A RESILIENT DISPERSAL SCHEME FOR MULTI-CLOUD STORAGE

V. Sarala [1], Ch. Rakesh Kumar,

[1]**Assistant professorof PG Department ,** Dantuluri Narayana Raju College**, Bhimavaram, Andharapradesh**
Email:- vedalasarala21@gmail.com
[2]**PG Student of MCA, Dantuluri** Narayana Raju College**, Bhimavaram, Andharapradesh**
Email:- chinimillirakeshkumar@gmail.com

ABSTRACT

We have entered an era where copious amounts of sensitive data are being stored in the cloud. To meet the rising privacy, reliability, and verifiability needs, we propose Gecko, a multi-cloud dispersal scheme where: (a) the key used to encrypt the data file is the secret in a Latin-square-autotopism secret sharing scheme, (b) data files and encryption keys are dispersed separately to multiple clouds, and (c) a blockchain-based integrity-check protocol is devised to pinpoint faulty data. Gecko enables fast and thorough key renewal: when a portion of the key (the secret) is leaked, we replace all shares of the partially-leaked secret without replacing the secret itself; this immediately resists targeted attack to certain file without re-encrypting the data file itself. Key renewal is further accelerated by the blockchain-based integrity check. We evaluate Gecko theoretically and experimentally against the traditional AONT-RS dispersal scheme, drawing two conclusions: 1) Gecko admits powerful key renewal and identification of damaged data, with a minor transfer overhead; and 2) Gecko performs key renewal three to five times faster than AONT-RS hybrid-slice renewal (the closest thing AONT-RS has to key renewal).

1 INTRODUCTION

Cloud storage is convenient but gives rise to multiple concerns, such as privacy [1], fault tolerance [2], and verifiable integrity [3], particularly when the stored data is sensitive and large-scale. As one example among many, CareCloud1 stores information such as electronic health records using Amazon S3 and Amazon Glacier [4]; for privacy protection and accuracy of analysis, CareCloud requires the outsourced data to be kept confidential, retrievable, and intact. For sensitive systems such as CareCloud, a multi-cloud dispersal scheme [2] is beneficial, in which a sensitive file is fragmented into file slices and each of file slices (with corresponding integrity credentials) is dispersed to a distinct cloud service provider (CSP). With proper redundancy, the file remains retrievable even when some CSPs fail or are malicious. To fragment data files, symmetric encryption can be combined with erasure codes to balance security and the consequential overhead [5]–[7]. Regrettably, key-protection methods in current schemes are insufficient.

Literature Survey

Security and privacy challenges in cloud computing environments

The cloud computing paradigm is still evolving, but has recently gained tremendous momentum. However, security and privacy issues pose as the key roadblock to its fast adoption. In this article, the authors present security and privacy challenges that are exacerbated by the unique aspects of clouds and show how they're related to various delivery and deployment models. They discuss various approaches to address these challenges, existing solutions, and future work needed to provide a trustworthy cloud computing environment.

3 IMPLEMENTATION STUDY

EXISTING SYSTEM:

Ideally an encryption key that has been dispersed should have stronger protection than an encrypted file does, so the two dispersals should be performed independently; thus, combining file and key then dispersing, as in [6], [7], is not the most secure option. In [5], secret sharing is utilized independently for encryption keys, which guarantees high-level key secrecy; however, key recovery in case of failures or accidents is not efficient due to the recovery procedure requiring too much computation. In addition, a leaked key in some schemes may be useful in a number of attacks.

Proposed System & alogirtham

In this project, we proposed Gecko, a multi-cloud dispersal scheme that utilizes a Latin-square autotopism secret sharing scheme and blockchain technology. It improves on the traditional AONT-RS by having additional functionality: (a) distributed integrity verification, which is performed after a slice is downloaded, and (b) key renewal and file-slice repair, which are both faster than complete renewal. Experimental results indicate that Gecko performs key renewal 3 to 5 times faster than AONT-RS performs complete renewal, and the upload and download overhead is minor. It is also important to note that cloud latency and the blockchain latency plays a determining role in the latency of both Gecko and AONT-RS.

