3D Encryption

Whitepaper

1. Overview

At the latest since the Schrems II ruling, the public has been sensitized to the fact that unprotected storage of data in a cloud - especially one that is not hosted on an European server - can jeopardize the confidentiality and integrity of the information. Nevertheless, nowadays no company wants to do without the enormous advantages offered by using a cloud storage service. Recognizing this dilemma between the need for cloud storage and ensuring data protection, new technologies have emerged to address these concerns – one specifically to mention: Confidential Computing.
Confidential Computing represents a breakthrough advancement in data security. It enables environments - whether container, application, or virtual machines - to run in a fully encrypted form. This means that throughout the entire operational cycle, from startup to termination, these environments remain encrypted. Data and program flows are cryptographically isolated from the rest of the system thanks to this runtime encryption. Only the CPU - and no other components or processes - can decrypt this encrypted environment, execute instructions, and then store results in encrypted form again.

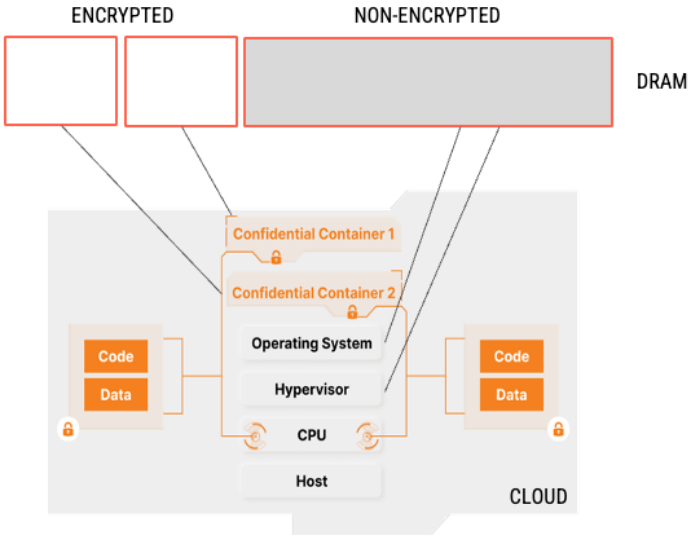
2. 3D encryption

While Confidential Computing provides an overview of all-encompassing encryption, the term "3D Encryption" goes a step further. It describes the holistic encryption of data, regardless of what state it is in.
enclaive's technology ensures that data in the cloud is encrypted at all times through 3D Encryption:

- *During use* ("data in use encryption"): When data is actively being processed.
- *At* rest ("data at rest encryption"): When data is kept in storage systems.
- *During transmission* ("data in transit"): When data is transferred between systems or across networks.

In summary, 3D Encryption ensures that data is always encrypted, regardless of its state. enclaive's offering can be figuratively viewed as a cryptographic vault where data can not only be securely stored, but also processed.

2.1 Encryption during use



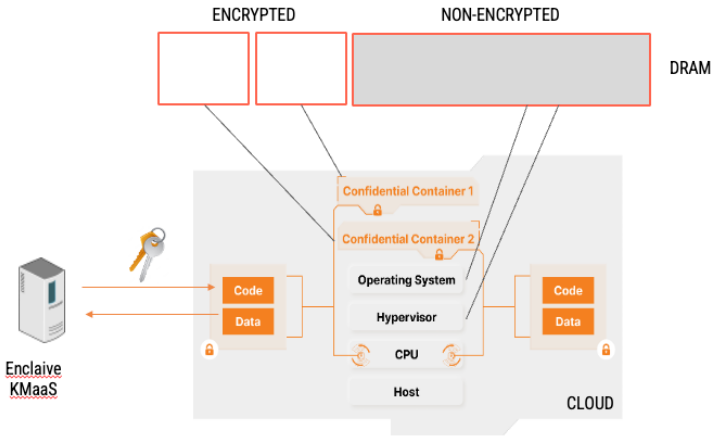
Modern processors have a security module (Intel SGX/TDX or AMD SEV) that adds memory encryption capabilities to the processor (CPU). When a program is started, leveraging enclaive technology, the following steps happen:

1. The program is decoded and so-called microinstructions are loaded into the main memory (DRAM).
2. A special memory area is addressed, the so-called enclave, of which the processor knows, the memory area is to be kept particularly secret.
3. The CPU generates a key, called the ephemeral enclave key (EEK), for each program and encrypts the microinstructions before writing them to memory.
4. The key is stored in a specially secured register on the processor.
5. If the CPU is now to execute the program, it loads the encrypted microinstructions step by step from memory, decrypts them with the ephemeral enclave key, executes them, encrypts the result again with the ephemera enclave key, and writes the ciphertext to memory.

