ownCloud's Data Encryption Model

Data protection is a critical requirement for all enterprise-grade file sync and share solutions. It is the foundation upon which trusted parties share information. As described in this paper, ownCloud provides robust server-side encryption for data at rest. ownCloud's open architecture also integrates with toolkits such as OpenSSL to protect in-flight data, and can be easily extended to support other advanced security requirements such as client-side encryption.

Core ownCloud Encryption Model

Referring to Figure 1, ownCloud's server-side encryption application performs the following functions:

- ownCloud automatically generates a 4096-bit strong private/public key-pair for each user. Private keys are encrypted with the user's login password using AES-256 cryptography.

Note: users sometimes forget their passwords. ownCloud allows administrators to optionally enable a recovery key feature that can be used to restore data access in the case of a lost password. The recovery key feature is enabled centrally, after which each user can choose whether to enable it for their ownCloud account.

- When a new file is added or sync'd, ownCloud generates an associated file-key and uses it to encrypt the file using AES-256. Thus, every file known to ownCloud has a unique file-key.

- ownCloud also encrypts the file-key itself using the public key of each user who can access the file. The result of this encryption is one or more share-keys. Each user has a unique share-key for every file s/he can access.

- When an authorized user asks to access a file, ownCloud decrypts the file-key with a combination of the user's private key and the appropriate share-key, then uses the file-key to decrypt the physical data file.

- When a file is subsequently shared with a new user, the file-key is again encrypted with the new user's public key to create a new share-key. Although this abstraction requires ownCloud to re-encrypt the file-key, it eliminates the much more expensive task of re-encrypting the entire physical file when the file is shared with new users. The same benefit is achieved when revoking a user's access to one or more files.

Advantages of ownCloud's Encryption Model

- It is highly secure – it has been implemented using proven, broadly adopted technologies like OpenSSL and standards such as AES-256 that are endorsed by organizations such as NIST.

- It is optimized to perform well even when an organization has many users and very large files.

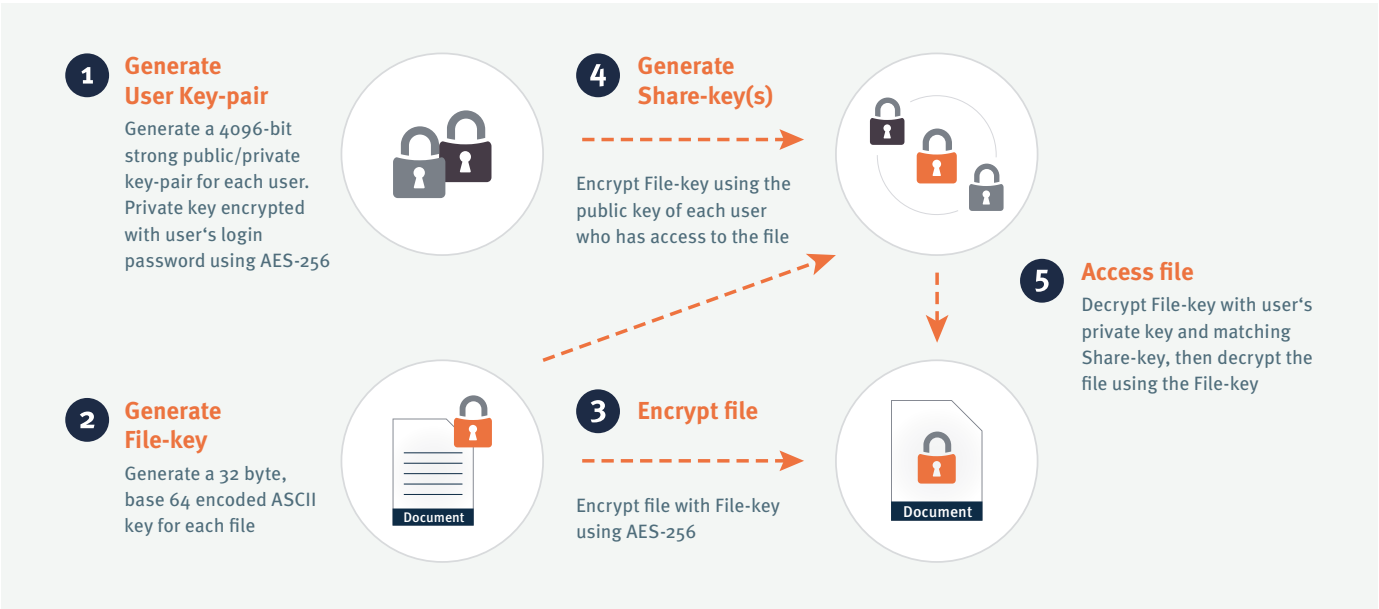

Figure 1: ownCloud's Server Side Encryption Functions

- Files can be stored securely on any ownCloud-accessible storage share in any supported format, and they can be stored externally without exposing data to 3rd parties.

- Unlike cloud-only FSS vendors, ownCloud administrators maintain complete control over their encryption keys.

- ownCloud's open model is flexible to accommodate future needs. Future expansion into client-side encryption will not require a significant overhaul if/when those needs arise.

Summary

ownCloud's data encryption model combines proven server-side encryption for data at rest with an architecture that can be easily extended to support other advanced security requirements. Based on a proven, broadly adopted foundation, ownCloud offers data protection across a variety of storage formats without putting data at risk. Importantly, ownCloud's encryption model is highly scalable and allows administrators to maintain complete control over their encryption keys.

Secure, fast, scalable and flexible – ownCloud offers peace of mind to organizations that need to meet a broad range of file sharing objectives.

Fore more information also check out the "Optimizing ownCloud Security" Whitepaper at https://owncloud.com/whitepapers

Technical References

1. Generate private/public key pairs for each user: http://www.php.net/manual/en/function.openssl-pkey-new.php

2. Generate base64 file key: http://www.php.net/openssl_random_pseudo_bytes

3. Encrypt file with file-key: http://php.net/manual/en/function.openssl-encrypt.php

4. Encrypt file-key with the public key from all users with access to the file, creating share-key: http://php.net/manual/en/function.openssl-seal.php

5. Decrypt the file-key: http://php.net/manual/en/function.openssl-open.php

6. Use the file-key to decrypt the file: http://php.net/manual/en/function.openssl-decrypt.php

ownCloud, Inc.
57 Bedford Street
Suite 102
Lexington, MA 02420
United States

www.owncloud.com/contact
phone: +1 (877) 394-2030

www.owncloud.com

@ownCloud
facebook.com/owncloud
gplus.is/owncloud
linkedin.com/company/owncloud