Appears as a non-archival poster presentation in the 22nd USENIX Conference on File and Storage Technologies (FAST'24)

Plugging the Leaks in Secure Archival Systems

Christopher Smith
Stony Brook University

Maliha Tabassum
Stony Brook University

Soumya Daruru
Stony Brook University

Gaurav Kulhare
Stony Brook University

Arvin Wang
Stony Brook University

Ethan Miller
UC Santa Cruz

Erez Zadok
Stony Brook University

Abstract

Archival systems aim to store data durably for long time periods while ensuring low storage costs and reasonable access times. Data stored in archives is often valuable and highly sensitive. Without security mechanisms, an adversary could read, delete, or tamper with archival data such as medical records, military intelligence, and trade secrets. Owing to the valuable nature of archival data and its long lifetime, this adversary is potentially quite powerful. They may be an Advanced Persistent Threat (APT) with the ability to infiltrate complex systems and expend vast computational resources. Particularly troublesome is the possibility of a “Harvest Now, Decrypt Later” attack that steals encrypted data with the hopes of decrypting it years later (e.g., with a quantum computer and improved cryptanalytic methods). Because this threat casts doubt on the security of any encoding method that relies on computational intractability assumptions (e.g., AES), ensuring the long-term confidentiality of data is one of the most difficult problems in secure archival [4].

Data encodings that provably protect data secrecy and do not rely on any computational assumptions are said to have *information-theoretic* (also known as *unconditional*) security. Shamir’s (t,n) secret sharing [14] is a well-known example of such an encoding, where the data is split into n shares such that t or more shares suffice to recover the secret, but fewer than t shares reveal no information about the data. Shamir’s secret sharing is widely used in secure archival works [3, 7, 17, 18], and it pairs well with a multi-cloud [2] storage architecture: a user secret-shares their data and each share is distributed to an independent cloud storage provider.

Prior works on secret-shared datastores fail to consider the threat of side-channel attacks. These works typically assume a provider compromise is detectable, and that a provider’s entire share is stolen. In reality, however, an APT may prefer to infiltrate a system undetected, by exploiting a hidden side channel, and leak only partial information about a share (and do so slowly over time). Shamir’s secret sharing scheme—among others—has been shown to be vulnerable to such leakage attacks [8]. Identifying all potential side-channels in a heterogeneous, multi-cloud archival system that evolves over time is a futile task. Perhaps a more realistic mitigation strategy is to integrate *leakage resilience* into the data encoding. This can be accomplished by *leakage-resilient secret sharing* (LRSS) schemes: a subject of recent theoretical study in the cryptographic community [1, 5, 6, 10–12].

While LRSS appears an attractive remedy to the threat of side-channels, it remains to be seen which, if any, LRSS approaches are feasible for use within secure archival systems. The primary considerations are security vs. storage efficiency trade-off, target leakage model, and compatibility with security renewal techniques. Existing analyses [1, 10, 12] suggest that Shamir’s secret sharing is not suitable for LRSS: the threshold must be set too high, and security depends on the number of shares, which is unfavorable for storage efficiency. Custom LRSS schemes achieve better security/storage trade-offs, but target various threat models for leakage we must choose from. One promising model for secure archival is the *local leakage* model, where adversaries independently leak bounded bits of information from each share. This model captures a large class of side channels corresponding to independent vulnerabilities within each cloud provider. More relaxed leakage models fail to capture common side channels, while more stringent models often result in impractical LRSS schemes (exponential storage overhead). Plus, it is difficult to envision realistic attacks against archives in more stringent models that are not already captured by local leakage.

Once we restrict ourselves to evaluating custom LRSS schemes against local leakage, it is still unclear whether these schemes can be made *proactive*. That is, can we “refresh” the shares without changing the underlying secret such that any leakage obtained by the adversary is rendered useless. Proactive security is a key component of secure archival systems, but all existing proactive schemes for secret sharing [9, 13, 15, 16] explicitly use Shamir’s. Combining LRSS with a proactive refreshing protocol is part of our ongoing work. Future work includes exploring Proofs of Retrievability for efficient integrity checks, improving system storage efficiency, and implementing a system prototype and simulator.

References

[1] Fabrice Benhamouda, Akshay Degwekar, Yuval Ishai, and Tal Rabin. On the local leakage resilience of linear secret sharing schemes. *Journal of Cryptology*, 34:1–65, 2021. https://link.springer.com/article/10.1007/s00145-021-09375-2.

[2] Robert Bohn, Craig Lee, and Martial Michel. The nist cloud federation reference architecture, 2020. https://doi.org/10.6028/NIST.SP.500-332.

