Password-Hardened Encryption Services – Whitepaper

August 9, 2019

1 Introduction

Cyberattacks against databases are happening every day and more and more data is stolen. Gemalto estimates that in 2018 about 944 data breaches led to 3.3 billion data records being compromised worldwide, which translates to 18 million stolen records every day.

Criminals bypass the security measures and steal the databases. Simply encrypting these databases does not solve the problem. This is because, to access the database, the decryption key must be readily accessible by the system. Therefore criminals, who gained access to the system, can steal the decryption key as well and gain full access to all data.

Current best practice in industry considers this problem as unsolvable. Instead, it is recommended to use firewalls, separate servers and web servers, locally encrypt the databases, and run security audits, in the hope that decryption keys would not be stolen. Despite having these security measures, reality shows that it is a matter of time before criminals slip through the cracks.

2 Password-Hardened Encryption (PHE)

2.1 A Paradigm Shift

Password-hardened encryption (PHE) services solve the seemingly unsolvable by introducing a paradigm shift:

- Accept the fact that no localized security mechanism cannot be defeated
- Seek help from external server(s), so that compromising both the local and the external servers is required to steal data

2.2 Security Guarantees

In more detail, PHE is a cryptographic solution to strengthen the security of any application which employs password-based authentication and stores user-specific data in a database. A PHE scheme consists of protocols run between a (local) server storing all (encrypted) data, and an (external) rate-limiter responsible for only cryptographic operations. PHE provides the following security guarantees:

- Cryptographic security against localized attacks: a fully compromised server or a fully compromised rate-limiter alone is unable to decrypt any data

- A defense mechanism for detecting and rate-limiting online attacks, *e.g.*, from an attacker trying to guess the password of a user through the login interface, or from a fully compromised server trying to decrypt user data through the PHE interface
- A key rotation mechanism for recovering from a localized attack: Once an attack is discovered, the server and the rate-limiter can refresh their secret keys, so that old and potentially stolen secret keys are rendered useless

2.3 Efficiency Guarantees

PHE schemes are designed with efficiency and compatibility in mind:

- Achieve high-performance by minimizing the number of "public-key" operations
- Allow gradual adoption from traditional solutions based on salted-hash to PHE
- No impact to end-user experience: adoption of PHE is done in the back-end only

3 Workflow Overview

The typical workflow of a PHE scheme consists of four parts: system setup, end user registration, end user login, and key-rotation.

3.1 System Setup

The server and the rate-limiter (individually or jointly) generate their respective secret keys (and potentially the corresponding public keys).

3.2 End User Registration

1. A potential end user requests registration by providing a username and a password to the server.
2. The server and the rate-limiter jointly create a user-specific secret key and a PHE ciphertext encrypting the key. In the process, the rate-limiter learns nothing about the user password and the user-specific secret key.
3. The server uses the user-specific secret key to secure the data specific to the registering end user, *e.g.*, encrypting it with AES using the user-specific secret key.
4. The server stores the PHE ciphertext and the encrypted data in the database in an entry indexed by the username.
5. The server securely deletes the user-specific secret key.
6. The server and the rate-limiter securely delete all intermediate values generated during the joint ciphertext creation.

3.3 End User Login

1. An end user logs in by providing a username and a candidate password to the server.
2. The server retrieves the ciphertexts from the database entry indexed by the username.
3. The server and the rate-limiter jointly decrypt the PHE ciphertext. The protocol is designed so that the rate-limiter can detect if any two login sessions belong to the same end-user. Using this feature, the rate-limiter can rate-limit the login attempts of individual users. As in registration, the rate-limiter learns nothing about the user password and the user-specific secret key.
4. If the joint decryption is successful, *i.e.*, the login attempt is not rate-limited, the server obtains the user-specific secret key from the decryption result. It then decrypts the data ciphertext using the user-specific secret key to obtain the data of connecting end user.
5. The server securely deletes the user-specific secret key.
6. The server and the rate-limiter securely delete all intermediate values generated during the joint decryption.

3.4 Key-Rotation

1. If the server suspects that the database and / or the server secret key is compromised, it informs the rate-limiter and requests a key-rotation.
2. If the rate-limiter suspects that the rate-limiter secret key is compromised, it informs the server and requests a key-rotation.
3. The server and the rate-limiter jointly generate their new secret keys. The server additionally obtains an update token.
4. The server uses the update token to update each database entry so that it is compatible with the new secret keys. Note that this step is done locally by the server. No further interaction with the rate-limiter or any end users is required.
5. The server and the rate-limiter securely delete the old secret keys and all intermediate values generated during the key-rotation.

4 Technical Paper

The state-of-the-art of PHE is the scheme proposed by Lai et al. [1]. The technical paper and its presentation can be found online at `https://www.usenix.org/conference/usenixsecurity18/presentation/lai`.

References

[1] Russell W F Lai, Christoph Egger, Manuel Reinert, Sherman S M Chow, Matteo Maffei, and Dominique Schröder. Simple Password-Hardened Encryption Services. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1405–1421, Baltimore, MD, 2018. {USENIX} Association.