


## Backup Encryption and Encrypted Backup Operation Performance in SQL Server

Zühre Aydın

Energy Market Regulatory Authority, 06530 Çankaya, Ankara, Türkiye

zaydin@epdk.gov.tr 

Received date:24.02.2022, Accepted date: 30.08.2022

### Abstract

Database security features are important cases to be focused on database system development. However, using security features can affect the system's performance. Encryption database backup is a kind of security algorithm option that includes keys and certificates. It is essential for additional layer of security. This study aims to observe how Transparent Data Encryption (TDE) feature, as a solution tool for data security will impact the related database performance. In this study, there will be an overview of database backup encryption and encrypted backup operation through backup testing. Hence, the research will be conducted on backup testing for monitoring database system's performance when implementing TDE solution. The study will contribute to the literature with using backup encryption algorithms in terms of efficiency and benchmarking of encrypted backup operation performance with TDE by finding how significant performance change is happening.

**Keywords:** Backup encryption, backup operation performance, database security, extensible key management, transparent data encryption

## SQL Server'da Şifreli Yedekleme İşlemlerinde Performans

### Öz

Veritabanı güvenlik özellikleri, veritabanı sistemi geliştirmeye odaklanmak için önemli araçlardır. Ancak güvenlik özelliklerinin kullanılması sistemin performansını etkileyebilir. Şifreleme yöntemiyle gerçekleştirilen veritabanı yedeklemesi, anahtarları ve sertifikaları içeren bir tür güvenlik algoritması seçeneğidir. Ek güvenlik katmanı için gereklidir. Bu çalışma, veri güvenliği için bir çözüm aracı olan Şeffaf Veri Şifreleme (TDE) özelliğinin bulunduğu ilgili veritabanı performansını nasıl etkileyeceğini gözlemlemeyi amaçlamaktadır. Çalışmada, ilgili veritabanında gerçekleştirilen yedekleme testi ile şifreli yedekleme işlemine genel bir bakış yapılacaktır. Bu nedenle araştırma, TDE çözümünü uygularken veritabanı sisteminin performansını izlemek için yedekleme performans testleri üzerinde duracaktır. Çalışma, veritabanı yedek şifreleme algoritmalarını kullanarak, TDE ile şifrelenmiş yedekleme operasyonu performansının ne derece performans değişikliğine sebep olduğunu bularak verimlilik ve kıyaslama bulgularıyla literatüre katkıda bulunacaktır.

**Anahtar Kelimeler:** Yedek şifreleme, yedekleme işlemi performansı, veritabanı güvenliği, genişletilebilir anahtar yönetimi, şeffaf veri şifreleme

### INTRODUCTION

Data security is important to save the data from data threats such as; unauthorised database access, unauthorized transactions, backup theft, database injections, DDoS attacks, malicious attacks or phishing activities. Database security includes the layers for data level, user-level and system level. Organisations must take measures to ensure fast recovery, and always have a backup available for saving and maintaining the data. The aim of a theft

operation against databases is to gain data from related database or to accessing the instance. If they can reach to backups then they don't need to access database instance. Because stoled database can be restored onto their own instance.

Encryption in databases relates with selecting, implementing and managing encryption keys an encrypting algorithm. Public, private, and symmetric encryption keys are used to keep data safely.

SQL includes TDE, Always Encrypted (AE), Dynamic Data Masking (DDE) database security solutions within certificates and asymmetric keys that give information about using Public Key Cryptography (PKC) (Guyer et al. 2019, Alain et al. 2019).

TDE provides opportunities of real time I/O encryption and decryption of data files. The encryption uses a symmetric database encryption key (DEK). This key is secured by a certificate on master database or by an Extensible Key Management (EKM) module protected asymmetric key (Guyer et al. 2019).

A database administrator's responsibility is heavy because of gigantic amount of data, millions of rows that relates with the processes of big managements. Apart from keeping the data secure, a database administrator has to keep the system up and running. An unauthorized access to the system, could simply restore a copy of database. So, the encrypted backup solutions of related systems should be implemented. Ensuring the security of database backups are equally needed and important for the security of the database itself. Reliable and full backups are an essential part of recovery. This study outlines common database backup security encryption concepts and performance with TDE in SQL database and why it should be implemented to maintain the data security and integrity. Backup testing operations show how system performance changes when TDE enabled or disabled.

