Journal of Nonlinear Analysis and Optimization Vol. 15, Issue. 1, No.15 : 2024 ISSN : **1906-9685** 



## A NEW LIGHT WEIGHT SYMMETRIC SEARCHABLE ENCRYPTION SCHEME FOR STRING IDENTIFICATION

V. Sarala<sup>1</sup> Ch. Surekha<sup>2</sup> <sup>1</sup>Assistant professor, PG DEPT, Dantuluri Narayana Raju College, Bhimavaram, Andhra Pradesh Email: - vedalasarala21@gmail.com <sup>2</sup>PG Student of MCA, Dantuluri Narayana Raju College, Bhimavaram, Andhra Pradesh Email: - rekhachalla0770@gmail.com

## ABSTRACT

In this paper, we provide an efficient a new light weight symmetric and easy-to-implement symmetric searchable encryption scheme (SSE) for string search, which takes one round of communication, O(n) times of computations over n documents. Unlike previous schemes, we use hash-chaining instead of chain of encryption operations for index generation, which makes it suitable for lightweight applications. Unlike the previous SSE schemes for string search, with our scheme, server learns nothing about the frequency and the relative positions of the words being searched except what it can learn from the history. We are the first to propose probabilistic trapdoors in SSE for string search. We provide concrete proof of non-adaptive security of our scheme against honest-but-curious server based on the definitions of [12]. We also introduce a new notion of search pattern privacy, which gives a measure of security against the leakage from trapdoor. We have shown that our scheme is secure under search pattern indistinguishability definition. We show why SSE scheme for string search cannot attain adaptive indistinguishability criteria as mentioned in [12]. We also propose modifications of our scheme so that the scheme can be used against active adversaries at the cost of more rounds of communications and memory space. We validate our scheme against two different commercial datasets.

## **1 INTRODUCTION**

The cloud is designed to hold a large number of encrypted documents. With the advent of cloud computing, growing number of clients and leading organizations have started adapting to the private storage outsourcing. This allows resource constrained clients to privately store large amounts of encrypted data in cloud at low cost. However, this prevents one from searching. This gives rise to

a newly emerging field of research, called searchable encryption (SE). SE can be classified into symmetric searchable encryptions (SSE) and asymmetric searchable encryptions (ASE). In this paper, we study the SSE for string search. In the SSE, the client encrypts the data and stores it on the cloud. It may be noted that client can organize the data in an arbitrary manner and can maintain additional data structures to achieve desired data efficiently. In this process, the initial client for both client and the cloud server. Since huge volumes of documents are stored in a cloud server, searching against a keyword may result into large number of documents, most of which are not intended, causing unnecessary network traffic. This So motivates the idea of searching against a string, which allows the search to be more specific. Searching for string is a multi keyword search where the ordering of keywords is preserved.

## LITERATURE SURVEY

There are numerous lightweight symmetric encryption techniques that can be utilized for string identification projects that are already available. [1] The paper by Wheeler and Schroeppel presents an optimization technique for the SHA-1 cryptographic hash function using assembly language. It is considered an important contribution to the optimization of hash functions and has been widely cited in the literature.[2] The paper by Rogaway and Shrimpton provides a comprehensive overview of the basics of cryptographic hash functions, including definitions and properties of preimage resistance, second-preimage resistance, and collision resistance. The paper has been cited extensively in the literature as a reference for these topics.[3] The paper by Merkle introduces the concept of a certified digital signature, which provides a way to prove the authenticity and integrity of a digital document. This paper is considered a seminal contribution to the field of digital signatures.[4] The Handbook of Applied Cryptography by Menezes et al. is a widelyused reference book in the field of cryptography. It covers a wide range of topics, from basic concepts to advanced techniques, and is often used as a textbook for courses on cryptography.[5] A Course in Number Theory and Cryptography by Koblitz is a textbook that covers the mathematical foundations of cryptography, including number theory and algebraic structures.

## **IMPLEMENTATION STUDY**

#### **EXISTING SYSTEM:**

Dynamic SSE was first considered by Song et al. [19], but no solution with sublinear search time existed before the work of Kamara et al. [13]. Recently, two new dynamic SSE schemes have been proposed. The first one, by Cash et al. [9], which is an extension of [10]. They showed that SSE is feasible on very large databases. In [9], authors designed and implemented dynamic

symmetric searchable encryption schemes that efficiently and privately search server held encrypted databases with tens of billions of record-keyword pairs.

#### **DISADVANTAGES:**

- The system is not More secure and an efficient due to lack of lightweight cryptography.
- The system is not under Symmetric key encryption and Searchable encryption.

