Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 75 (2014) 27996-28000

ISSN: 2229-712X

A novel approach for evaluation of message digest used in data possession

Vibhakar Pathak[1] and Sneha Sharma[2]

[1]Computer Science Engineering Department, Arya College of Engineering and IT, Kukus Jaipur, India.

[2]Computer Science Engineering Department, Poornima University Jaipur, India.

ARTICLE INFO

Article history:

Received: 22 August 2014;

Received in revised form:

15 October 2014;

Accepted: 29 October 2014;

Keywords

Provable Data Possession,

Message Digest Algorithm,

File Formats.

ABSTRACT

In cloud storage, the server that stores the client's data is not trusted. Users would like to check if their data has been tampered or deleted. Provable data possession is used for security by read the file for hash and later specifies its location to verify by applying message digest algorithm which results in generating hash code. Generated hash code is compare with the other hash code generated on different storage. Results showed that precompressed data's that is the data which is designed by the precompressed technique has their difference in message digest present at local connection side and secure connection side at both secure and insecure server.

Introduction

Cloud storage systems are more available and becoming cheaper. Users would like to check if their data has been tampered with or deleted. Provable data possession is a technique for ensuring the integrity of data in outsourcing storage service [1][15][25][26].

Provable Data Possession (PDP) Model

In this the data is preprocessed by the client, and for verification purposes metadata is produced. The file is then sent to an un-trusted server for storage, and the client may delete the local copy of the file. The client keeps some information to check server's responses later. The server proves the data has not been tempered or deleted by responding to challenges sent by the client [15].

Overview of hash functions:

A function that maps a large message into a message digest of fixed small size is known as a hash function. The input to a hash function is called as a 'message' or the 'plain text' and the output is referred as 'message digest' or the 'hash value'. The message digest serve as representative image of an input string and can be used for uniquely identifiable with that string.

(i)Message Digest (MD) Algorithm: MD5 message digest algorithm is used in cryptographic hash function that producing a 128-bit (16-byte) hash value, typically expressed in text format as a 32 digit hexadecimal number. MD5 has been used in a wide variety of cryptographic applications, and is also commonly used to verify data integrity.

(ii)Secure Hash Algorithm: SHA1 stands for "Secure hashing algorithm". It is designed by the United States national security agency. SHA1 outputs a 160-bit digest of any sized file or input. In construction it is similar to previous MD4 and MD5 hash functions.

Data Security & Integrity

(i)Using Cryptographic Algorithm: Cryptography is an algorithm or a technique to encrypt and decrypt information. Once the information has been encrypted, it can be stored on an insecure media or transmitted on an insecure network (like the Internet so that it cannot be read by anyone except the intended recipient.

(ii)Using Hash Function: The message digest should serve as representative image of an input string and can be used as an uniquely identifiable with that string. If any portion of the data is modified, a different hash will be generated.

(iii)Using MAC (Message Authentication Code): MAC operation uses a secret key and cipher algorithm to produce a value which later can be used to ensure the data has not been modified. MAC is appended to the end of a transmitted message. The receiver of the message uses the same MAC key, and algorithm as the sender to reproduce the MAC. If the receiver's MAC matches the MAC sent with the message, the data has not been altered.

(iv)Using HMAC (Hash MAC): HMAC operation uses a cryptographic hash function and a secret shared key to produce an authentication value.

Related Work

Many types of solutions have been proposed, such as

Cooperative provable data possession:

It is based on homomorphic verifiable response and hash index hierarchy to prove the security of scheme based on multi-prover zero-knowledge proof system, which can satisfy completeness, knowledge soundness, and zero-knowledge properties. It allows anyone, not only the owner, to challenge the server for data possession. It is a lightweight PDP scheme based on cryptographic hash function and symmetric key encryption, but the servers can deceive the owners by using previous metadata or responses due to the lack of randomness in the challenges [25].

Designated verifier provable data possession

Designated verifier provable data possession approach is used when the client cannot perform the remote data possession checking. This approach removed expensive bilinear computing. DV-PDP scheme is provable secure and high efficiency in terms of designated verifier provable data possession .This approach provides authorized verification on remote data and enables a designated trusted third party to check data integrity under data owner's permission [26].

