УДК 004.056.55
https://doi.org/10.47533/2024.1606-146X.02

Е. Ж. АЙТХОЖАЕВА*, Д. С. АХМЕТШӘРІПОВ

Казахский национальный исследовательский технический университет им. К.И.Сатпаева, Алматы, Казахстан

ТЕХНОЛОГИЯ ИСПОЛЬЗОВАНИЯ СПЕЦИАЛЬНЫХ ФУНКЦИЙ ШИФРОВАНИЯ В POSTGRESQL

Обсуждаются криптографические механизмы шифрования в серверной СУБД PostgreSQL с открытым исходным кодом. Анализируются специальные высокоуровневые и низкоуровневые функции симметричного шифрования модуля Pgcrypto для обеспечения информационной безопасности данных. Выполняется анализ специальных функций сторонней программы GPG, которые позволяют выполнить асимметричное шифрование в сервере баз данных PostgreSQL. Предлагается комбинированная технология шифрования данных в таблицах баз данных, использующая иерархию ключей (асимметричное и симметричное шифрование) с парольной фразой, которая также шифруется для обеспечения лучшей защиты зашифрованных данных. Выполняется апробация данной технологии на примере разработки функции для зашифрования данных, которая запускается триггером при вставке строк в таблицу базы данных, и функции расшифрования данных при чтении зашифрованных данных из таблицы.

Ключевые слова: *сервер баз данных PostgreSQL, функции шифрования и расшифрования.*

Введение. Большинство данных, в том числе персональных и других конфиденциальных данных, хранится в серверах баз данных (БД). Исходя из этого, в целях обеспечения информационной безопасности данных, хранящихся в сервере БД, необходимо использовать криптографические механизмы шифрования, которые являются эффективным средством обеспечения информационной безопасности. Разработчики серверов БД заложили в свои продукты разные возможности в плане шифрования данных. Ниже обсуждаются возможности использования шифрования в сервере БД PostgreSQL, который в последнее время все больше привлекает внимание разработчиков различного масштаба в качестве базового ПО для своих продуктов. PostgreSQL является объектно-реляционной системой управления базами данных (ORDBMS), наиболее развитой из открытых СУБД в мире. Имеет открытый исходный код и является альтернативой коммерческим серверам БД [1,2,3].

Нужно заметить, что сам подход в реализации шифрования симметричными и асимметричными ключами в PostgreSQL отличается от других серверных СУБД, так как в PostgreSQL отсутствует хранилище мастер-ключа, создание и использование созданных симметричных ключей шифрования осуществляется в едином процессе. Это справедливо и относительно асимметричных ключей шифрования. Всё это создает определенные трудности при выполнении шифрования данных и требует разработки другой технологии использования функций шифрования по сравнению с технологией, применяемой в серверах БД Oracle и MS SQL Sever, где создание и использование ключей шифрования представляют собой разные процессы [4,5,6]. Следует заметить, что использование криптографических методов в серверах БД является экспертной областью.

* E-mail корреспондирующего автора: akdauren4@gmail.com

В PostgreSQL предусмотрено использование симметричных как блочных алгоритмов шифрования с ключами разной длины (DES, 3DES, 3DES с тремя ключами, DESX, AES), так и потоковых алгоритмов шифрования (RC2, RC4), а также асимметричного алгоритма RSA с ключами разной длины. Тем не менее, из предлагаемых алгоритмов симметричного шифрования рекомендуется использовать более криптостойкий алгоритм AES с длиной ключа 256 бит, а из асимметричного шифрования рекомендуется использовать алгоритм RSA с длиной ключа 4096 бит.

Методы и результаты. Предметом исследования являются методы и существующие механизмы обеспечения криптографической безопасности данных в сервере БД PostgreSQL.

На сегодняшний день PostgreSQL является одной из самых популярных и доступных серверных СУБД с открытым исходным кодом, что по сравнению с другими серверами БД, дает возможность подстроить систему под свои нужды и требования. Безусловно, данное преимущество является плюсом, но и минусом одновременно, так как не имеет постоянной техподдержки, обновлении и модификации. Но нужно учитывать, что данный минус компенсируется за счет Community решений. Также нужно учесть, что PostgreSQL не проигрывает в плане производительности, безопасности, масштабируемости и стоимости, таким востребованным коммерческим серверам БД, как Oracle и MS SQL Sever.

