FUJITSU

White paper

Reinforcing your enterprise data security strategy with Transparent Data Encryption

Every organization, irrespective of its size, relies on data and information exchange to perform certain business operations. Effective use of data also enables organizations to offer new services and engage with their customers better and in a timelier fashion.

Data is considered a valuable business asset and therefore is the target of malicious actors both inside and outside the organization. Fujitsu's implementation of Transparent Data Encryption hardens your organization's data security with minimum performance overhead, no additional storage required, and without changes to existing applications.

Content

Executive Summary

Organizations are accountable for the safety and confidentiality of its business data, client data, and employee information. There are several regulatory and compliance requirements that must be adhered to. Data breaches can have severe consequences such as downtime, expensive legal fees, and more importantly, reputational damage. It is therefore imperative that companies employ data security mechanisms and procedures to protect their data against threats.

Realizing an effective security policy

An effective data security policy leads to the protection of data from unauthorized access, use, change, disclosure, and destruction, and comprises two essential components.

Data classification

Data classification is an important aspect of the security policy. Classification is based on the level of sensitivity and impact on an organization, should that data be accessed, modified or deleted without authorization. This classification helps determine what baseline security controls are appropriate for safeguarding that data. These levels vary between organizations, depending on the nature of the business. Classifying data as confidential, private or public is one such example.

- Confidential data: Leakage of data classified as confidential can cause a significant level of risk to the organization or its partners. The highest level of security controls should be applied to such data.
- Private data: Unauthorized access of data that is classified as private could result in a moderate level of risk to the organization or its partners. A reasonable level of security controls should be applied.
- Public data: Generally, this is public information. While little or no control is required to protect the confidentiality of public data, care should be taken to prevent its unauthorized modification.

This classification is never static and needs to be constantly assessed over the life cycle of the data itself.

Layered security framework

The second component of data security generally follows a layered approach in protecting sensitive data from intruders, often referred to as defense in depth.

The layers include human, physical, network, application, data and other detection technologies, deployed in such a way that a breach in one layer does not compromise the entire system of data protection.

The level of security and costs is often determined by the value of the data, which in turn determines the classification of data. The tools and techniques deployed are further challenged, as hackers become more sophisticated in their attacks, bypassing security measures.

The implementation of the security policy, can be viewed as securing the data in its three digital states:

- Data at Rest
- Data in Motion
- Data in Use

In this white paper, we will specifically discuss technologies that enables securing data at rest.

Securing data at rest

Data at rest refers to persistent data that is stored in any digital form – *i.e.*, files, spreadsheets, databases, etc. In order to prevent this data from being accessed, modified or stolen, organizations use different security protection measures.

There are many ways to protect data, and some of them include strong user authentication, role-based access control, multi-factor authentication, data encryption or a combination of methods. There are also dynamic monitoring tools used to detect and prevent intrusion, based on rules, patterns and policies.

It is crucial for organizations to know where their sensitive data resides and deploy a combination of techniques that best match their needs. Following are some of the key enabling technologies for securing data at rest.

Passwords

Passwords are the most widely used method to prevent unauthorized access to systems, applications, files, and data. Having a good password policy is essential to keeping computer systems secure. There are several other methods of authentication as well, including biometric, multi-factor authentication and token-based access control.

Role-based Access Control

Role-Based Access Control (RBAC) is based on the premise that users do not have discretionary access to enterprise objects. Instead, access permissions are associated with roles. Users are made members of roles as determined by their responsibilities, which determines access permissions. The benefit of RBAC is that users can be easily reassigned from one role to another without modifying the underlying access structure.