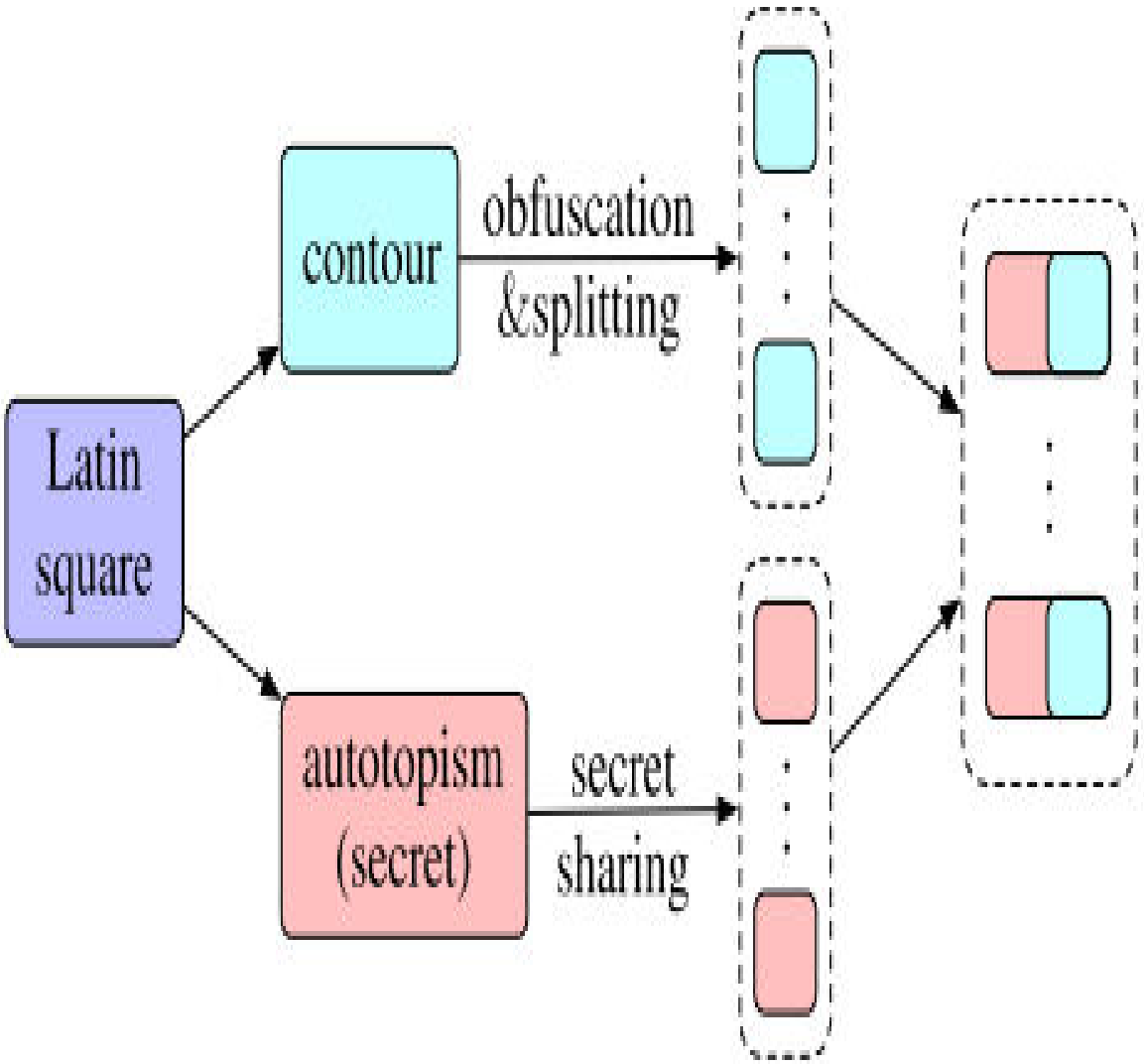


Fig:3.1 System Architecture

IMPLEMENTATION

MODULES

Owner

- Register
- Login
- File Upload
- View Upload Details