Для реализации криптографических механизмов шифрования в PostgreSQL используются функции шифрования PGP криптографического модуля Pgcrypto, которые реализуют часть шифрования стандарта OpenPGP (RFC 4880) [7, 8]. Для корректной работы Pgcrypto требуется криптографическая библиотека с открытым исходным кодом OpenSSL.

Поддерживается шифрование симметричными и асимметричными ключами. Шифрование симметричным ключом реализуется на основе высокоуровневой функции шифрования Pgp_sym_encrypt (), которая принимает на вход шифруемые данные, метод шифрования и парольную фразу, на основе которой генерируется симметричный ключ для шифрования данных. Дешифрование осуществляется с помощью высокоуровневой функции Pgp_sym_decrypt () аналогично, так же на основе парольной фразы.

Помимо высокоуровневых функций шифрования, PGP дает возможность использования низкоуровневых функций шифрования, таких как encrypt () и decrypt (). Но эти функции не работают с текстовыми данными. Шифрование и дешифрование осуществляется также на основе парольной фразы. Данные функции поддерживают: алгоритмы шифрования AES длиной ключа 128, 192, 256 бит и Blowfish; режимы cbc (шифрование следующего блока зависит от предыдущего (по умолчанию)) и ecb (каждый блок шифруется отдельно (только для тестирования)) и параметры допустимого дозаполнения pkcs (данные могут быть любой длины (по умолчанию) и none (размер данных должен быть кратен размеру блока шифра) [7,8].

Для реализации асимметричного шифрования используется сторонняя программа GPG (также известная как GnuPG), которая позволяет сгенерировать публичные и приватные ключи. Шифрование реализуется на основе функции pgp_pub_encrypt (), которая шифрует данные публичным ключом и функции pgp_pub_decrypt (), которая дешифрует их приватным ключом. При этом используется парольная фраза.

Перечисленные механизмы шифрования можно применять на различных уровнях организации данных. Можно шифровать базу данных целиком, отдельные таблицы

(сущности) или же применять шифрование к отдельным столбцам (атрибутам). Применяемая обычно при этом одноуровневая или двухуровневая иерархия ключей шифрования не сможет обеспечить высокую надежность защиты от дешифрования данных.

Ниже предлагается комбинированная технология многоуровневого использования специальных функций шифрования модуля PGP и программы GPG в сервере баз данных PostgreSQL для шифрования столбцов таблицы (атрибутов сущности) при вставке новых строк в таблицу базы данных. При этом используется иерархия ключей (асимметричное и симметричное шифрование) с парольной фразой, которая также шифруется для обеспечения лучшей защиты зашифрованных данных.

Предлагается следующая технология зашифрования:

– программой GPG генерируются публичный и приватный ключи. В качестве алгоритма шифрования выбирается алгоритм RSA длиной ключа 4096 и указывается его срок годности (рисунок 1, указан срок годности 100 дней). При генерации ключей приватный ключ для большей безопасности и конфиденциальности шифруется задаваемой парольной фразой, которая передается в качестве параметра по запросу системы (рисунок 2, задана парольная фраза dauren4157@,);

```
bob@bob-IdeaPad-Gaming-3-15IMH05:~$ gpg --full-generate-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Выберите тип ключа:
   (1) RSA и RSA (по умолчанию)
   (2) DSA и Elgamal
   (3) DSA (только для подписи)
   (4) RSA (только для подписи)
  (14) Имеющийся на карте ключ
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
        0 = не ограничен
      <n>  = срок действия ключа - n дней
      <n>w = срок действия ключа - n недель
      <n>m = срок действия ключа - n месяцев
      <n>y = срок действия ключа - n лет
Срок действия ключа? (0) 100
Ключ действителен до Ср 17 Мам 2023 18:58:32 +06
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: dauren
Адрес электронной почты: my.email@mail
Примечание:
Вы выбрали следующий идентификатор пользователя:
    "dauren <my.email@mail>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? o
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
gpg: ключ 5B41FCC85589D057 помечен как абсолютно доверенный
gpg: создан каталог '/home/bob/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/home/bob/.gnupg/openpgp-revocs.d/CAEA51F6AA8CEDDA07B895055B41FCC85
589D057.rev'.
открытый и секретный ключи созданы и подписаны.

pub   rsa4096 2023-02-06 [SC] [годен до: 2023-05-17]
      CAEA51F6AA8CEDDA07B895055B41FCC85589D057
uid                      dauren <my.email@mail>
sub   rsa4096 2023-02-06 [E] [годен до: 2023-05-17]

bob@bob-IdeaPad-Gaming-3-15IMH05:~$
```

