только клиенту и серверу. Такой подход исключает возможность осуществления атаки «человек-по-середине», от которой не защищен алгоритм Диффи—Хеллмана.

Настройка защиты данных, передаваемых по сети

Для настройки шифрования и контроля целостности передаваемых между клиентом и сервером данных необходима установленная опция Oracle Advanced Security. В общем случае и клиент, и сервер СУБД имеют каждый по списку поддерживаемых ими алгоритмов шифрования с необходимыми настройками. При установке соединения используется первый алгоритм из списка сервера, присутствующий в списке клиента, в противном случае попытка установить соединение оборачивается неудачей, с выдачей соответствующего сообщения об ошибке.

Задача настройки параметров шифрования и контроля целостности лежит на администраторе сети либо на администраторе по безопасности. Все параметры настройки алгоритмов шифрования (список используемых алгоритмов и их параметров) задаются в файле конфигурации SQLNET.ORA как для клиента, так и для сервера СУБД. Далее приведены основные параметры настроек:

· Отдельные параметры encryption_server / encryption_client задают строгость требований при выборе используемых алгоритмов шифрования при установлении соединения между сервером и клиентом соответственно и могут принимать значения: REJECTED, ACCEPTED, REQUESTED и REQUIRED. Строгость требований возрастает в соответствии с указанным порядком значений. Аналогичные параметры есть и для конфигурации контроля целостности — crypto_checksum_server / crypto_checksum_client.

· Параметр crypto_seed позволяет задавать пользовательскую последовательность символов, используемую в дальнейшем для генерации случайных чисел (при генерации сессионного ключа).

· Параметры encryption_types_server / encryption_types_client задают списки доступных для использования алгоритмов шифрования.

· Параметры crypto_checksum_types_server / crypto_checksum_types_client задают списки доступных для использования алгоритмов контроля целостности.

Шифрование и контроль целостности передаваемых по сети данных могут быть полезны, если необходимо обеспечить безопасность всего трафика, передаваемого между клиентом и сервером СУБД (для всех приложений-клиентов). Простые и достаточно гибкие настройки делают процедуру задания конфигурации данного сервиса несложной.

Выборочное шифрование данных

СУБД Oracle, как и другие промышленные системы, построена по принципу суперпользователя, который способен прочитать любые данные в системе. Суперпользователем в СУБД Oracle является SYS, обладающий всеми привилегиями в системе и способный их раздавать другим пользователям. Таким образом, в системе всегда существует хотя бы один человек, которому необходимо оказывать высокую степень доверия. Выборочное шифрование данных предоставляет пользователю СУБД Oracle возможность защитить свои собственные данные от всех, включая администратора БД — пользователя SYS. По сути, выборочное шифрование данных возможно благодаря предоставляемому СУБД Oracle криптографическому программному интерфейсу, реализующему последние алгоритмы шифрования и хеширования.

Функциональность выборочного шифрования данных предоставляют два серверных PL/SQL пакета DBMS_CRYPTO и DBMS_OBFUSCATION_TOOLKIT (последний остался для совместимости с предыдущими версиями СУБД Oracle), поддерживающие следующие возможности:

· алгоритмы шифрования: DES (Data Encryption Standard), 3DES (Triple Data Encryption Standard), AES (Advanced Encryption Standard, с возможностью вариации длины ключа), RC4;

· алгоритмы хеширования: SHA-1, MD4, MD5;

· режимы шифрования: CBC, CFB, ECB, OFB.

Пакет DBMS_CRYPTO имеет интуитивно понятный интерфейс шифрования данных: есть две общие функции ENCRYPT и DECRYPT, а ссылка на конкретный алгоритм передается параметром. Поддержка многобайтовых кодировок позволяет передавать зашифрованные данные между различными БД. В пакете DBMS_CRYPTO также поддерживается возможность генерации псевдослучайных последовательностей чисел, которые могут быть использованы при генерации ключей шифрования.

Для того чтобы пользователь мог работать с пакетом DBMS_CRYPTO, администратор БД (или администратор по безопасности) должен предоставить ему соответствующую привилегию (например, GRANT EXECUTE ON DBMS_CRYPTO TO User1).

Однако при использовании возможностей выборочного шифрования данных остается открытым вопрос управления криптографическими ключами. Главным образом, задача состоит в том, как эти ключи хранить: на уровне БД, на уровне ОС, использовать ли дополнительные преобразования над ключами или шифруемыми данными. В целом же выборочное шифрование данных является хорошим решением для защиты данных в АС, обеспечивая достаточный функционал и не требуя подключения внешних криптографических библиотек.