In these contexts; there are literature studies studied on data security, database level precautions and database encryption. Mukherjee (2018) studied on database encryption methodologies in SQL database and gave an overview on TDE, Cell Level Encryption (CLE) at disk level security and AE encrypts at column level security. Soni et al. (2019) stated that, backup data is essential for availability and maintainance of database. Bouchti et. Al. (2020), proposed solutions to protect the keys of different database encryption algorithms for columns level and tables level. They emphasized that a database encryption algorithm should provide a strong security. Malik et. Al. (2016) discussed possible database attacks and gave collected information of

different threats and database security. Madyatmadja et. al. (2021), performed performance testing on TDE enabled databases and found out that TDE is useful and necessary for database security and it adds a security level on the system although it shows a very little performance disadvantages on CPU usage. Shmuelia et. al. (2014) benchmarked architectures of database encryption systems and presented that these encryption architectures have a high level of security, but they show a significant impact on performance and on application layer, so they suggested a novel architecture based on encryption module within the database management system. Cherry (2015) researched potential attack vectors on SQL Server databases and how to protect databases from these attacks. Nakamura et. al. (2003), proposed a non-deterministic model for database backups with an optimal interval solution under suitable situations. Deshmukh et. al. (2011) studied on creation ways of master key, certificate that protected by the master key on SQL Server 2008. TDE feature encrypts files and logs in real time at disk level (Coleset al. 2008, Hammouchi et al.2019). This provides a solution where data can be secured by using a DEK, and this can prevent the prospect of using data without having a key and increasing the security of the database by preventing unauthorized access,. TDE provides protection to open the data contents of physical files (Guyer et al. 2019, Alain et al. 2019).

## MATERIAL AND METHODS

Backups are copies of application data that stored on backup media. Backup medias should be in a secure physical location. Securing database backup is concern of data-level, user-level and server-level security. In addition, a database admin can secure backups with regular tests, more than one copies, role based multiple-factor authentication for the database backups, 3 2 1 backup approach and backup encryption. Here, the 3 2 1 implies having 3 copies of production data 2 different media sets with at least 1 copy off site for disaster recovery. In the study, there will be use of backup encryption for reducing the risks and threads of backups in SQL Server.

**Table 1.** Trial system model details

		Trial System with TDE	Trial System without TDE
<b>Operating System</b>		Windows Server 2019	Windows Server 2019
<b>CPU</b>		Intel® Xeon® E5-2630 v3 @ 2.40GHz (2 CPUs)	Intel® Xeon® E5-2630 v3 @ 2.40GHz (2 CPUs)
<b>Memory</b>		32 GB	32 GB
<b>Database System</b>	<b>Management</b>	SQL Server 2019	SQL Server 2019

### Securing Database Backup

The study will carry out backup performance testing to monitor system’s performance within TDE. In addition; study will monitor the efficiency based on CPU usage, read/write rates and backup duration of database backup. The trial system model details is given as in Table1.

Backup testing shows how the system performance can be affected by a backup operation after implemented security measurement, TDE. The aim of backup testing is to show how and in how much time a secured or unsecured database system can restore related data when there is a data loss. Backup testing can find out how a backup can be affected by the performance of the existing transaction through repeated tests in different times. The study will conduct the tests by a backup process using compressed and uncompressed backups on trial system with TDE and on trial system without TDE. The database size that will be backed up is 41548.75 MB. We will then record the time for the backup process by displaying the time before and after the backup in milliseconds. We use SQL Server Management Tools (SSMS) to perform backup queries and gain time to perform backup activities. Thus, backup testing can help the examiner to determine the performance.