Рисунок 1 – Генерация асимметричного ключа программой GPG

– далее, для создания симметричного ключа шифрования, необходимо сгенерировать псевдослучайную парольную фразу;

– на основе сгенерированной псевдослучайной парольной фразы создается симметричный ключ, который шифрует текст с использованием высокоуровневой функции pgp_sym_encrypt() имеющей такие параметры, как идентификатор шифруемого текста, парольная фраза, алгоритм шифрования;

– созданным ранее публичным ключом асимметричного шифрования шифруется использованная парольная фраза;

– полученное значение зашифрованной парольной фразы, повторно шифруется низкоуровневой функцией шифрования encrypt (), и записывается в заранее созданную таблицу master_symkey с указанием даты, времени записи и идентификатора.

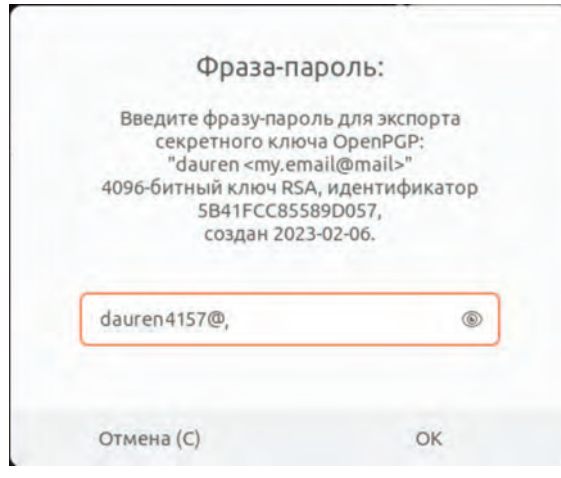

Рисунок 2 – Задание парольной фразы для асимметричного ключа

Для расшифрования данных необходимо выполнить следующие шаги:

необходимо расшифровать парольную фразу из таблицы master_symkey:

a) симметричным ключом низкоуровневой функции расшифрования decrypt ();

b) приватным ключом высокоуровневой функции pgp_pub_decrypt (), который перед этим расшифровывается;

– расшифрованную парольную фразу передать в функцию pgp_sym_decrypt () для расшифровки данных.