- View Requests
- Logout

Attacker

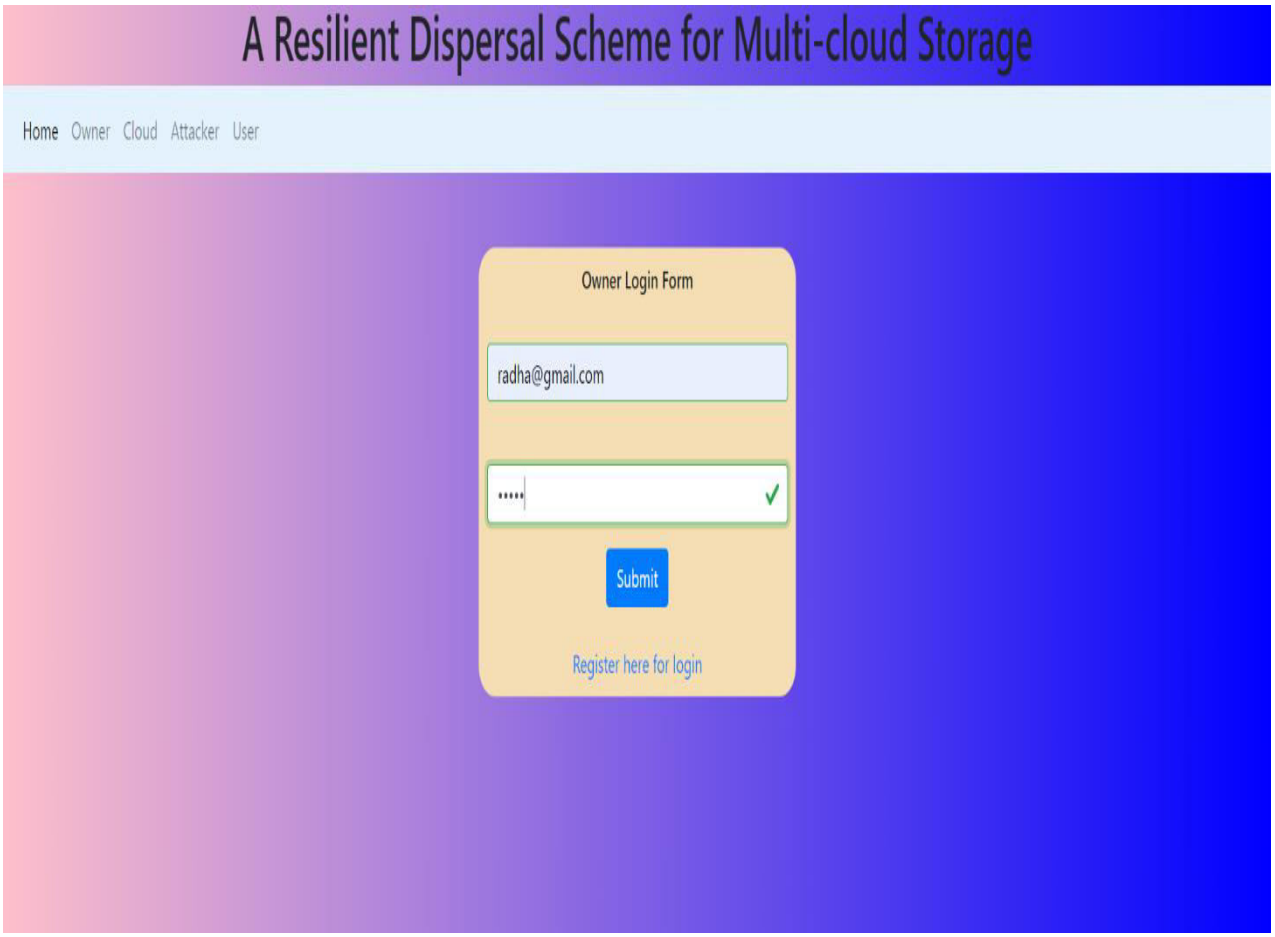
- Login
- View Owners
- View users
- View Cloud Requests
- Logout

