JOURNAL OF
EMERGING INVESTIGATORS

Article

LawCrypt: Secret Sharing for Attorney-Client Data in a Multi-Provider Cloud Architecture

Joseph Zhang[1] and David Taylor[1]

[1] Lynbrook High School, San Jose, California

SUMMARY

The accelerated employment of cloud computing among law firms is due to multiple benefits, including wide accessibility and inter-organizational information sharing. Nevertheless, the project's preliminary case study revealed lawyers' mass neglect of standard precautionary measures and, consequently, a high proportion of security breaches, putting confidential and important documents at risk. The goal of the multi-provider cloud secret sharing architecture was to ensure confidentiality, availability, and integrity of attorney documents while maintaining greater efficiency than traditional encryption algorithms. After an exhaustive development phase involving considerable testing and optimizations, software assessments of the architecture indicate the low computational overhead of adding the secret-sharing approach to a multi-provider law firm sharing environment. The efficient combination of constructions satisfies the engineering criteria as ChaCha20-Poly1305 warrants authenticity and privacy, and secret sharing ensures availability and perfect privacy. Compared with AES in CFB mode, widely used for encrypting data in the Cloud today, the secret sharing implementation boasts almost a 40% improvement over all file sizes. The only possible way of compromising this system is if multiple cloud providers collude, which is still unlikely given that the documents are additionally encrypted.

INTRODUCTION

Law firms are rapidly adopting cloud computing due to the increased convenience in sharing legal information among business partners (1). Specific benefits include the wide accessibility and inter-organizational sharing of client documents, which are all stored in one centralized cloud server. Due to this ubiquitous access, improved and less costly legal services are created, opening new legal business models (2). However, this new cloud computing paradigm implies severe security and privacy risks, raising widespread concerns about the privacy of information from third-parties, including the cloud providers (3-4). Furthermore, law firms have access to important and sensitive information, often documents and communications vital to businesses.

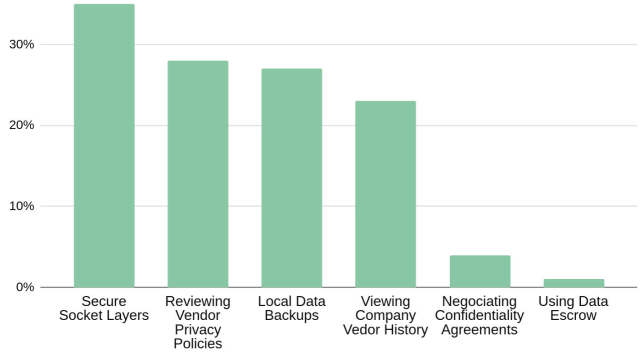

Figure 1. Most Common Security Measures in Law Firms. The survey from the American Bar Association examined the use of cloud computing across law firms of all sizes in America. The security measures listed in the figure are commonly expected when protecting sensitive legal information.

Using the 2019 Legal Technology Survey from the American Bar Association (4), we evaluated the current security of cloud computing use in the legal industry (**Figure 1**). Statistics from the survey showed an alarming neglect in standard precautionary measures in securing cloud data despite the considerable concern from lawyers regarding confidentiality and control over information (4). For instance, only 35% of law firms used Secure Socket Layers for securing communications.

In our literature review, the keyword search of the scope of cloud computing within the legal industry revealed 1440 publications, with only 10 regarding security and privacy issues (**Table 1**). The current mechanisms in place concern novel encryption algorithms, traitor tracing, and improving access control.

The following includes background information on key constructions incorporated in the project. Data stored in the Cloud is commonly encrypted with the Advanced Encryption Standard with 256-bit keys in Cipher Feedback Mode (AES256-CFB) to ensure data confidentiality. However, AES in Galois Counter Mode (AES256-GCM) warrants message authenticity in addition to confidentiality as it is an authenticated encryption with associated data (AEAD)

Table 1. Results of the Keyword Search of Cloud Computing in the Legal Industry.