Сформулированные выше положения были апробированы на примере для зашифрования и расшифрования данных в таблице customers базы данных Delivery сервера PostgreSQL. Был использован язык plpgsql. Для шифрования данных была спроектирована и реализована функция зашифрования encryption. В функции encryption используется описанная выше технология зашифрования с парольной фразой “dauren4157@”. Команда Update, при замене незашифрованного текста на зашифрованный, использует функцию pgp_sym_encrypt() для получения зашифрованных данных. При записи в таблицу master_symkey зашифрованной парольной фразы в качестве идентификатора используется ID покупателя. Созданная функция encryption запускается триггером encryption_trigger типа After при выполнении инструкции Insert, то есть вставки данных в таблицу (рисунок 3).

```
create or replace function encryption() returns trigger as $$
DECLARE
    passphrase text;
    pub_enckeyvalues bytea;
    public_key text:='-----BEGIN PGP PUBLIC KEY BLOCK-----

mQGNBGPjedwBDADega40RSN04FG+apsN5+64s9d4ds7bcFfcEV6jf4E9Q6hNAYeC
8AprTbBHMTnnK/C28ypxfjHktf/FqtyYDqSytjXmhh8r+TQldgn7ZwNh+xcZZHws
s1uFFhPsofB4g5J2beUP5OdusQ2v3Ogx33m2mMQMentN4C5E92yhWefzNEbQ2yid
WslF4SYt0iF9q6VHmLDadwDJYxFePHQ70l2on4fSut5houFMns3YBknSxogd8ryO
sBy20ufhm3SNz1KSOh/emRxHvABnvq+gVAcHpf4UHFLmX7cWHM9esW1MgdY6KCtP
D9DOcBh1UC+6zNSy1WprCT70tGY+f+pe1Fpvv///sjw/Z1g9TqKtKKv3cgHlZ9LB
-----END PGP PUBLIC KEY BLOCK-----';
Begin
IF (TG_OP = 'INSERT') THEN
-- Генерация парольной фразы длинной 256 символов
select gen_key() into passphrase;
--Шифрования вставленных данных в таблице
Update customers set last_name=pgp_sym_encrypt(new.last_name, passphrase, 'compress-algo=1, cipher-algo=aes256') where last_name=new.last_name;
Update customers set first_name=pgp_sym_encrypt(new.first_name, passphrase, 'compress-algo=1, cipher-algo=aes256') where first_name=new.first_name;
Update customers set address=pgp_sym_encrypt(new.address, passphrase, 'compress-algo=1, cipher-algo=aes256') where address=new.address;
Update customers set phone_number=pgp_sym_encrypt(new.phone_number, passphrase, 'compress-algo=1, cipher-algo=aes256')  where phone_number=new.phone_number;
-- Шифрование парольной фразы публичным ключом
select pgp_pub_encrypt(passphrase, dearmor(public_key)) into pub_enckeyvalues;
--Шифрование и вставка в таблицу мастер ключа для симетричного ключа зашифрованной парольной фразы
insert into master_symkey (skey_id,key_values,generate_date)
values (new.customer_id,encrypt(pub_enckeyvalues,'Pass'::bytea,'aes'),now()::timestamp);
END IF;
RETURN NULL;
END;
 CREATE TRIGGER enctyption_trigger
 AFTER INSERT ON customers
     FOR EACH ROW EXECUTE FUNCTION encryption();
```

Рисунок 3 – Скрипт реализации функции encryption и триггера для зашифрования

На рисунке 4 показана инструкция Insert (выполняется вставка данных в таблицу customers базы данных Delivery – четыре записи), которая запускает триггер encryption_trigger, чтобы зашифровать вставляемые данные. Шифруются все данные, кроме параметра ID покупателя, который является идентификатором и необходим для идентификации как записи, так и соответствующей этой записи зашифрованной парольной фразы в таблице master_symkey. На рисунке 4 также видны этапы процесса шифрования в виде последовательности использования ключей.

На рисунке 5 показан результат работы функции encryption: в таблицу customers записались четыре зашифрованные записи (строчки таблицы), идентификатором для каждой записи является незашифрованный ID покупателя.

Рисунок 4 – Запуск триггера encryption_trigger для зашифрования данных

Рисунок 5 – Результат работы функции encryption и триггера для зашифрования

Для расшифрования зашифрованных данных была реализована функция decryption, которая принимает в качестве параметра ID покупателя, по которому выполняется обратный процесс расшифрования. Процесс расшифрования данных состоит из двух этапов.

На первом этапе, при котором расшифровывается парольная фраза, используется запрос Select с вложенным внутренним запросом Select, так как парольная фраза была зашифрована сначала публичным ключом, потом симметричным ключом. Основной запрос Select использует функцию pgp_pub_decrypt () и записывает результаты данного запроса в переменную priv_deckeyvalues. Функция pgp_pub_decrypt () в качестве первого параметра принимает результат вложенного запроса Select; в качестве второго параметра принимает приватный ключ; в качестве третьего параметра парольную фразу к приватному ключу (dauren4157@,). Вложенный запрос Select выполняет низкоуровневую функцию дешифрования decrypt (), которая в качестве первого параметра принимает запись столбца key_values из таблицы master_symkey (зашифрованную симметричным ключом парольную фразу), идентифицируемую по ID покупателя; в качестве второго параметра принимает парольную фразу для симметричного ключа (pass), которым будет проводиться расшифровка зашифрованной парольной фразы; в качестве третьего параметра принимает алгоритм расшифрования (aes).