Dynamic provable data possession:

In Provable data possession (PDP) model client preprocesses the data and sends it to an un-trusted server for

Tele:

E-mail addresses: vibhakarp@rediffmail.com

storage, while keeping a small amount of meta-data for verification purpose. The client later asks the server by sending challenge that the stored data has not been tampered with or deleted. These schemes provide probabilistic guarantees of possession, where the client checks a random subset of stored blocks with each challenge. Dynamic provable data possession (DPDP) model extends the PDP model to support provable updates on the stored data. Dynamic provable data possession follows some phases as key generated, prepare update, perform update, verify update, challenge, prove and verify to check that the data is secured or not. The advantage is that DPDP scheme is efficient and practical for use in distributed applications [15].

Proof of data retrivability:

Outsourcing of data means that the data owner (client) moves its data to a third-party provider (server) which is supposed to store the data and make it available to the owner and others on demand. This features of outsourcing reduced costs from savings in storage, increased availability of data.PDP technique based entirely on symmetric key cryptography, while not requiring any bulk encryption. It allows outsourcing of dynamic data and efficiently supports operations, such as block modification, deletion and append. The problem of Provable Data Possession (PDP) is also sometimes referred as Proof of Data Retrivability (POR). The central goal in PDP is to allow a client to efficiently, frequently and securely verify that a server who stores client's large amount of data is not cheating the client. The problem is further complicated if the client might be a small device with limited CPU, battery power and communication facilities. To solve this public-key-based technique allowing any verifier to query the server and obtain an interactive proof of data possession. This property is called public verifiability. The advantage of this technique is that it allows efficiently and securely verifying the data. The limitation is that it is based upon symmetric key cryptography which is unsuitable for public (third party) verification [1].

Proposed System

In literature survey, I have read many papers on data security in cloud computing. The researchers had presented many techniques and methodology to secure the data but I am not found any analysis of the data possession methodology applicable on various types of files on local, secure and public cloud. So I have undertaken this objective.

System Design

The design specification flow diagram provides the total conceptual diagram used for experimenting and simulating the result. Following are the steps of designing the targeted work:

- Hash files are read.
- To verify the file location is selected for storage.
- Calculate hash files to verify and performed bit padding, append length operations on the file.
- Initialize the message digest algorithm like MD5 and SHA (secure hash algorithm) to perform.
- Process message in 16 word block and find its output.
- Last phase is comparison.
- Comparison done on hashes for hash file and calculated hash to check that the hash files are matched or not for security.

Proposed Method

Proposed techniques are used for analysis of message digest algorithm on various cloud storage formats of file for checking the integrity of the data by generating and comparing hash code. [1][15][25][26].

KeyGen: (sk, pk) is an algorithm run by the client. It takes as input a security parameter, and outputs a secret key sk and a public key pk. The client stores the secret and public keys, and sends the public key to the server.

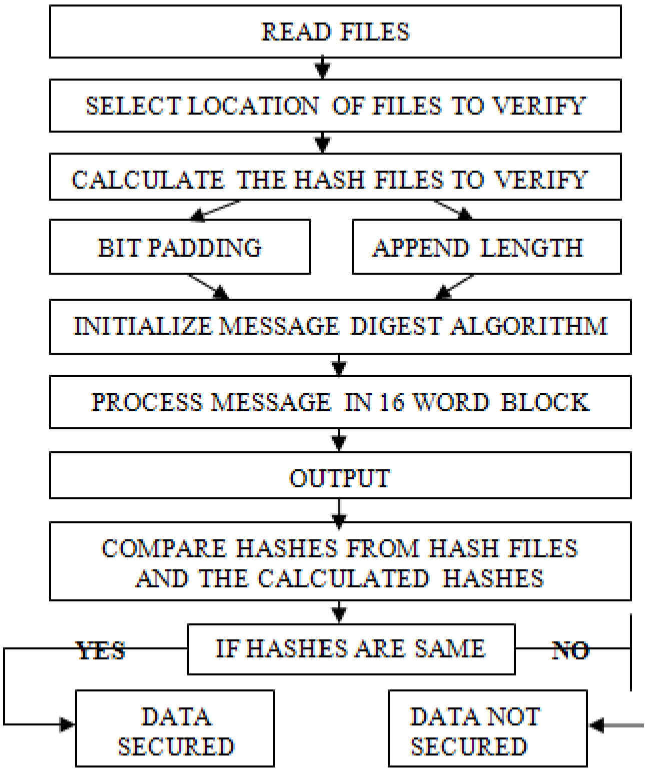
READ FILES
SELECT LOCATION OF FILES TO VERIFY
CALCULATE THE HASH FILES TO VERIFY
BIT PADDING
APPEND LENGTH
INITIALIZE MESSAGE DIGEST ALGORITHM
PROCESS MESSAGE IN 16 WORD BLOCK
OUTPUT
COMPARE HASHES FROM HASH FILES AND THE CALCULATED HASHES
YES
IF HASHES ARE SAME
NO
DATA SECURED
DATA NOT SECURED