### Encryption hierarchy process and TDE

SQL Server has the capability of encrypting backups to ensure that stolen backups cannot be restored. During encrypting backups, an encryption algorithm and an encryptor should be specified for the encryption key [2]. SQL Server supports the AES 128, AES 192, AES 256, and Triple DES as encryption algorithms and a certificate or asymmetric

key as encryptors. SQL Server keeps encryption keys on a secure key manager. To encrypt a database, database admins must be sure about that master database includes a master key. Catalog views related to encryption in SQL Server are; sys.dm\_database\_encryption\_keys includes encryption key information and state of encryption, sys.symmetric\_keys, sys.asymmetric\_keys, sys.certificates and sys.database\_principals. These catalog views give information about database symmetric keys, and certificates installed in the related instance. The sys.database\_principals catalog view gives information about the principals.

SQL Server’s cryptography methodology depends on a hierarchical processes of encryption keys and related certificates on master database. Backup encryption processes include; Transact SQL (T-SQL) functions, asymmetric keys, symmetric keys, certificates and TDE. In asymmetric encryption process there are a public key for encryption and a private key for decryption. Asymmetric encryption is a methodology of Public Key Infrastructure (PKI) and Public Key Cryptography (PKC).

TDE executes encryption and decryption within the database engine itself. EKM gives to SQL Server the ability to store the encryption key for key security and management. Asymmetric and symmetric keys are stored in SQL Server EKM module.

In the SQL Server encryption hierarchy if there is no master key, creation of master key on master database should be provided by CREATE MASTER KEY statement. After that the system needs a backup of certificate kept in master database safely. This certificate will be in secure with created database master key and will be utilized to encrypt the related

database backup. The CREATE CERTIFICATE command can be used to get a certificate from a trusted Certificate Authority (CA), to generate a key pair and to create a self-signed certificate for database-level security. After creation of backup certificate, master key backup should be taken and backup certificate should be exported to a file. It is critical that you backup the master DB key and the database backup certificate to a secure location.

### **Encrypt the backup with TDE built-in security mechanisms**

After the creation of master key and certificate files, full backup encryption can be performed by

```
BACKUP DATABASE [BackupEncrypted]
TO DISK N'C:\Tmp\BackupEncrypted_FULL.bak'
WITH COMPRESSION, ENCRYPTION (
ALGORITHM = AES_256,
SERVER CERTIFICATE
=MasterCert_BackupEncrypted),
```

statement for AES\_256 algorithm.

This statement shows us encrypting a SQL Server database full backup process during by using SQL Server's built-in security mechanisms.

Differential and log backup encryption can also be performed with SET ENCRYPTION ON command by using SQL Server's built-in security mechanisms. However, inserting a data entry for testing data verification is important before the backup encryption. Also before the log and differential backup operation, verifying some of data is important.

Checking the encrypted backup SQL statement for backup media set, algorithm of the key, encryptor information, compression status and database name is essential after backup process.

## **RESULTS AND DISCUSSION**

### **Effects of TDE on backup performance**

Backup operations have many impacts on performance due to disk I/O load, CPU rates and memory usage. This study shows how TDE solution affects database backup performance with a backup operation test. The related tests performed on both TDE enabled compressed and uncompressed, TDE disabled compressed and uncompressed full backup operations. Tests results' are shown in Table 2 that includes CPU as time spent on CPU in milliseconds, reads and writes as read or written pages count during the query, duration as the total backup process time. Considering the different load processes on the server, the backup performance test implemented 3 times on SQL Server 2019 during backup operations. Table 2 and Table 3 includes average results of 3 times repeated backup tests.

Table 2 and Table 3 shows that having TDE enabled on compressed backup there is a little high usage of CPU during backup operation because each data page being read from to the disk or written to the disk must be encrypted and that affects performance of entire workload . In addition, TDE enabled backup operation duration is longer than TDE disabled depending on backup compression. Since the backed up database file has 41548.75 MB size TDE enabled uncompressed backup operation duration is the longest process. TDE enabled uncompressed backup operation has the biggest Cpu usage also. From the results, the study shows that TDE feature affects CPU usage and process duration. This is because of the backup process encrypted by the DEK that takes longer for the security of the backup file.