#### **PROPOSED SYSTEM & ALOGIRTHAM**

In the proposed system, the system studies the SSE for string search. In the SSE, the client encrypts the data and stores it on the cloud. It may be noted that client can organize the data in an arbitrary manner and can maintain additional data structures to achieve desired data efficiently. In this process, the initial client-side computation is thus as large as the data, but subsequent computations to access data is less for both client and the cloud server. Since huge volumes of documents are stored in a cloud server, searching against a keyword may result into large number of documents, most of which are not intended, causing unnecessary network traffic.

#### **4.1 ADVANTAGES:**

- The system proposes a non-adaptively secure SSE scheme for string search which takes one round of communication, O(n) times of computation over n documents, O(n) additional memory in the server side and no memory in the client side.
- The system provides a formal and proof to show that the scheme is non adaptively secure.



Fig:3.1 System Architecture

## **IMPLEMENTATION**

#### MODULE

#### **Cloud Data Server**

In this module, the Data Server login by using valid user name and password. After login successful he can do some operations, such as View Owners & Authorize, View Users & Authorize, View User Request, View Cloud Server Files, View Transactions, View Attackers, View Time Delay Results, and View Throughput Results

#### View and Authorize Users

In this module, the admin can view the list of users who all registered. In this, the admin can view the user's details such as, user name, email, address and admin authorize the users.

#### **Data Owner**

In this module, there are n numbers of Data Owners are present. Data Owner should register before doing any operations. Once Owner registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful Owner will do some operations like Upload, View My Files, View My Profile, Verify, Delete File

# 5 RESULTS AND DISCUSSION SCREENSHOTS 5.2.1 TOMCAT MANAGER



## Fig: 5.1 5.2.2 TOMCAT WEB APPLICATION MANAGER

💌 🗷 /manager 🗙 +					- o >
← → C () localhost:8080/manager/html					☆ ▷   • :
The Apache Software Foundation http://www.apache.org/	ı				
	Tomcat We	eb Application Man	ager		
Message: OK					
Manager					
List Applications	HTML Manager He	R	Man	ager Help	Server Status
Applications					
Path		Display Name	Running	Sessions	Commands
L		Welcome to Tomcat	true	Q	Start   Stop   Reload   Undeploy     Expire sessions   with idle ≥ 30   minutes
/A New Lightweight Symmetric Searchable Encryption Scheme	e for String Identification		true	Q	Start   Stop   Reload   Undeploy     Expire sessions   with idle ≥ 30   minutes
/docs		Tomcat Documentation	true	۵	Start   Stop   Reload   Undeploy     Expire sessions   with idle ≥ 30   minutes
/examples		Servlet and JSP Examples	true	۵	Start   Stop   Reload   Undeploy     Expire sessions   with idle ≿ 30   minutes
/host-manager		Tomcat Manager Application	true	Q	Start   Stop   Reload   Undeploy     Expire sessions   with idle ≥ 30   minutes
					Start Stop Reload Undenlov

# Fig: 5.2

## 5.2.3 DATA OWNER REGISTER

VeuTube X Register X +		- 0 ×
← → C O localhost:8080/A%20Nev%20Lightweight%20Symmetric%20Searchable%20Encryption%20Scheme%20for%20String%20Identification/Register.jsp	<b>@ </b> ☆	D :
		*
Search our ster		
Menu		
Home		
Cloud Data Server		
Data Owner		
End User Name (neguined) surekha		
Password (required)		
Email Address (required) rekhachalla0770@gmail.co		
Mobile Number(required) 6305912578		
Your Address		
20/2/2002		
Gender(required) Female 🗸		
Pincode 534245		
Location bhimavaram		
Select Profile Pic(required) Choose File No file chosen		
Submit		
2 89'F Q Search (m ) 2 (0 M ) 2 (0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0	ට ENG ඉ දා ම 🔒	19:51

# Fig: 5.3

## 5.2.4 END USER REGISTER

Search our ste: Q	End User Regis	ter	
Menu			
Home		1113 constant	
Cloud Data Server		REJON	
Data Owner			
End Licer	Manua (maniland)	aurakha	
Ellu Usel	Name (required)	surekna	
	Password (required)	reiche abelle 0770 @erneil es	
	Email Address (required)	reknachaliao///o@gmail.col	
	Mobile Number(required)	6305912578	
		BHINAVAKAM	
	Your Address		
	DOB (required)	20/2/2002	
	Gender(required)	Female V	
	Pincode	534245	
	Location	phimavaram	
	Select Profile Pic(required)	Choose File No file chosen	