5 RESULTS AND DISCUSSION

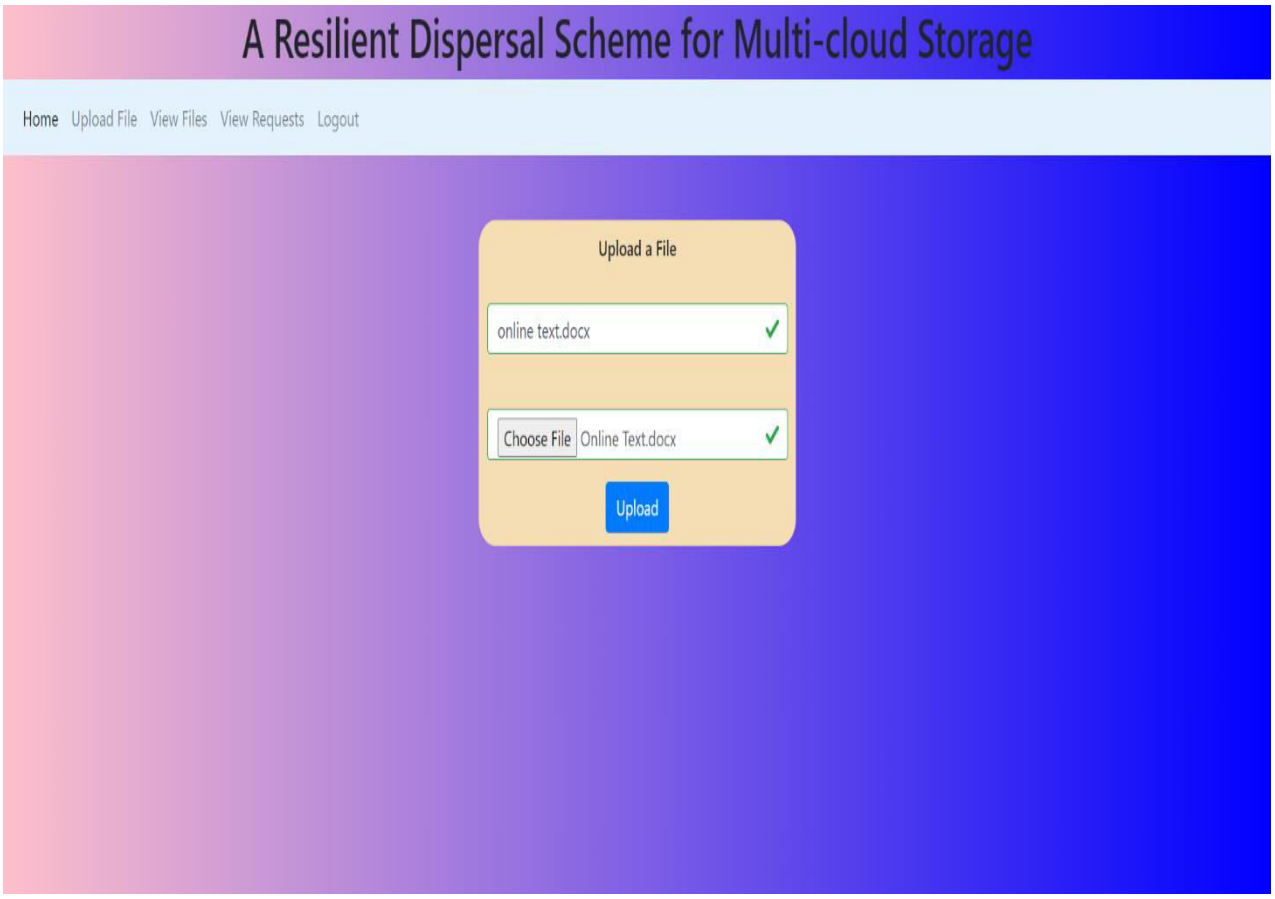
SCREEN SHOTS

7.1 HOME PAGE

7.2 OWNER LOGIN

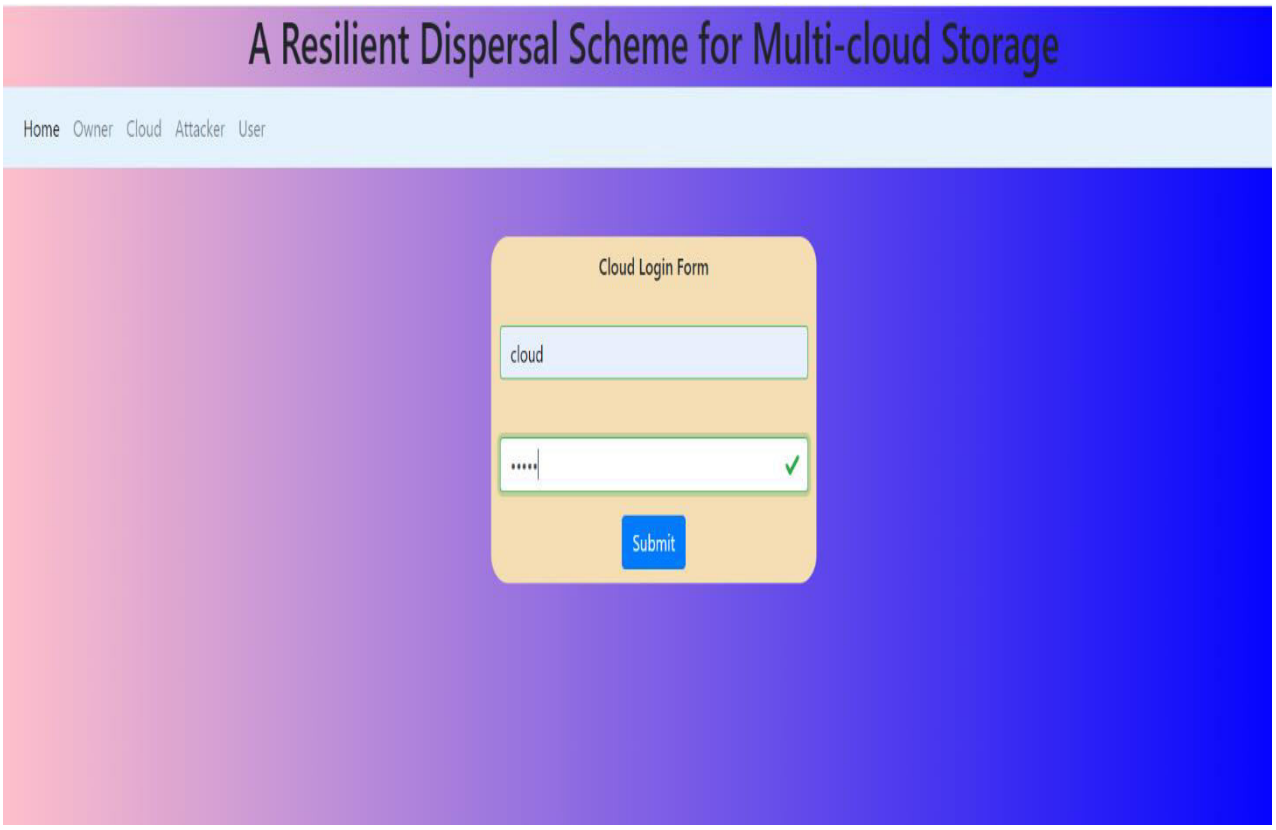


7.3 FILE UPLOAD



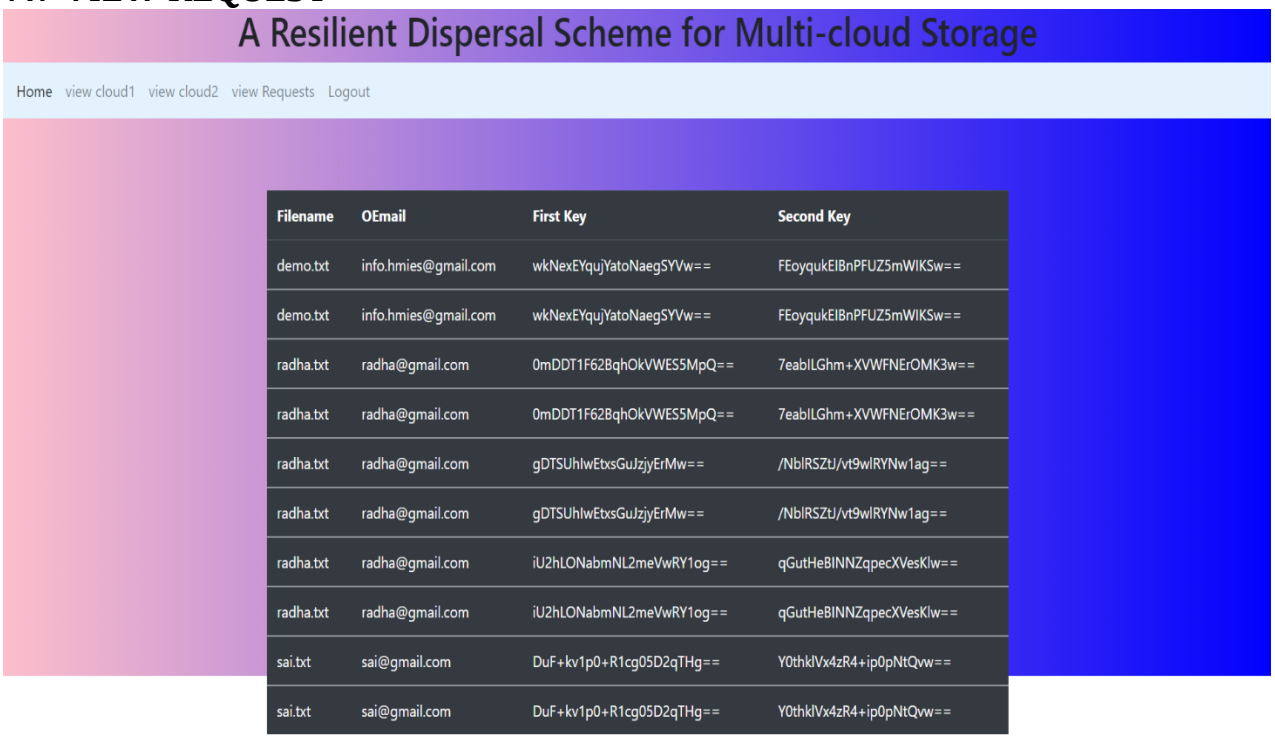
7.4 VIEW FILES

7.5 CLOUD LOGIN FORM



7.6 VIEW CLOUD

7.7 VIEW REQUEST

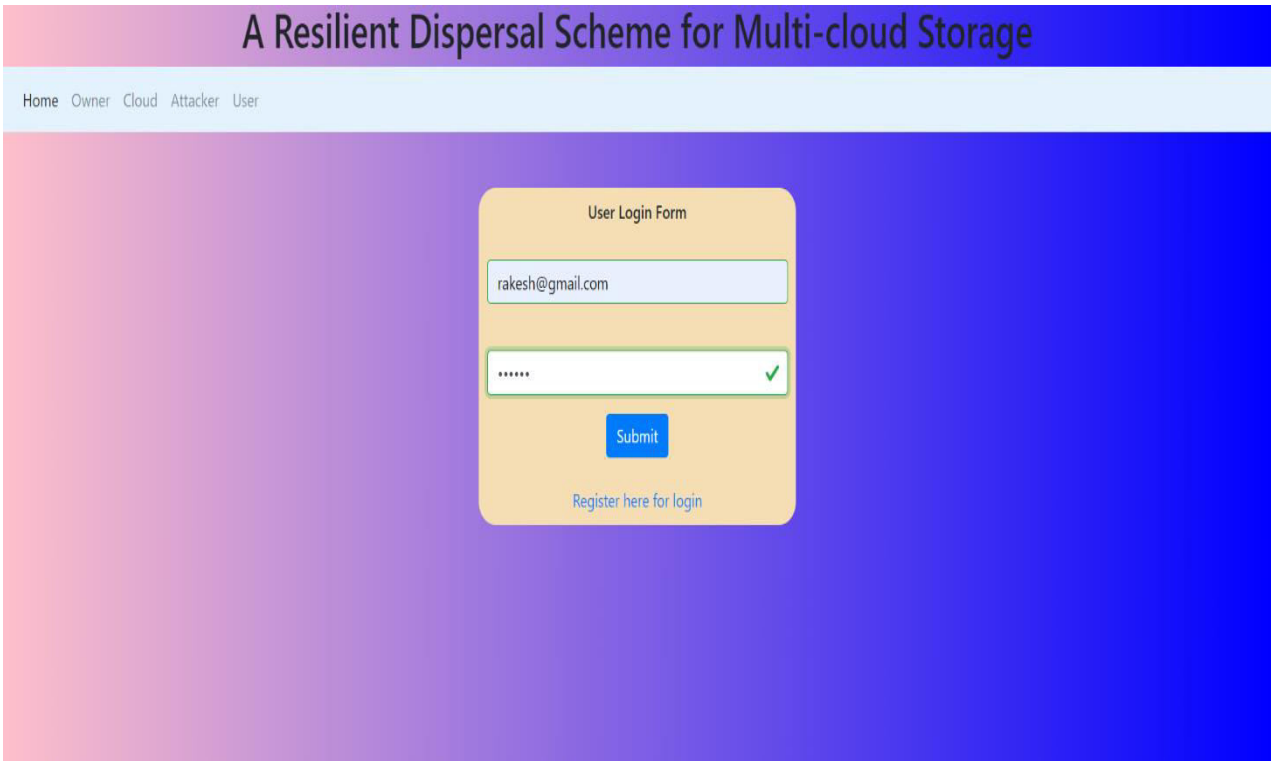


Filename	OEmail	First Key	Second Key