Подсистема обеспечения конфиденциальности информации

Рассмотренные выше методы криптографической защиты информации позволяют обеспечивать конфиденциальность данных на разных этапах обработки данных в АС. Какие средства и в каких случаях применять, во многом зависит от специфики АС и предъявленных к ней требований. Однако можно представить типовой вариант защиты АС на основе СУБД Oracle и рекомендации, которые помогут улучшить предлагаемый вариант защиты в зависимости от особенностей конкретной АС.

При построении защиты АС, анализе требований к системе, определении вероятных угроз и оценке рисков, неизбежно встают вопросы: какие технологии использовать для защиты системы и как их интегрировать между собой и ней? При ответе на эти вопросы в случае использования АС на основе СУБД есть две крайности: использовать средства и методы, предлагаемые СУБД, либо использовать внешние решения. В первом случае можно достигнуть значительной экономии средств и ресурсов за

счет наличия практически готового решения (лишь бы оно удовлетворяло всем заявленным требованиям). Во втором — появляется определенная свобода выбора и действий, что порой может оказаться очень даже выгодно. Как правило, выбор падает на компромиссное решение, сочетающее использование возможностей, предлагаемых СУБД, и некоторых внешних технологий.

Рассмотрим подсистему обеспечения конфиденциальности информации в АС на базе СУБД Oracle с использованием вышеописанных криптографических возможностей СУБД. Возьмем в качестве примера АС типовой вариант на клиент-серверной архитектуре (Рис. 1).

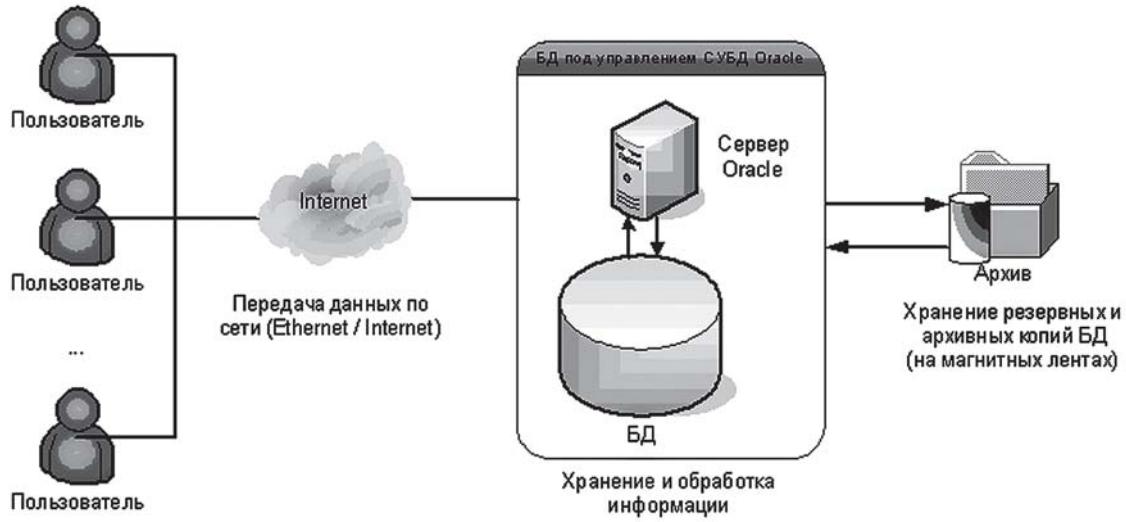

Рис. 1. Схема АС на базе СУБД Oracle

Центр обработки и хранения информации построен на основе СУБД Oracle. В его задачу входит предоставление доступа конечным пользователям к конфиденциальной информации, хранимой в БД. Конечные пользователи имеют доступ к серверу СУБД либо через внутреннюю локальную сеть, либо через Интернет. Также предусмотрено периодическое резервное и архивное копирование БД на магнитные ленты, хранящиеся в архиве.

Как видно из схемы, конфиденциальность информации, хранимой и обрабатываемой в АС, не может быть гарантирована без ряда архитектурных изменений. Сразу можно выявить следующие уязвимости АС:

1. Канал передачи данных между пользователями и сервером СУБД не защищен, данные передаются в открытом виде. Злоумышленник может прослушивать канал и читать всю передаваемую информацию.

2. Данные хранятся в БД в открытом виде, поэтому любой пользователь, обладающий достаточным набором привилегий, может читать/изменять данные других пользователей.