Fig: 5.4

## 5.2.5 CLOUD DATA SERVER LOGIN

	Platform as a Service(PaaS) Infrastructure as a Service(laaS).
Search our ster Control of Contro	Cloud Data Server Login

## 5.2.6 CLOUD DATA SERVER MAIN

Search our ste:	Q Welcome Cloud Data Server Main
Data Serve	er Menu
View Owners & Authori View Users & Authorize View User Request View Cloud Server File View Transactions View Attackers	zz s
View Time Delay Resul View Throughput Resul	ts Its
Log Out	

## Fig: 5.6

## 5.2.7 VIEW DATA OWNER & AUTHORIZE

	Ci la			Infra Servi	structure as ice(IaaS).	a	-	
View Data O	wner 8	Authorize						
Owner Image	<b>Owner Name</b> Rajesh	E.Mail Rajesh.1230gmail.com	<b>Mobile</b> 9535866270	Address #8928,4th Main,Rajajinagar,Bangalore- 21	DOB 05/06/1989	Location Bangalore	Status Authorized	l
	Manjunath	tmksmanju118gmail.com	9535866270	#7827,3rd Block,Rajajinagar,Bangalore	05/06/1989	Bangalore	Authorized	l
	dinesh	aa§aa.com	9347225321	Asyb	11-may-1999	vakp	Authorized	l
	surekha	rekhachalla0770@gmail.com	6305912578	bhimavaram	20/2/2002	bhimavaram	Authorized	
<u>Go.Back</u>								•

#### 5.2.8 VIEW END USER & AUTHORIZE



Fig:

5.8

### 5.2.9 END USER LOGIN PAGE

👻 🔣 End User 🛛 🗙 🕂		-	0 X	5
← → C ③ localhost:8080/A%20New%	20Lightweight%20Symmetric%20Searchable%20Encryption%20Scheme%20for%20String%20Identification/EndUser.jsp	Ð	0 :	8
				•
	Cloud storage, Symmetric key, Searchable encryption, Cloud storage, A hash-chain, lightweight cryptography. A hash-chain, lightweight cryptography.			
				l
Sea	arch our ste: Q End User Login			l
Me	enu			l
Home	e 🔶			l
Cloud	d Data Server			l
End L	User Name (required)			l
				l
	rásswora (requireg)			1
	Register Submit			

## Fig: 5.9

# 5.2.10 VIEW USER REQUEST & PERMIT

Utser Nume   Nume   Owner Nume   Req Date   Res Date   MAC   Sk   Status     tmksmanju   ToyExample.txt   Manjunath   30/09/2018 13:17:48   30/09/2018 13:18:16:06   -1ace7be87e4dc02b0c0e8ef65bf892c1211a245b   [B9dd561]   Xas     tmksmanju   Cloud.java   Manjunath   30/09/2018 33:19:29   30/09/2018 31:19:39   38166df30f3bef7168133bfb80eba3cd82be71be   [B9146bcdb   Xas     tmksmanju   Lit.ism   Bainab   30/09/2018   30/09/2018   2252733716a7c0aa92a013c456a06F0a82ae2ae2   [B9147ba]   Yas	View (	Jser Reque	st & Per	rmit		Service(PaaS)	1		
tmksmanju   ToyExample.txt   Manjunah   30/09/2018 33:17:48   30/09/2018 13:18:06   -lace7be87e4dc03b0c0e8ef65bf892cl211a245b   [B9dd561]   Xas     tmksmanju   Cloud.java   Manjunah   30/09/2018 33:19:29   30/09/2018 31:19:39   30166d30f3bcf7168133b6b0eba3cd82b671bc   [B9146bcdb   Xas     tmksmanju   Cloud.java   Manjunah   30/09/2018 30/09/2018   20152733716a7c0aa920013c4660b79a2dc22d   [B9145bcdb   Xas	User Name	File Name Req	Owner Name					Status	
tmksmanju   Cloud.java   Manjunath   30/09/2018 13:19:29   30/09/2018 13:19:39   30/09/2018   30/09/2018   30/09/2018   30/09/2018   30/09/2018   30/09/2018   25733371637c0as9200104560059a;82de224   [189146b6db   Yes	tmkamanju	ToyExample.txt	Manjunath	30/09/2018 13:17:48	30/09/2018 13:18:06	-1ace7be87e4dc03b0c0e8ef65bf892c1211a245b	[B0dd5681	Yes	
TmFemaniu 3tt.im Baigh 30/09/2018 30/09/2018 2557733716a7c0as92a073c466606593a92de224 [R81a773a] Yee	tmksmanju	Cloud.java	Manjunath	30/09/2018 13:19:29	30/09/2018 13:19:38	38166df30f3bcf7168133b6b80eba3cd82b671be	[B0146b6db	Yes	
13:26:25 13:26:35 14:26:04 14:	tmkamanju	Att.jap	Rajesh	30/09/2018 13:26:25	30/09/2018 13:26:35	2£52733716a7c0aa92a073cd46e06b9ac82de224	[B01e779a1	Yes	
raj sample.txt dizesh 14/06/2024 14/06/2024 -58f2e42cb65d6741f4552ea76d3bc7376654e599 [B01f68272 Yea	raj	sample.txt	dinesh	14/06/2024 14:56:32	14/06/2024 14:57:04	-58f2e42cb65d6741f4592ea76d3bc7376654e599	[B@1f68272	Yes	