demo.txt	info.hmies@gmail.com	wkNexEYqujYatoNaegSYVw==	FEoyqukEIBnPFUZ5mWIKSw==
demo.txt	info.hmies@gmail.com	wkNexEYqujYatoNaegSYVw==	FEoyqukEIBnPFUZ5mWIKSw==
radha.txt	radha@gmail.com	0mDDT1F62BqhOkVWES5MpQ==	7eabILGhm+XVWFNErOMK3w==
radha.txt	radha@gmail.com	0mDDT1F62BqhOkVWES5MpQ==	7eabILGhm+XVWFNErOMK3w==
radha.txt	radha@gmail.com	gDTSUhIwEtxsGuJzjyErMw==	/NblRSZtJ/vt9wlRYNw1ag==
radha.txt	radha@gmail.com	gDTSUhIwEtxsGuJzjyErMw==	/NblRSZtJ/vt9wlRYNw1ag==
radha.txt	radha@gmail.com	iU2hLONabmNL2meVwRY1og==	qGutHeBINNZqpecXVesKlw==
radha.txt	radha@gmail.com	iU2hLONabmNL2meVwRY1og==	qGutHeBINNZqpecXVesKlw==
sai.txt	sai@gmail.com	DuF+kv1p0+R1cg05D2qTHg==	Y0thklVx4zR4+ip0pNtQvw==
sai.txt	sai@gmail.com	DuF+kv1p0+R1cg05D2qTHg==	Y0thklVx4zR4+ip0pNtQvw==