Database	Pro-Quest	EBSCO-host	JSTOR	Science Direct	Total
Hits	532	60	466	382	1440
Relevant Hits	3	6	0	1	10

construction. As a more recent development in applied cryptography, ChaCha20-Poly1305 is another AEAD construction with a 256-bit key that is a viable alternative to AES256-GCM. Poly1305 most notably excels on consumer hardware with no dedicated central processing unit (CPU) instructions.

None of the papers in the literature review consider probing cloud providers who often reserve the right to monitor client information. Although most cloud providers encrypt their data, if decryption keys become compromised or the cloud provider has improper encryption implementations, sensitive documents in their entirety are revealed. However, in the secret sharing scheme, splitting a document into fragments, or shares, and distributing them privately to parallel cloud providers ensures that the encrypted legal information can only be compromised if cloud providers collude. Creating shares from a document is a matter of efficiently retaining the full information with all the shares, but mathematically establishing that a small number of shares will not contain useful information. Thus, each of the untrusted cloud providers only have access to partial, encrypted information which can be combined and decrypted to reconstruct the original document.

Derived from the case study and publications found in the literature review, our engineering goal was to create a secret sharing architecture for legal information that would ensure simultaneous authenticity of client documents during storage, availability of information in the Cloud, confidentiality of the content of documents against external parties, and un-linkability between documents and clients. We hypothesized that secret sharing, when protected by an authentication construction, would be more secure and efficient than traditional cloud encryption methods since client documents are shared as private fragments with parallel, independent cloud providers.

RESULTS

Authentication Construction Evaluation

After assessing multiple secret-key encryption algorithms, we chose to compare two authentication constructions since both satisfy the criteria of ensuring integrity and privacy. We evaluated AES256-GCM against ChaCha20-Poly1305, and we found that the latter is significantly more efficient (**Table 2**) and sufficiently secure due to its use of a 256-bit key. This is because ChaCha20-Poly1305 uses Addition-Rotation-XOR (ARX) instructions, which are CPU friendly, while AES is usually performed with dedicated instructions, not suitable on many mobile devices and some consumer hardware law firms may have. Additionally, the implementation of AES256-GCM is vulnerable to cache-timing attacks which may have influenced Google's Transportation Layer Security (TLS) and OpenSSH to switch to Poly1305 recently (7). Therefore, before fragmenting the document into shares, the document is encrypted through ChaCha20-Poly1305.

Table 2. Evaluation of the two AEAD constructions.

Cipher Measured	40 bytes	576 bytes	1500 bytes	Internet Mix[1]
AES256-GCM	43.47	34.96	35.02	35.57
ChaCha20-Poly1305	37.84	14.13	14.65	15.95

Units in cycles per byte.

[1] Internet Mix denotes the typical volume of traffic passing through Internet routers and switches in real-world conditions. Thus, Internet Mix test profiles are a distribution, meaning that the size of payloads will vary throughout the test, to represent the pattern of realistic network traffic.

Secret Sharing Assessment

Secret sharing is often defined as a threshold scheme (5-6), where only a subpopulation of t or more out of the total number of participants *n* can derive the original secret (**Figure 2**). Each participant is given some partial information called a share *D*. The shares are distributed so that no participant knows the share given to another participant. In this case, the owner of the document is an attorney or a law firm, and the participants *P* are the cloud providers.

Assessing the threshold scheme, it is evident it uses polynomial interpolation in a finite field. The secret shares are randomly generated using equation (1.1).

$$f(x) = s + a_1 x + a_2 x^2 + \ldots + a_{t-1} x^{t-1} mod\, p \tag{1.1}$$

where *p* is a publicly known prime number larger than *s* and *n, s* is the secret, $a_i \epsilon Z_p$, i = {1, 2,..., *t* - 1}, and $a_1,...,a_{t-1}$ and distinct $x_1,...,x_n$ are randomly chosen. The prime *p* should be large because an attacker knows *p* > *s* =>*s* ϵ {0, 1,..., *p* - 1}, so if *p* is low, there are less possible values an attacker can