1803

# UPLOAD Main Menu Data Ower Man Lig Ont Data Ower Menu Urada Wer My Files Wer My Files Dete File Wer My File Wer My Files Wer My Hy My My

## Fig: 5.11

# 5.2.12 DATA UPLOAD SUCCESFUL

iStock by Getty Images*	iStock by Getty Images*	iStock by Getty Images
Search our ste:	Upload File	
Main Menu		
Data Owner Main Log Out	Data Uploaded Successful	ly !!!
Data Owner Menu	BACK	
Upload		
View My Files View My Profile		
Verify		
Delete File		

Fig:5.12

5.2.11

#### **5.2.13 VIEW MY FILE**

DCK Images"	iStoc by Getty Ima	es is	tock		iStc by Getty 1	DCK Images"	
CLOU	JD CON	IPUTING			6	CLC	DUD
iSto	ock	iStock		Stoc	K	iS	tock
by Getty I	lmages"	by Getty Images"	р	y Getty Image	es"	by G	etty Images"
by Getty I	Images"	by Getty Images"	bj	y Getty Image	es"	by G	etty Images"
by Getty I	e	by Getty Images"	b	y Getty Image	ies"	by G	etty Images"
by Getty L View My File Owner Image	e Owner Name	by Getty Images" E-Mail	Mobile	y Getty Image	DOB	by G Location	etty Images" Status

## Fig: 5.13 5.2.14 VERIFY FILE



B	Service(PaaS)
Search our ste: Q	Search for String Identification
End User Menu	File ID File Name Rank Download
User Main	9 rekha.bd 0 <u>rekha.bd</u>
Log Out	
	Back
User Menu	
View My Profile	
View Cloud Files	
Request Sk	
View File Response	
Download	

## 4.2.15 SEARCH FOR STRING IDEMTIFICATION

## Fig: 5.15

Fig: 5.16

## 5.2.16 REQUEST SECRETKEY & PERMISSION

	Platform as a Service(PaaS) Infrastructure as a Service(IaaS).
Search our ster	Request Secret Key & Permission
User Main Log Out	Hi Mr.surekha ur request sent to Cloud Audit Server
User Menu	Back
View My Profile	
View Cloud Files	
Request Sk	
View File Response	
Jownood	

## 5.2.17 VIEW USER REQQUEST & PERMIT

View U	lser Reque	st & Per	rmit		Infrastructure as a Service(IaaS).	/		
User Name	File Name Req	Owner Name	Req Date	Res Date	MAC	Sk	Status	
tmkemanju	ToyExample.txt	Manjunath	30/09/2018 13:17:48	30/09/2018 13:18:06	-1ace7be87e4dc03b0c0e8ef65bf892c1211a245b	[B@dd5681	Yes	
tmkemanju	Cloud.java	Manjunath	30/09/2018 13:19:29	30/09/2018 13:19:38	38166df30f3bcf7168133b6b80eba3cd82b671be	[B0146b6db	Yes	
tmksmanju	Att.jsp	Rajesh	30/09/2018 13:26:25	30/09/2018 13:26:35	2£52733716a7c0aa92a073cd46e06b9ac82de224	[B01e779a1	Yes	
raj	sample.txt	dinesh	14/06/2024 14:56:32	14/06/2024 14:57:04	-58f2e42cb65d6741f4592ea76d3bc7376654e599	[B01f68272	Yes	
surekha	rekha.txt	surekha	04/07/2024 22:09:33	04/07/2024 22:11:16	7a74c89c8305bc00ed5a279395£63£3£423a8ee9	[B@c3e82b	<u>Yes</u>	
<u>Go Back</u>								