Figure 2: System design of targeted work

Prepare Update: (sk, pk, F, info, Mc) is an algorithm run by the client to prepare (a part of) the file for un-trusted storage. As input, it takes secret and public keys. The file is defined by F with the info of the update to be performed and the previous metadata Mc. The output is an "encoded" version of the file e (F) along with the information e (info) about the update, and the new metadata e (M). The client sends e (F), e (info), e (M) to the server.

Perform Update: (pk, Fi−1, Mi−1, e (F), e (info), e (M)) is an algorithm run by the server in response to an update request from the client. The input contains the public key pk, the previous version of the file Fi−1, the metadata Mi−1 and the client-provided values e (F), e (info), e (M). The output is the hash code along with the information of new version of the file Fi and the metadata Mi.

Challenge (sk, pk, Mc): {c} is a probabilistic procedure run by the client to create a challenge for the server. It takes the secret and public keys, along with the latest client metadata Mc as input, and outputs a challenge c that is then sent to the server.

Prove (pk, Fi, Mi, c) : {P} is the procedure run by the server upon challenge from the client. It takes as input the public key, the latest version of the file and the metadata, and the challenge c.

Verify (sk, pk, Mc, c, P) **:** {accept, reject} is the procedure run by the client upon receipt of the proof P from the server. It takes as input the secret and public keys, the client metadata Mc, the challenge c, and the proof P sent by the server. An output of accept ideally means that the server still has the file intact.

Platform Used For Experimentations

Cloud Simulation Tool: It is a new generalized and extensible simulation framework that allows seamless modeling, simulation, and experimentation of Cloud computing infrastructures and application services. It is a tool (library) for cloud computing simulation written in Java language.

Table 1: Comparative Table of Various Data Possession Techniques

Scheme	Algorithm	Single/ multi cloud	Merits	Demerits
PDP (Provable Data Possession)	PDP	Multi Cloud	It provides security to data and public verifiability based on RSA scheme.	Applicable for only static file. Insecure against dynamic block of data
SPDP (Scalable PDP)	PDP	Single Cloud	It provides efficient PDP by encryption andit is light weight PDP scheme to support homomorphic hash function	By using the previous challenges, client can deceive the server leads to lacks in randomness.
DPDP-I (Dynamic PDP –I)	Authenticated Skip List	Single Cloud I	Block modification and updation of block is allowed.	Construction of rank based scheme is difficult.
DPDP-II (Dynamic PDP –II)	RSA trees	Single Cloud	Blockless verification can be queried for integrity verification. RSA trees use homomorphic tag where tag are small and easy to use	DPDP scheme with RSA tree construction is efficient with dynamic option but it cannot be adapt to multi-cloud.
POR (Proof of retrievability)	MAC	Multi Cloud	Preprocessing steps can be made by client before storing their data. And it is the simple way to audit the server.	It is difficult to build the system.
Cooperative Provable Data Possession	PDP	Single Cloud	This Scheme can satisfy completeness, knowledge soundness, and zero-knowledge properties.	This scheme focuses on data possession issues at an un-trusted servers in a single cloud providers. It is not suitable for multi cloud environment.