**Table 2.** Results of compressed backup tests

<b>Compressed Backup</b>				
	<b>Cpu (Ms)</b>	<b>Reads</b>	<b>Writes</b>	<b>Duration (Ms)</b>
<b>TDE Enabled</b>	5580	408	16	309600
<b>TDE Disabled</b>	5362	367	14	183552

**Table 3:** Results of uncompressed backup tests

<b>Uncompressed Backup</b>				
	<b>Cpu (Ms)</b>	<b>Reads</b>	<b>Writes</b>	<b>Duration (Ms)</b>
<b>TDE Enabled</b>	7995	422	19	749600
<b>TDE Disabled</b>	5520	345	14	250200

**CONCLUSION**

Data theft can be occurred by stealing backup media of an instance. The mitigation of backupmedia theft can be provided through the encryption of data and log files within TDE SQL Server solution. In TDE solution, backups can't be restored unless the certificate is available.

In the study; the backup encryption and performance impact of TDE is shown on TDE enabled and disabled database backups through. The results can be affected by database size, data compression, parallel backup operations, network and backup device quality.

This study tried to emphasize the processes, advantages and backup performance of TDE enabled databases. Since there is a very little performance decreasing on TDE eabled systems, database admins can use TDE enabled security solutions on databases safely if system hardware and network quality is sufficient.

**CONFLICT OF INTEREST**

The author report no conflict of interest relevant to this article

**RESEARCH AND PUBLICATION ETHICS STATEMENT**

The author declares that this study complies with research and publication ethics.

**REFERENCES**

Alain, N., Kibe, A., Cheruiyot and W. K., (2019). "International Journal of Scientific Engineering and Technology," Use of Enhanced Transparent Data Encryption to Protect Database Against Exposure of Backup Data, pp. 477-481.

Bouchti, K. El, Ziti, S., Omary, F. and Kharmoum, N. (2020). New Solution Implementation to Protect Encryption Keys Inside the Database Management System, *Advances in Science, Technology and Engineering Systems Journal* Vol. 5, No. 2, 87-94.

Carter, P. (2018). *Securing SQL Server: DBAs Defending the Database*, ISBN-13 (pbk): 978-1-4842-4160-8, Apress

Cherry, D. (2015). *Database Backup Security Securing SQL Server (3rd Edition) Protecting administrator Database from Attackers*, pp. 293-311.

Coles, M. and Landrum, R. (2008). *Expert SQL Server Encryption*, Springer Natur, 2011. Page 11/17

Deshmukh, A.P. A. and Qureshi, G. R. (2011). Transparent Data Encryption- Solution for Security of Database Contents, (IJACSA) *International Journal of Advanced Computer Science and Applications*, Vol. 2, No.3.

Guyer, K., V. Milener, G. and Ray, M. (2019). "Transparent Data Encryption (TDE). [Online]. Available: <https://docs.microsoft.com/enus/sql/relationaldatabases/security/encryption/transparent-data-encryption?view=sql-server-ver15>.

Research article/Araştırma makalesi  
DOI:10.29132/ijpas.1077979

- Hammouchi, H., Cherqi, O., Mezzour, Ghogho, G. M. and Koutbi, M. E. (2019). "International Symposium on Machine Learning and Big Data Analytics for Cybersecurity and Privacy," Digging Deeper into Data Breaches: An Exploratory Data Analysis of Hacking Breaches Over Time, pp. 1004-1009.
- Madyatmadja, E.D., Nur Hakim, A., D. And Sembiring, J. M., (2021). Performance Testing on Transparent Data Encryption for SQL Server's Reliability and Efficiency, Journal of Big Data.
- Malik, M. and Patel, P. T. (2016). Database security attacks and control methods. International Journal of Information 6.1/2, 175-183.
- Mukherjee, S. (2018). Popular SQL Server Database Encryption Choices, SSRG International Journal of Computer Science and Engineering (SSRG-IJCSE) – Volume X Issue.
- Nakamura, S., Qian, C. and Fukumoto, S. (2003). Optimal backup policy for database system with incremental and full backups", Mathematical and Computer Modelling, vol. 38, no. 11–13, pp. 1373-1379.
- Shmuelia, E., Vaisenberg, R. and Gudesc, E. (2014). Implementing database encryption solution design and implementation issues, Computers & Security., vol. 44, pp. 33-50.
- Soni, S. and Mathew R. (2019). Database Security: Attacks and Solutions, Proceeding of the International Conference on Computer Networks, Big Data and IoT.