## Fig: 5.17

## 5.2.18 VIEW USER REQUEST STATUS OF FILES

								storage, 8 hain, light
						Infrastructure as a		
		2			E.	Service(IaaS).		
						Service(IaaS).		1
View	Request	Status o	f Files			Service(IaaS).		1
View	Request	Status o	f Files			Service(IaaS).		
View	Request File Name Req	Status o	f Files Req Date	Res Date	MA	Service(IaaS).	Sk	Status
View User Name	Request File Name Req a rekha.txt	Status o Owner Name suzekha	f Files Req Date 04/07/2024 22:09:33	Res Date	7a74c89c8305bc00ed5a2	Service(IaaS).	Sk [B8c3e02b	Status Yes

Fig: 5.18

#### **5.2.19 DOWNLOAD FILES**

6		Platform as a Service(PaaS)	
Search our ste: Q Down	load Files		
User Main	Enter File Name :-	rekha.txt	
Log Out	Enter Owner Name :-	surekha	
Data Owner Menu View My Profile	MAC :-	7a74c89c8305bc00ed5a279395f63f3f423a8ee9	
View Cloud Files	Secret Key	[B@c3e82b	
Request Sk	•-		
View File Response			
Download		Download	

## Fig: 5.19

#### **5.2.20 DOWNLOAD FILES**

Chambra	
Search our ste:	Download Files
End User Menu	
User Main	File Contents
Log Out	hello
User Menu	
View My Profile	
View Cloud Files	
Request Sk	
Download	
	Download

Fig:5.20

# 6. CONCLUSION AND FUTURE WORK CONCLUSION

With the increasing number of documents stored in cloud, searching for the desired document can be a difficult and resource intensive task. One solution may be to use symmetric searchable encryption (SSE) which allows one party to outsource the storage of its data to another party (a cloud) privately while enabling to search selectively over it. In this paper we revisited the security definitions of [12] and proposed a new lightweight SSE scheme IIs, s for string search. We have shown that our scheme is secure under the non-adaptive indistinguishability definition [12]. For active adversary, we propose modification of the scheme IIs, s at the additional cost of memory at client's end and two rounds of communications for one modification of document collection.

## 7. REFRENCES

[1] https://github.com/iskana/pbwt-sec/tree/master/sample dat.

[2] <http://www.fon.hum.uva.nl/david/ma ssp/2007/timit/train/dr5/fsdc0/>.

[3] Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable Encryption Revisited: Consistency Properties, Relation to Anonymous IBE, and Extensions. volume 21, pages 350–391. Springer, 2008.

[4] Mihir Bellare, Alexandra Boldyreva, and Adam ONeill. Deterministic and Efficiently Searchable Encryption. In Annual International Cryptology Conference, pages 535–552. Springer, 2007.

[5] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public Key Encryption with Keyword Search. In International Conference on the Theory and Applications of Cryptographic Techniques, pages 506–522. Springer, 2004.

[6] DanBoneh,EyalKushilevitz,RafailOstrovsky,andWilliamESkeithIII. Public Key Encryption That Allows PIR Queries. In Annual International Cryptology Conference, pages 50–67. Springer, 2007.

[7] Ning Cao, Cong Wang, Ming Li, Kui Ren, and Wenjing Lou. PrivacyPreserving Multi-Keyword Ranked Search Over Encrypted Cloud Data. volume 25, pages 222–233. IEEE, 2014.

[8] David Cash, Paul Grubbs, Jason Perry, and Thomas Ristenpart. LeakageAbuse Attacks Against Searchable Encryption. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pages 668–679. ACM, 2015.

[9] David Cash, Joseph Jaeger, Stanislaw Jarecki, Charanjit S Jutla, Hugo Krawczyk, Marcel-Catalin Rosu, and Michael Steiner. Dynamic Searchable Encryption in Very-Large Databases: Data Structures and Implementation. volume 2014, page 853. Citeseer, 2014. [10] David Cash, Stanislaw Jarecki, Charanjit Jutla, Hugo Krawczyk, MarcelC<sup>\*</sup> at<sup>\*</sup>alin Ros,u, and Michael Steiner. Highly-Scalable Searchable Symmetric Encryption With Support for Boolean Queries. In Advances in Cryptology–CRYPTO 2013, pages 353–373. Springer, 2013.

[11] David Cash and Stefano Tessaro. The Locality of Searchable Symmetric Encryption. In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pages 351– 368. Springer, 2014.

[12] Reza Curtmola, Juan Garay, Seny Kamara, and Rafail Ostrovsky. Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions. volume 19, pages 895–934. IOS Press, 2011.