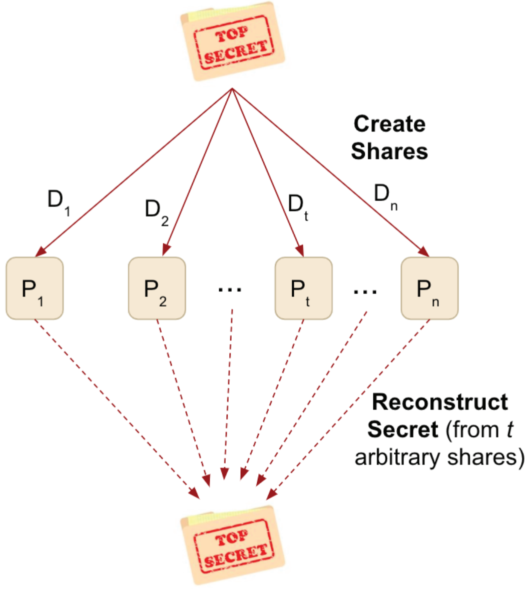

Figure 2. A (*t, n*)-threshold scheme where D_i is a share and P_i is a participant. Initially, the secret is split into shares, or fragments of the original secret that do not reveal any information about it. Each share D_i is distributed to the corresponding cloud provider P_i . In order to reconstruct the secret, at least *t* out of the original *n* shares must be recovered.

guess from to obtain the secret *s*. For the scheme, we chose a 19-digit prime that is just below the 64-bit integer limit in C++, the programming language used for this project.

The owner can reconstruct the shares by calculating the Lagrange interpolating polynomial as shown in equation (1.2).

$$f(x) = \sum_{j=1}^{t} y_j \prod_{k=1, k\neq j}^{t} \frac{x - x_k}{x_j - x_k} \; (mod\, p) \qquad (1.2)$$

where x_j and y_j are the input and output of equation (1.1) respectively. In both equations, the *mod p* provides additional security since the resulting curve is disjoint and unpredictable. Thus, these mathematics ensure privacy (confidentiality and un-linkability) and space efficiency.

To quantify the architecture's runtime, we measured the runtimes of secret sharing with a dataset of different file types and sizes (0-135 MB). In total, we tested 59 files of 29 different file types, and we ran AES-256-CFB and secret sharing at the same time (**Figures 3-4**). Secret sharing is noticeably more efficient than AES for both creating shares and reconstructing documents.

DISCUSSION

The experimental results support the criteria detailed in the hypothesis, particularly secret sharing's speed advantage over traditional encryption algorithms. Based on **Figures 3-4,** it is evident that the architecture is efficient even relative to other common encryption algorithms as the runtime lead of secret sharing increases with file size, ultimately giving it a substantial speed advantage over AES, widely used for encrypting data in the Cloud today. Due to numerous retests and revisions of the algorithms, the architecture adheres to the mathematics governing this scheme, guaranteeing sufficient privacy and data availability. The (2, 3)-threshold scheme provides data redundancy for increased availability since a share can be lost without preventing the reconstruction of the original secret. Therefore, secret sharing is a necessary preventative measure in the case of cloud provider key loss or simply adversaries revoking access to documents.

Overall, our experimental results indicate the low computational overhead of adding the secret-sharing approach to a multi-cloud environment, even on consumer grade hardware. This is important for integrating law firms as active participants into the architecture in the future. The efficient combination of algorithms satisfies the engineering goal as ChaCha20-Poly1305 warrants authenticity and privacy and secret sharing ensures availability and privacy. Compromising this system transpires only if multiple cloud providers collude, which is still unlikely given that the shares are additionally encrypted.

Our main goal for the future is to fully implement the secret sharing architecture specifically tailored to certain partner law firms where we will additionally evaluate corresponding security assumptions and processes. We also aim to better involve the client into the architecture, such as through giving them partial control over shares of their documents. Specific challenges for our work include addressing the problem of ownership of information and reliably auditing and amending stored documents.