На втором этапе, выполняется запрос Select, который использует функцию расшифрования pgp_sym_decrypt() и возвращает таблицу с расшифрованными данными (рисунок 6). Функция pgp_sym_decrypt () в качестве первого параметра принимает

Рисунок 6 – Скрипт реализации функции расшифровывания decryption

зашифрованные данные или записи столбцов из таблицы customers (last_name, first_name и т.д), идентифицируемые по ID покупателя; в качестве второго параметра принимает значения переменной priv_deckeyvalues.

На рисунке 7 показан результат работы функции decryption при выборке из таблицы записи с ID покупателя равным 1.

Рисунок 7 – Результат работы функции расшифровывания decryption

Заключение. Анализ функций зашифрования и расшифрования криптографического модуля Pgcrypto в PostgreSQL и программы GPG и апробированная предложенная технология их использования позволяет сделать вывод о хорошей функциональности PostgreSQL в области шифрования данных. Кроме представленных выше возможностей, в PostgreSQL имеются стандартные функции хеширования и специальные функции хеширования паролей, можно зашифровать пароли учетных записей (пароли пользователей хранятся в виде хешей), шифровать канал передачи данных между сервером и клиентом. Присутствует несколько методов аутентификации, управление правами доступа как на уровне строк, так и на уровне столбцов, функции получения случайных данных. До недавнего времени в PostgreSQL не было встроенного прозрачного шифрования (Trasparent Date Encryption - TDE), которое используется в Oracle и MS SQL Sever. Но в 2023 году компания EnterpriseDB реализовала TDE для PostgreSQL [9]. В настоящее время PostgreSQL можно рекомендовать использовать для работы с конфиденциальными данными, наряду с востребованными коммерческими серверами БД, такими как Oracle и MS SQL Sever.

Шифрование позволяет скрыть исходные данные и является признанным методом обеспечения защиты данных. Но следует учитывать, что процессы шифрования и расшифрования требуют ресурсов и поэтому негативно влияют на производительность системы. Применение шифрования требует определения критичности данных и решения, какие данные требуют шифрования.

ЛИТЕРАТУРА

1 Рогов Е. В. PostgreSQL 15 изнутри [Текст]. - Москва: ДМК Пресс, 2023. - 662 с.

2 Schonig H. Mastering PostgreSQL 13: Build, administer, and maintain database applications efficiently with PostgreSQL 13, 4th Edition 4th ed. Edition [Текст]. – Birmingham-Mumbai: Packt Publishing, 2020.- p.476.

3 PostgreSQL [Электронный ресурс] / The PostgreSQL Global Development Group. – Режим доступа: http://www.PostgreSQL.org, свободный. – Загл. с экрана.

4 Oracle Database Advanced Security Guide, 21c [Электронный ресурс] / Oracle. – Режим доступа: https://docs.oracle.com/en/database/oracle/oracle-database/21/asoag/advanced-security-guide.pdf

5 Документация по SQL. Шифрование SQL Server. [Электронный ресурс] - Режим доступа: https://docs.microsoft.com/ru-ru/sql/relational-databases/security/encryption/sql-server-encryption?view=sql-server-ver15, свободный. -Загл. с экрана.

6 Ахметшәріпов Д.С., Айтхожаева Е.Ж. Сравнительный анализ криптографических механизмов в серверах баз данных [Текст] / Сборник материалов Международной научно-практической конференции «Фундаментальные научные исследования: теоретические и практические аспекты». – Кемерово: ЗапСибНЦ, 2023 – с.41-44. Электронная версия на сайте https://sibscience-new.ru/images/doc_temp/2023/Sbornik_28_02_23.pdf.

7 Pgcrypto [Электронный ресурс] / The PostgreSQL Global Development Group. – Режим доступа: https://www.PostgreSQL.org/docs/current/pgcrypto.html, свободный. – Загл. с экрана.

8 Шифрование данных в PostgreSQL [Электронный ресурс]. - Режим доступа: https://hardyantz.medium.com/data-encryption-in-PostgreSQL-a4b51a60dfc2, свободный. – Загл. с экрана.

9 EnterpriseDB Postgres Plus Advanced Server [Электронный ресурс]. - Режим доступа: https://complit.kz/p37468357-enterprisedb-postgres-plus.html, свободный. -Загл. с экрана.

REFERENCES

1 Rogov, E.V. (2023). PostgreSQL 15 from the inside. Moscow: DMK Press. - 662 p.

2 Schonig, H.-J. (2020). Mastering PostgreSQL 13: Build, administer, and maintain database applications efficiently with PostgreSQL 13. 4th Edition 4th ed. Edition. Packt Publishing.