[3] Johannes Braun, Johannes Buchmann, Denise Demirel, Matthias Geihs, Mikio Fujiwara, Shiho Moriai, Masahide Sasaki, and Atsushi Waseda. Lincos: A storage system providing long-term integrity, authenticity, and confidentiality. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, page 461–468, New York, NY, USA, 2017. Association for Computing Machinery. https://dl.acm.org/doi/10.1145/3052973.3053043.

[4] Johannes Braun, Johannes Buchmann, Ciaran Mullan, and Alex Wiesmaier. Long term confidentiality: a survey. *Designs, Codes and Cryptography*, 71:459–478, 2014. https://eprint.iacr.org/2012/449.pdf.

[5] Nishanth Chandran, Bhavana Kanukurthi, Sai Lakshmi Bhavana Obbattu, and Sruthi Sekar. Short leakage resilient and non-malleable secret sharing schemes. In *Annual International Cryptology Conference*, pages 178–207. Springer, 2022. https://eprint.iacr.org/2022/216.pdf.

[6] Eshan Chattopadhyay, Jesse Goodman, Vipul Goyal, Ashutosh Kumar, Xin Li, Raghu Meka, and David Zuckerman. Extractors and secret sharing against bounded collusion protocols. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1226–1242, 2020.

[7] Gregory R Ganger, Pradeep K Khosla, and CARNEGIE-MELLON UNIV PITTSBURGH PA. Pasis: A distributed framework for perpetually available and secure information systems, 2005. https://apps.dtic.mil/sti/citations/ADA436245.

[8] Venkatesan Guruswami and Mary Wootters. Repairing reed-solomon codes. *IEEE Transactions on Information Theory*, 63(9):5684–5698, 2017. https://ieeexplore.ieee.org/document/7922614.

[9] Amir Herzberg, Stanisław Jarecki, Hugo Krawczyk, and Moti Yung. Proactive secret sharing or: How to cope with perpetual leakage. In *Advances in Cryptology—CRYPT0'95: 15th Annual International Cryptology Conference Santa Barbara, California, USA, August 27–31, 1995 Proceedings 15*, pages 339–352. Springer, 1995. https://link.springer.com/chapter/10.1007/3-540-44750-4_27.

[10] Ohad Klein and Ilan Komargodski. New bounds on the local leakage resilience of shamir's secret sharing scheme. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023*, pages 139–170, Cham, 2023. Springer Nature Switzerland. https://link.springer.com/chapter/10.1007/978-3-031-38557-5_5.

[11] Hemanta K. Maji, Hai H. Nguyen, Anat Paskin-Cherniavsky, Tom Suad, Mingyuan Wang, Xiuyu Ye, and Albert Yu. Leakage-resilient linear secret-sharing against arbitrary bounded-size leakage family. In Eike Kiltz and Vinod Vaikuntanathan, editors, *Theory of Cryptography*, pages 355–383, Cham, 2022. Springer Nature Switzerland. https://link.springer.com/chapter/10.1007/978-3-031-22318-1_13.

[12] Hemanta K. Maji, Hai H. Nguyen, Anat Paskin-Cherniavsky, and Mingyuan Wang. Improved bound on the local leakage-resilience of shamir's secret sharing. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 2678–2683, 2022.

[13] Rafail Ostrovsky and Moti Yung. How to withstand mobile virus attacks (extended abstract). In *Proceedings of the Tenth Annual ACM Symposium on Principles of Distributed Computing*, PODC '91, page 51–59, New York, NY, USA, 1991. Association for Computing Machinery. https://doi.org/10.1145/112600.112605.

[14] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, nov 1979. https://doi.org/10.1145/359168.359176.

[15] Akshayaram Srinivasan and Prashant Nalini Vasudevan. Leakage resilient secret sharing and applications. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 480–509, Cham, 2019. Springer International Publishing. https://eprint.iacr.org/2018/1154.pdf.

[16] Douglas R Stinson and Ruizhong Wei. Unconditionally secure proactive secret sharing scheme with combinatorial structures. In *International Workshop on Selected Areas in Cryptography*, pages 200–214. Springer, 1999. https://link.springer.com/chapter/10.1007/3-540-46513-8_15.

[17] Mark W. Storer, Kevin M. Greenan, Ethan L. Miller, and Kaladhar Voruganti. Potshards—a secure, recoverable, long-term archival storage system. *ACM Trans. Storage*, 5(2), jun 2009. https://dl.acm.org/doi/10.1145/1534912.1534914.

[18] T.M. Wong, Chenxi Wang, and J.M. Wing. Verifiable secret redistribution for archive systems. In *First International IEEE Security in Storage Workshop, 2002. Proceedings.*, pages 94–105, 2002. https://ieeexplore.ieee.org/document/1183515.