Encryption

Encryption offers protection by scrambling data, so only the owner of the key or password can read the data. This protects the confidentiality of the data so that if an unauthorized person gains access to the storage device or service, they will not be to obtain any information. It also protects the integrity of the data so that it cannot be tampered with without the owner knowing it. There are several types of encryption:

- Full-disk encryption (FDE) is the encryption of all data on a disk drive, including the program that encrypts the bootable OS partition. FDE prevents unauthorized drive and data access. Some disk encryption solutions have support for a Trusted Platform Module (TPM). These implementations can wrap the decryption key using the TPM, thus tying the hard disk drive (HDD) to a particular device, so the disk cannot be used elsewhere.
- Filesystem-level encryption, often called file-based encryption or file/folder encryption, is a form of disk encryption where individual files or directories are encrypted by the file system itself. Types of filesystem-level encryption include:
  - Cryptographic filesystems layered on top of the main file system
  - Encrypted general purpose filesystems

- Database Encryption

  Database administrators and database users must understand the sensitivity or classification associated with a database and its contents to ensure that sufficient security controls are applied. In cases where all of a database's contents are of the same sensitivity or classification, an organization may choose to classify the entire database at this level. Alternatively, in cases where a database's contents are of varying sensitivity or classification levels, and database users have differing levels of access to such information, an organization may choose to apply classification at a more granular level within the database.

  Limiting a database user's ability to access, insert, modify or remove content based on their responsibilities ensures that the need-to-know principle is applied, and the likelihood of unauthorized modifications is reduced.

  Since its inception, encryption has long been held as one of the top data protection techniques available. This security approach enables the user to scramble the content of protected systems using keys and utilize a decryption key to decipher it.

Transparent Data Encryption

This is a database encryption technology that solves the problem of encrypting data at rest. It is an integral element in the data security continuum. The term *transparent* denotes the fact that the encryption method is transparent to authorized users of the database, as no change is required in the applications or existing access policies.

At a high level, the encryption method protects the data in the database by encrypting the underlying files. So, no meaningful information can be obtained if data is accessed through unauthorized access to the disk or the database. In order to access the data, the original encryption certificate and key are required.

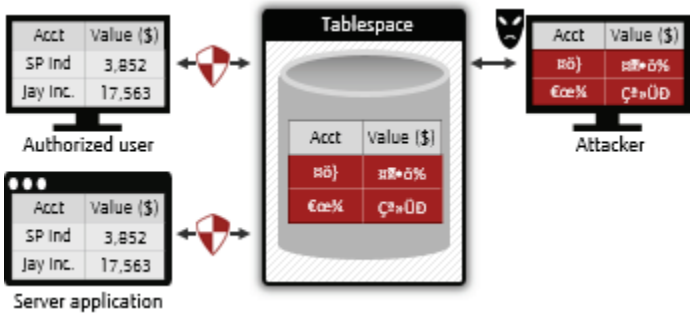

Figure 1 - Even if a data breach occurs, data is not compromised

Minimum overhead allows you to protect more data without impacting performance

Encryption/decryption is performed by manipulating entire blocks, instead of one bit at a time, which increases its performance and results in minimum overhead to the process. Overhead can be further minimized by using AES-NI built into processors that provide this feature, as is the case with several Intel and AMD processors.

As a result, you no longer need to reduce the scope of encryption to ensure application performance, and you can encrypt all data of an application with minimum impact.

FUJITSU Enterprise Postgres and Transparent Data Encryption

FUJITSU Enterprise Postgres is bundled with Transparent Data Encryption out of the box. Some of the key features are:

- Existing applications require no change, as data is transparently encrypted when it is written to the disk and decrypted when it is read from the disk
- Fast encryption/decryption, with minimum overhead
- Unlike other commercial databases, additional licenses are not required to use this functionality
- No overhead in storage areas, as the encryption algorithm does not alter the size of the object being encrypted.
- It is possible to encrypt a subset of the data as per the organization's data classification policy, so that stringent rules can be applied to that portion of the data.
- Multiple encryption keys can be deployed, which in turn are encrypted by the master encryption key. The master encryption key is also encrypted based on a passphrase.
- Encryption is extended to logs, backups, temporary tables, and temporary indexes, providing comprehensive security
- Support for streaming replication, as objects encrypted on the primary server are transferred in its encrypted format to the standby server.

Tablespace granularity

Encryption is applied at the tablespace level –all tables, indexes, temporary tables, and temporary indexes created in the specified tablespace will be automatically encrypted. This allows you to encrypt important data, but also maintain metadata and other reference data in a non-encrypted tablespace, to avoid the overhead (though minimal, as mentioned above) of encrypting and decrypting.