7.8 CLOUD RESPON7.9 CLOUD DATA

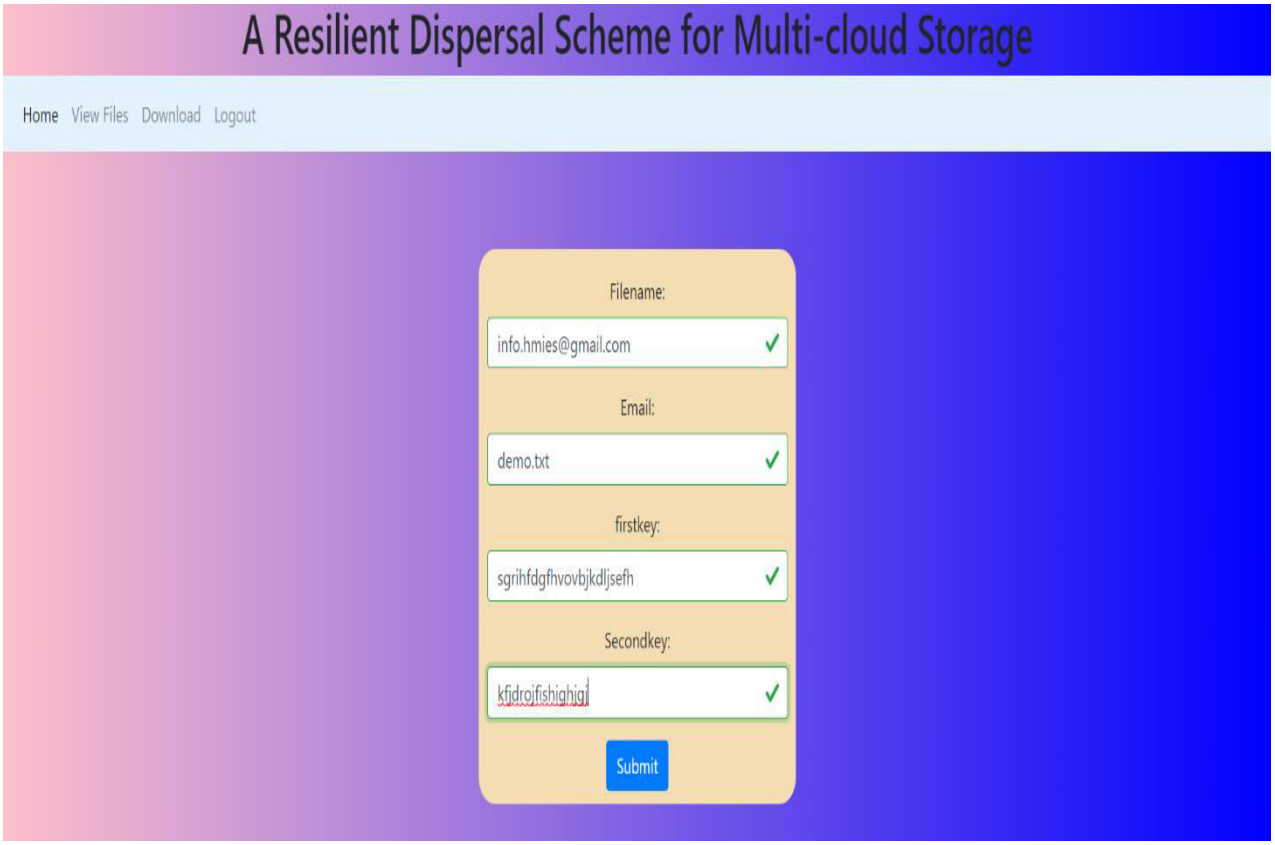
Uemail	filename	Oemail	Status	Response
rchmi.project@gmail.com	demo.txt	info.hmies@gmail.com	response sent	Response
rakesh@gmail.com	sai.txt	sai@gmail.com	pending	Response