3 PostgreSQL (n.d.) The PostgreSQL Global Development Group. http://www.PostgreSQL.org.

4 Oracle. (2023). Oracle Database Advanced Security Guide, 21c. Oracle. https://docs.oracle.com/en/database/oracle/oracle-database/21/asoag/advanced-security-guide.pdf

5 Microsoft (n.d.). SQL Server Encryption. https://docs.microsoft.com/ru-ru/sql/relational-databases/security/encryption/sql-server-encryption?view=sql-server-ver15

6 Akhmetsharipov, D.S., Aitkhozhayeva, Y.Zh. (2023). Comparative analysis of cryptographic mechanisms in database servers // Collection of materials of the International Scientific and Practical Conference Fundamental scientific research: theoretical and practical aspects. – Kemerovo: ZAPSIBNTS, 2023 – pp.41-44. https://sibscience-new.ru/images/doc_temp/2023/Sbornik_28_02_23.pdf .

7 Pgcrypto. (n,d.). The PostgreSQL Global Development Group. https://www.PostgreSQL.org/docs/current/pgcrypto.html.

8 Data encryptions in PostgreSQL. (n,d.). https://hardyantz.medium.com/data-encryption-in-PostgreSQL-a4b51a60dfc2 ,

9 EnterpriseDB Postgres Plus Advanced Server. (2023). https://complit.kz/p37468357-enterprisedb-postgres-plus.html.

Е. Ж. АЙТХОЖАЕВА, Д. С. АХМЕТШӘРІПОВ

Қ.И.Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті, Алматы, Қазақстан

POSTGRESQL-ДЕ АРНАЙЫ ШИФРЛАУ ФУНКЦИЯЛАРЫН ҚОЛДАНУ ТЕХНОЛОГИЯСЫ

Ашық бастапқы кодты PostgreSQL серверлік ДҚБЖ-да криптографиялық шифрлау механизмдері талқыланады. Деректердің ақпараттық қауіпсіздігін қамтамасыз ету үшін

Pgcrypto модулінің арнайы жоғарғы және төменгі деңгейлі симметриялы шифрлау функциялары талданады. PostgreSQL дерекқор серверінде асимметриялық шифрлауға мүмкіндік беретін үшінші тарап GPG бағдарламасының арнайы функцияларына талдау жасалады. Мәліметтер базасының кестелерінде кілттердің иерархиясын (асимметриялық және симметриялық шифрлау) қолдана отырып, шифрланған деректерді жақсы қорғауды қамтамасыз ету үшін шифрланған құпия сөз тіркесімен бірге деректерді шифрлаудың біріктірілген технологиясы ұсынылады. Деректер базасының кестесіне жолдар салынған кезде триггермен іске қосылатын деректерді шифрлау функциясын және кестеден шифрланған деректерді оқу кезінде деректерді дешифрлау функциясын әзірлеу мысалында осы технологияны сынақтан өткізу жүзеге асырылады.

Түйін сөздер: *PostgreSQL дерекқор сервері, шифрлау және дешифрлау функциялары.*

Y. ZH. AITKHOZHAYEVA, D. S. AKHMETSHARIPOV

Kazakh National ResearchTechnical University named after K.I.Satbayev,
Almaty, Kazakhstan

TECHNOLOGY FOR USING SPECIAL ENCRYPTION FUNCTIONS IN POSTGRESQL

Cryptographic encryption mechanisms in the open source PostgreSQL server DBMS are discussed. Special high-level and low-level functions of symmetric encryption of the Pgcrypto module for ensuring information security of data are analyzed. The analysis of special functions of the third-party GPG program is performed, which allow performing asymmetric encryption in the PostgreSQL database server. A combined data encryption technology in database tables is proposed, using a hierarchy of keys (asymmetric and symmetric encryption) with a passphrase, which is also encrypted to ensure better protection of encrypted data. This technology is being tested on the example of developing a function for encrypting data, which is triggered by a trigger when inserting rows into a database table, and a data decryption function when reading encrypted data from a table.

Key words: *PostgreSQL database server, encryption and decryption functions.*