METHODS

Throughout the design process, we focused on two parties: law firms and cloud providers shown in the final architecture designs (**Figures 5-6**). Following the design criteria, we ensured that the whole storage and retrieval process is protected against unauthorized access and modification through various cryptographic encryption and signature operations. This bolsters security along with adding scalability since law firms can customize the system to their own conditional access policies. However, the approach boasts end-to-end encryption and privacy from probing cloud providers, only assuming that classical network security protocols are administered, such as HTTPS for Internet data

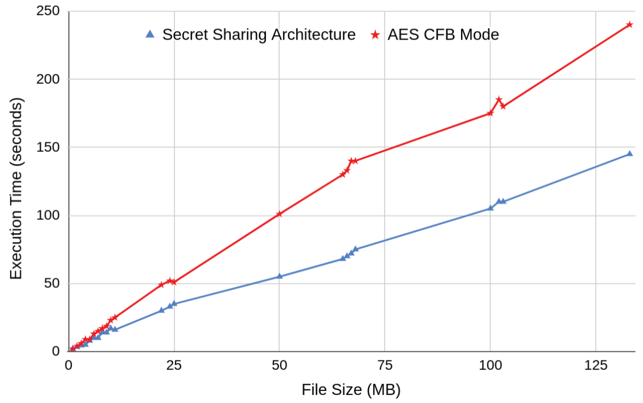

Figure 3. Performance Comparison of Creating Shares from the Document with a (2, 3) Scheme. To decrease sampling variability, each datapoint is the average of 10,000 individual trials of creating shares from a given file size. Compared with AES, the secret sharing implementation is about 39% faster for creating shares of the document.

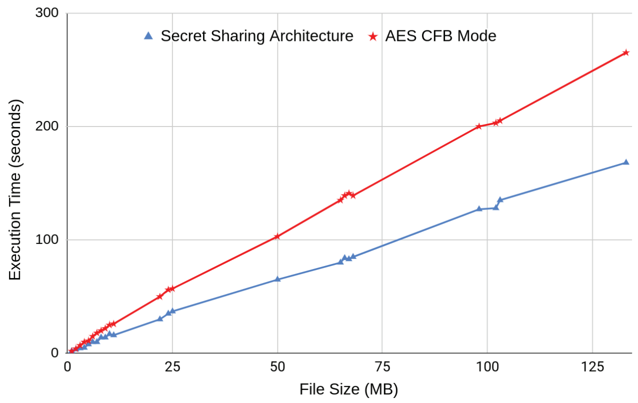

Figure 4. Performance Comparison of Document Reconstruction from Shares with a (2, 3) Scheme. To decrease sampling variability, each datapoint is the average of 10,000 individual trials of reconstructing a document of a given file size. Compared with AES, the secret sharing implementation is about 36% faster for reconstructing a document from its shares.

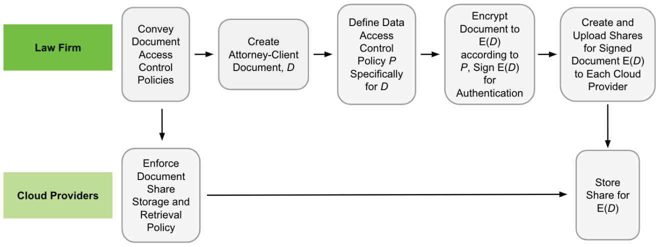
Law Firm
Cloud Providers
Convey Document Access Control Policies
Create Attorney-Client Document, *D*
Define Data Access Control Policy *P* Specifically for *D*
Encrypt Document to E(*D*) according to *P*, Sign E(*D*) for Authentication
Create and Upload Shares for Signed Document E(*D*) to Each Cloud Provider
Enforce Document Share Storage and Retrieval Policy
Store Share for E(*D*)

Figure 5. Law Firm Storage Process. The flowchart demonstrates the procedure a law firm would take to store documents. Access control policies ensure that unauthorized lawyers or other personnel cannot access the document while certain other lawyers are granted access for retrieval. Note that the document is encrypted to prevent even colluding cloud providers from compromising the document. The combination of cloud computing and access control policies produce a robust and scalable model for law firms to employ.

exchange and Virtual Private Networks between all party communications.