7.10 USER LOGIN

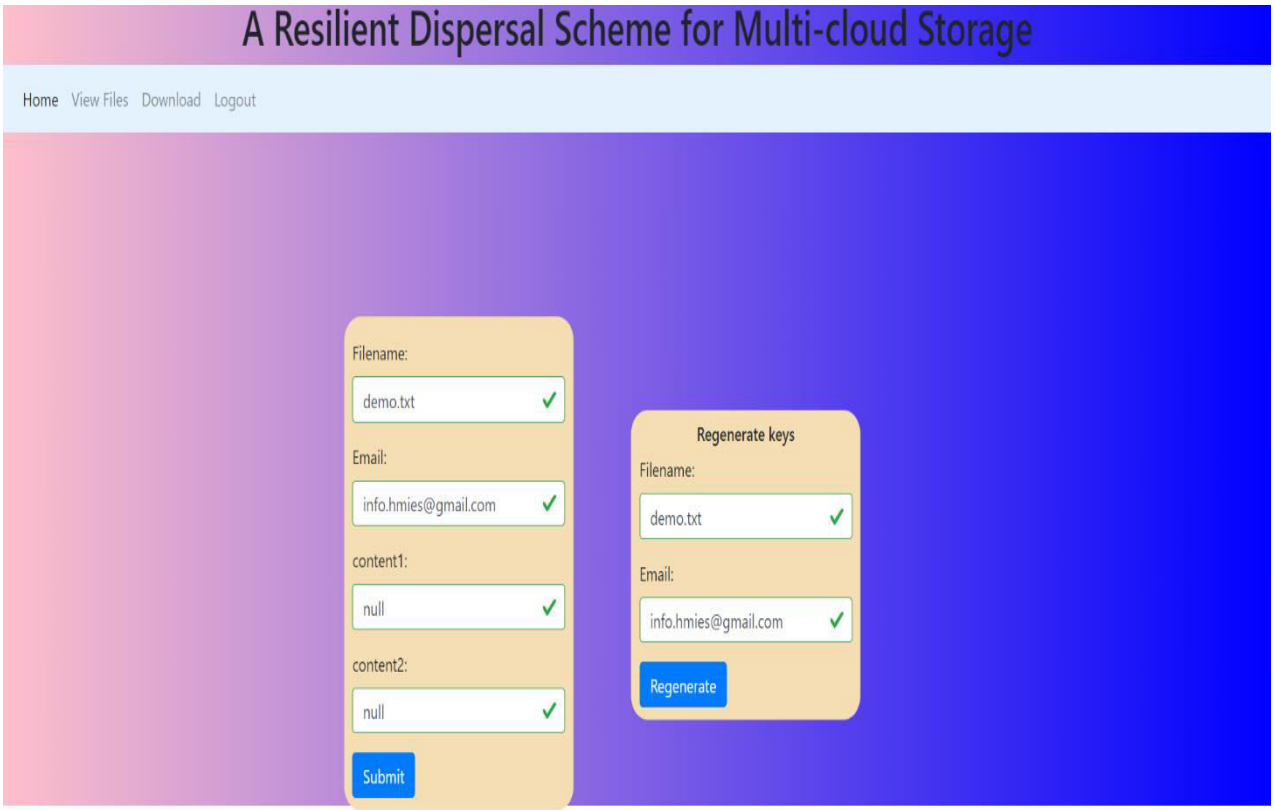


7.11 FILE DOWNLOAD REQUEST

7.12 DOWNLOAD FILE



7.13KEY GENERATION



6. CONCLUSION AND FUTURE WORK

CONCLUSION

In this paper, we proposed Gecko, a multi-cloud dispersal scheme that utilizes a Latin-square autotopism secret sharing scheme and blockchain technology. It improves on the traditional AONT-RS by having additional functionality: (a) distributed integrity verification, which is performed after a slice is downloaded, and (b) key renewal and file-slice repair, which are both faster than complete renewal. Experimental results indicate that Gecko performs key renewal 3 to 5 times faster than AONT-RS performs complete renewal, and the upload and download overhead is minor. It is also important to note that cloud latency and the blockchain latency plays a determining role in the latency of both Gecko and AONT-RS. One challenge of this work is deciding how to store proof of-data in the blockchain. We devise a double signature to achieve non-repudiation, accountability, and public verifiability. We

envisage Gecko being used as a kind of storage ‘‘middleman’’ (such as a logical layer in a trusted server): users send their data (either encrypted or unencrypted) to the Gecko operator, who maintains the data on multiple clouds on behalf of the users.

7. REFRENCES

[1] H. Takabi, J. B. D. Joshi, and G.-J. Ahn, ‘‘Security and privacy challenges in cloud computing environments,’’ IEEE Security Privacy, vol. 8, no. 6, pp. 24–31, Nov./Dec. 2010.

[2] D. K. Bowers, A. Juels, and A. Oprea, ‘‘HAIL: A high-availability and integrity layer for cloud storage,’’ in Proc. CCS, Nov. 2008, pp. 187–198.

[3] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ‘‘Provable data possession at untrusted stores,’’ in Proc. CCS, Oct. 2007, pp. 598–609.

[4] A. W. Services. (2019). AWS Partner Story: CareCloud. Accessed: Jan. 9. 2019. [Online]. Available: https://aws. amazon.com/cn/partners/success/carecloud/

[5] H. Krawczyk, ‘‘Secret sharing made short,’’ in Proc. Annu. Int. Cryptol. Conf. Berlin, Germany: Springer, 1993, pp. 136–146.

[6] J. K. Resch and J. S. Plank, ‘‘AONT-RS: Blending security and performance in dispersed storage systems,’’ in Proc. FAST, 2011, pp. 1–12.

[7] M. Li, C. Qin, P. P. C. Lee, and J. Li, ‘‘Convergent dispersal: Toward storage-efficient security in a cloud-of-clouds,’’ in Proc. HotStorage, Jul. 2014, pp. 1–5.

[8] C. Cachin, I. Keidar, and A. Shraer, ‘‘Trusting the cloud,’’ ACM SIGACT News, vol. 40, no. 2, pp. 81–86, Jun. 2009.

[9] J. Novet. (2017). Microsoft Confirms Azure Storage Issues Around the World (Updated). Accessed: Feb. 15, 2018. [Online]. Available: https:// venturebeat.com/2017/03/15/microsoft-confirms-azure-storage-issuesaround-the-world/

[10] T. Robinson. (2018). Open AWS S3 Bucket Exposes Private info on Thousands of Fedex Customers. Accessed: Nov. 6, 2018. [Online]. Available: https://www.scmagazine.com/

[11] L. Razavi. (2014). The IC Cloud Leak: Weak Security isn’t Only a Problem for Apple’s Backup Service. Accessed: Sep. 2, 2016. [Online]. Available: https://www.newstatesman.com/sci-tech/

[12] M. A. AlZain, E. Pardede, B. Soh, and J. A. Thom, ‘‘Cloud computing security: From single to multi-clouds,’’ in Proc. HICSS, Jan. 2012, pp. 5490–5499.

[13] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, ‘‘RACS: A case for cloud storage diversity,’’ in Proc. SoCC, Jun. 2010, pp. 229–240.