Through this approach, enclaive can ensure the following trust model:

- The ephemeral enclave key EEK is generated anew for each enclave.
- Neither the cloud service provider, enclaive, nor the customer can access the key.
- Only the CPU can access the key.
- If the server is restarted, all enclaves and EEKs will be deleted.

2.2 Encryption at rest



Encryption at rest is a complementary enclaive technology to encrypt data this time not in main memory but on a persistent storage like a hard disk. The principle is similar to hard disk encryption for desktop PCs, with enclaive providing key management as a service[1] (KMaaS):

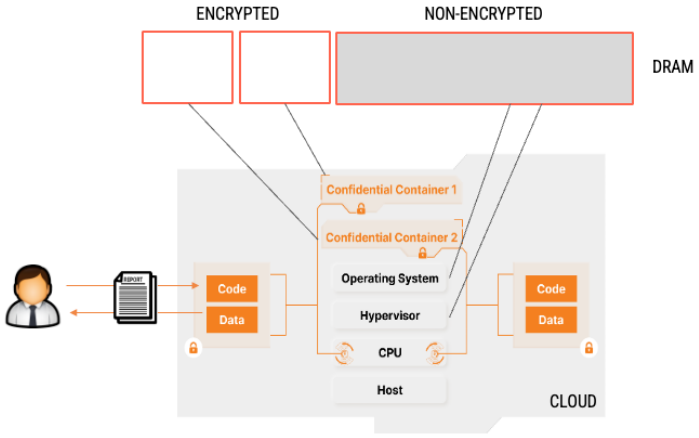
1. After the program is executed in the enclave, it will request access to the hard disk.
2. The files are encrypted with a key called the disc encryption key (DEK).
3. The enclave asks the enclaive Key Management Service (KMaaS) whether it is authorized to access the DEK.
4. Via Remote Attestation TLS (RA-TLS), a special extension of the TLS protocol, the KMaaS checks whether the enclave can get the authorization. The KMaaS checks whether (a) the enclave is actually encrypted, and (b) whether it is the correct and unmodified program.
5. The KMaaS sends the DEK via the encrypted TLS connection if the attestation is correct.
6. The enclave uses the DEK to decrypt or encrypt the encrypted file after the file has been modified.

Trust model

[1] Also, the customer can take over the key management.

- Within the enclave or the TLS channel, the DEK is at no time unencrypted.
- The cloud service provider cannot decrypt the data even if it has access to the hard disk. because it does not have access to the DEK. Consequently, he sees only randomized cipher rate.
- The KMaaS and thus enclaive cannot decrypt the data because enclaive has no access to the cloud.

2.3 Encryption in Transit



Communication between the user and the enclave takes place via the TLS protocol. An encrypted and authentic connection is created with the enclave.

Trust Model:

- The enclave endpoint is authenticated by the USERTRUST RSA Certification Authority, which verifies that the user is the rightful owner of the registered domain.

3. Cryptography

3.1 Algorithms

- Encryption in transit: AES256-GCM
- Encryption at rest: AES256-XTS
- Encryption during use: TLS 1.2

3.2 Compatibility with BSI TR-021202

The algorithms, protocols, and key lengths comply with the current recommendations of the BSI according to BSI TR-021202.[2]

	AES	TLS
BSI TR 02102-1	at least 128bit	at least 3000 bit (RSA method)
enclaive	256 bit	4096 bit

[2] https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-undZertifizierung/Technische-Richtlinien/TR-nach-Thema-sortiert/tr02102/tr02102_node.html

3.3 Updating cryptographic procedures

Over time, the requirements for the security of the cryptographic processes used could change due to new findings in cryptographic research or the breakthrough of quantum computers. The question is not whether there will be a leap in knowledge, but merely when the time will come. For example, the BSI regularly revises its recommendations, whereby changes in the choice of key length can be expected in a 5-10-year horizon.

Should the BSI suggest changes, only the key length or algorithm needs to be exchanged. The implementations used by enclaive are based on open-source standards, so updates can be implemented easily.

4. About enclaive

enclaive enables businesses to securely protect their sensitive data and applications in untrusted cloud environments by leveraging the use of Confidential Computing. Its comprehensive, multi-cloud operating system allows for Zero Trust security by encrypting data in use and shielding applications from both the infrastructure and solution providers.
With enclaive, businesses can confidently build, test, and deploy a wide range of cloud applications, all while maintaining complete control over their confidential information. enclaive's goal is to provide a universal, cloud-independent technology for enclaving sophisticated multi-cloud applications, that can be deployed with confidence and ease.

5. Learn more:

- **Youtube:** youtube.com/@confidentialcompute
- **Github**: github.io/enclaive
- **LinkedIn**: linkedin.com/company/enclaive
- **Website**: https://enclaive.cloud/

Contact information:

contact@enclaive.io
+49 30233292973
Chausseestr. 40, 10115 Berlin, Germany

Version of October 2023