3. Архивные и резервные копии БД хранятся на магнитных лентах, которые потенциально не защищены от хищения.

Вышеперечисленные уязвимости АС можно устранить, построив подсистему обеспечении конфиденциальности данных на основе технологий, предлагаемых сервером СУБД Oracle. Рассмотрим детально, как это можно сделать.

Для защиты канала передачи данных между сервером СУБД и конечными пользователями необходимо шифровать сетевой трафик и желательно проводить контроль целостности передаваемой информации. Одним из вариантов такой реализации может послужить возможность настройки шифрования и контроля целостности сетевого трафика, доступная в опции СУБД Oracle Advanced Security. Для этого в первую очередь необходимо определиться со списком алгоритмов шифрования и контроля целостности, которые будут использоваться для достижения заданной цели. После несложной настройки конфигурационных файлов на серверной и клиентской сторонах можно гарантировать конфиденциальность передаваемых между ними данных.

Для защиты своих данных отдельные пользователи могут использовать функциональности выборочного шифрования данных. Каждый пользователь при необходимости может использовать различные криптографические преобразования над своими данными, чтобы гарантировать их конфиденциальность. При желании можно применить предлагаемый функциональностью криптографический интерфейс для построения более сложных криптографических схем защиты.

Использование технологии прозрачного шифрования данных позволяет, совершенно не обременяя конечных пользователей необходимостью осуществления дополнительных действий, защитить БД даже при остановленном сервере СУБД. Т. е. обеспечить шифрование файлов БД на уровне ОС, что в рассматриваемом варианте применимо для резервных и архивных копий БД. В этом случае кража магнитных лент (либо других носителей) не позволит злоумышленнику раскрыть конфиденциальные данные. Гибкость технологии ПШД делает возможным при минимуме усилий со стороны администратора по безопасности настроить шифрование отдельных таблиц и даже их столбцов, содержащих непосредственно конфиденциальные данные, при этом практически не оказывая влияния на производительность системы.

Итоговая схема АС с подсистемой обеспечения конфиденциальности данных представлена на рис. 2.

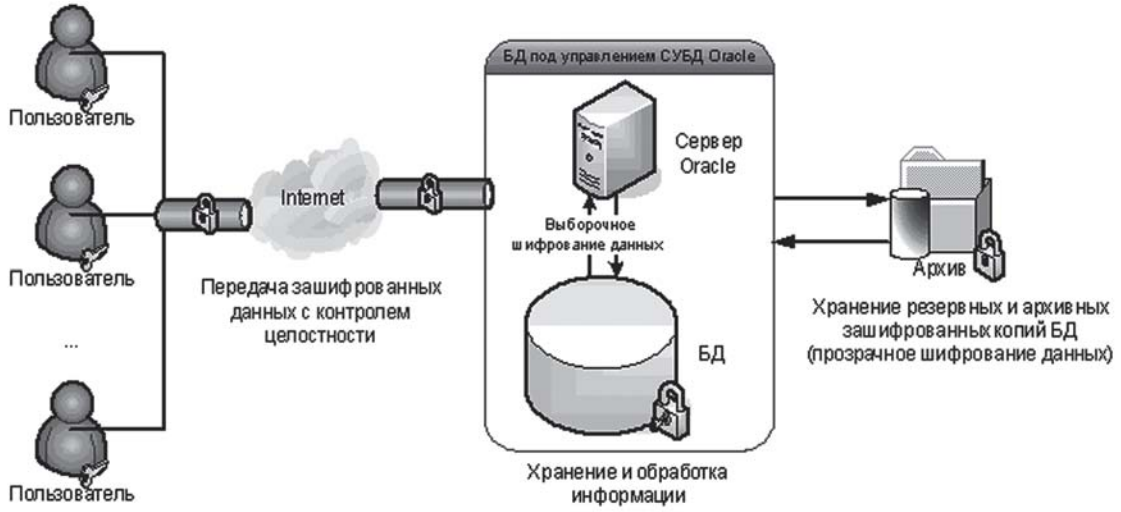

Рис. 2. Схема АС с подсистемой обеспечения конфиденциальности данных

В результате для обеспечения конфиденциальности данных, хранимых и обрабатываемых в АС, можно эффективно использовать предлагаемые СУБД Oracle технологии.

СПИСОК ЛИТЕРАТУРЫ:

1. Oracle Database Security Guide. http://www.oracle.com/pls/db102/to_pdf?partno=b14266.
2. Oracle Database Advanced Security Administrator's Guide. http://www.oracle.com/pls/db102/to_pdf?partno=b14268.