Following is a schematic representation of the concept:

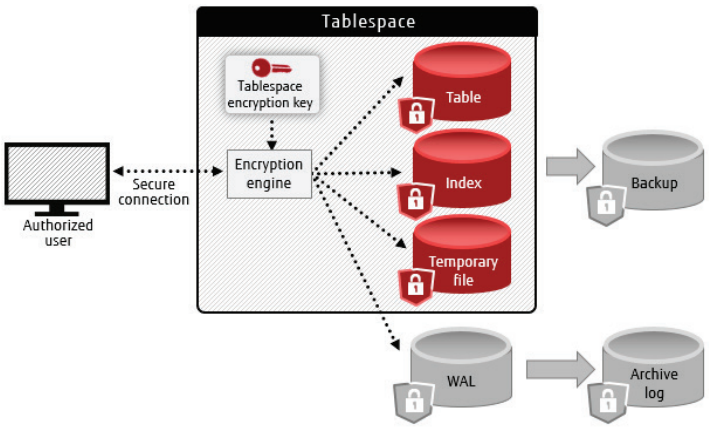

Figure 2 - Indexes on row-oriented data (reading data)

Simple to implement

There are 3 key steps in setting up Transparent Data Encryption:

1. Create a master encryption key.
   A. In `postgresql.conf`, specify the location of the encryption key.

```
keystore_location='/disk1/keystoreloc'
```

   B. Create the master key using a pass phrase.

```
SELECT pgx_set_master_key('mysecretkey');
```

   This creates a file called `keystore.ks` in the location specified in `keystore_location`.

2. Create an encrypted tablespace.
   A. Specify the encryption type desired– 128-bit or 256-bit Advanced Encryption Standard.

```
SET tablespace_encryption_algorithm = 'AES256';
```

   B. Restart the database server, opening the keystore.

```
pg_ctl --keystore-passphrase restart -D /home/db
```

   C. Create a tablespace, specifying its physical location.

```
CREATE TABLESPACE tbspc1 LOCATION '/data/encTbs1';
```

3. Create data files in the encrypted tablespace.

   Data files for tables and indexes in this tablespace will be encrypted. Even existing unencrypted tables and indexes will be automatically encrypted when moved to this tablespace.

Underlying data files will be safe at this point

You can check how data at rest is safe in encrypted tablespaces.

In the command line, gather the relevant database information.

```
SELECT oid FROM pb_tablespace WHERE spcname = tblspc;      A
SELECT oid FROM pg_database WHERE datname = d;             B
SELECT relfilenode FROM pg_class WHERE relname = table;    C
```

Display the content of the data file.

```
cat PGDATA/pg_tblspc/tblspcOid/PGvers/dbOid/tblNnode;
     A                B          C
```

Unlike with unencrypted data files, the content of the file above is garbled and will not expose any meaningful data.

Conclusion

Transparent Data Encryption should be an integral part of your organization's data security policy, as it protects all the data at rest. It provides the ability to comply with many laws, regulations, and guidelines as required in different industries. It also enables developers to secure their data using secure encryption algorithms without changing their applications.

Read more

For more information on FUJITSU Enterprise Postgres capabilities to realize a thorough data security strategy, we recommend our following resources:

- Data Masking white paper
- Dedicated Audit Log white paper
- High Availability white paper
- FUJITSU Enterprise Postgres and Enterprise Data Security page

Contact us

If you have any questions about the Transparent Data Encryption and other enterprise security features of FUJITSU Enterprise Postgres, please contact us at postgresql@fast.au.fujitsu.com.

About Fujitsu

Fujitsu is the 5th largest IT service provider in the world, offering a full range of technology products, solutions and services. Around 160,000 Fujitsu employees support customers in over 100 countries.

Contact

Fujitsu Australia Software Technology Pty Ltd
Address: 14 Rodborough Rd
Frenchs Forest NSW 2086
Australia

Email: postgresql@fast.au.fujitsu.com

Website: fast.fujitsu.com

2021-03-22 WW EN