We developed the document storage and retrieval process after understanding and choosing the specific secret sharing and encryption algorithm we will employ (**Table 2**). Most of our revising and testing centered around finding efficient implementations for each component of the architecture. We programmed in C++ for maximum efficiency along with the benefits of object-oriented programming. Finally, we created a web application framework for the program to enhance its usability. We worked on a Mac OS notebook computer with a 1.4 GHz dual core processor and 4 GB RAM on Visual Studio Code 1.42.1.

Finally, we took performance metrics of our software's runtime as shown in **Figures 3-4**, evaluating this against the prior engineering design criteria. We executed all the tests on the consumer-grade hardware a lawyer has access to and made necessary revisions and retests upon noticing errors or optimizations. This will ideally present a complete view of the feasibility of this system for future application in an authentic environment.

ACKNOWLEDGEMENTS

The first author would like to thank his physics teacher and invaluable mentor, David Taylor, for the time he took out of his busy schedule to review the work and make suggestions. Additionally, the first author would like to recognize Prof. Brendan Saltaformaggio for the excellent resources and learning opportunities he provided to facilitate Joseph Zhang's exploration of cybersecurity. Last, but not least, we appreciate the thoughtful support from the editors of the journal during the revision process.

Received: March 22, 2020
Accepted: July 8, 2020
Published: July 19, 2020

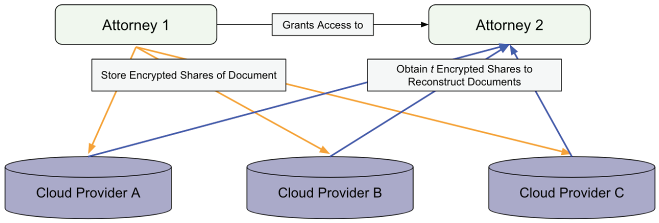
Attorney 1
Grants Access to
Attorney 2
Store Encrypted Shares of Document
Obtain *t* Encrypted Shares to Reconstruct Documents
Cloud Provider A
Cloud Provider B
Cloud Provider C

Figure 6. Secret Sharing Retrieval Process Assuming Attorney 1 Grants Access to Attorney 2. The law firm only needs to retrieve t out of the n original shares to reconstruct the document. Note that Attorney 2 must be granted access in the original access control policy to be able to receive the document.

REFERENCES

1. Goodman, Joanna, and Nicole Black. "The Tide Has Turned and The Cloud Is Here." *Legal IT Professionals*, 2012, www.legalitprofessionals.com/wpcs/cloudsurvey2012.pdf.
2. Horne, Dillon. "Cloud Computing, Virtual Law Firms, and the Legal Profession." *Cornell Law School Graduate Student Papers*, 2014, scholarship.law.cornell.edu/lps_papers/29.
3. Favro, Philip J. "Inviting Scrutiny: How Technologies Are Eroding the Attorney-Client Privilege." *SSRN Electronic Journal*, 2013, doi:10.2139/ssrn.2255206.
4. Kennedy, Dennis. "2019 Cloud Computing." *American Bar Association*, 2 Oct. 2019, www.americanbar.org/groups/law_practice/publications/techreport/abatechreport2019/cloudcomputing2019
5. Stinson, D. R. "An Explication of Secret Sharing Schemes." *Designs, Codes and Cryptography*, vol. 2, no. 4, 1992, pp. 357–390., doi:10.1007/bf00125203.
6. Shamir, Adi. "How to Share a Secret." *Communications of the ACM*, vol. 22, no. 11, 1979, pp. 612–613., doi:10.1145/359168.359176.
7. Langley, A., et al. "ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS)." *Internet Engineering Task Force (IETF)*, June 2016, doi:10.17487/rfc7905.