Table 2: Experiment Result for Input Image Formats

Algorithm	Type	Storage	Output
MD5	Image Image (zip) Image (rar)	Local Cloud	57DD62562EF06BE543CE95837DF03181 88E76F2472B38C00D3A2323B87F59175 172AF6D003B0F0DC2E9E822F3DAABEE9
	Pdf Pdf (zip) Pdf (rar)		5BF3D43FE8E66AE0B478F37143CC0B10 837A6C9527B7AA0DD56D16C9906719A1 915CC58A60C4A7F30FD34FBB8D0947E5
SHA1	Image Image (zip) Image (rar)	Local Cloud	8B65FC296DDE9099039EB2641918222C085FD81B 092580F585A08839E2FD3CE0F23AFFE713172065 850D70091C73168AE517F44A51504A3AB0FB9F85
	Pdf Pdf (zip) Pdf (rar)		D6C6BE9E5CB86FBB0D30C82AD6B73319EDDDA4E7 AB201A59F175A9F5FEA85C1E1CA1CE01A734B5DC 4E61AFF1AF6825F711C7765906AD0498242447E98
MD5	Image Image (zip) Image (rar)	Mail attached Cloud	ACADA417AFDF269345058F373B293917 ACADA417AFDF269345058F373B293917 06CAEA68632DCC9992AF547F14B94DDE
	Pdf Pdf (zip) Pdf (rar)		8BE2D848D48C79C32E1331E7345979E8 1ABD9DE3A462ED67A9785C54E85B1DB6 52ED0825B48A17FDE4096C957E0C6287
SHA1	Image Image (zip) Image (rar)	Mail attached Cloud	C5CF802F4430076C4E957582FB960ACSC487314D 884A5FDB831C16DC8F597E5DB126DEA8BE328949 6BEA1DC28CD217C3190D9099E8C9E53AF00AF951
	Pdf Pdf (zip) Pdf (rar)		3FE32240B918231ADB27697C27649D239CD5CB26 B5ADAEAC9829FEEF4CB017A5517EEE75BDC2C073 4A79B99C608F8079DFA75F9F2A52DABE294A433A
MD5	Image Image (zip) Image (rar)	Google Drive	ACADA417AFDF269345058F373B293917 63008A3A232C67835FB5662D433A6080 06CAEA68632DCC9992AF547F14B94DDE
	Pdf Pdf (zip) Pdf (rar)		8BE2D848D48C79C32E1331E7345979E8 52ED0825B48A17FDE4096C957E0C6287 1ABD9DE3A462ED67A9785C54E85B1DB6
SHA1	Image Image (zip) Image (rar)	Google Drive	C5CF802F4430076C4E957582FB960ACSC487314D 884A5FDB831C16DC8F597E5DB126DEA8BE328949 6BEA1DC28CD217C3190 D9099E8C9E53AF00AF951
	Pdf Pdf (zip) Pdf (rar)		4E10E63190ADE3A30B25D7A029702C547B3252 8D72DEBC30A66121AF95296A709FBF448004A01 4DBCA7DF3740E27DFAEFE7CCCCC22189B8888EEB

Netbeans Integrated Development Environment: The NetBeans IDE is open source and is written in the Java programming language. It provides the services common to creating desktop applications such as window and menu management, settings storage. The NetBeans platform and IDE are free for commercial and non-commercial use.

Java: Java is a computer programming language that is class-based, object-oriented. Java applications are typically compiled to byte code (class file) that can run on any Java virtual machine (JVM). Java is one of the most popular programming languages in use, particularly for client-server web applications. Java was originally developed by James Gosling at Sun Microsystems and released in 1995 as a core component of Sun Microsystems' Java platform. One characteristic of Java is portability, which means that computer programs written in the Java language must run similarly on any hardware/operating-system platform.

Experimental results and discussion

This section deals with the details of initial experimentation carried out, the experimental scenario considered, data assumed, the platform used along with the results obtained.

Experimental Scenario:

Experiment 1: Analysis of Data Possession methodology applicable on various format of file on public cloud, secure cloud and local cloud.

Case 1:

(a)When applying MD5 algorithm on an image file with different file formats on storage like mail attached cloud and Google drive cloud, same hash code is generated. It means that the data sent by the client and received by the server is secure. On the other hand data at local cloud different hash code is generated so the data is not secured.

(b)When applying SHA1 algorithm on an image file with different file formats on storage like mail attached cloud and Google drive cloud, same hash code is generated. It means that the data sent by the client and received by the server is secure. On the other hand data at local cloud different hash code is generated so the data is not secured

Results and Discussion

Analyze the Data Possession methodology applicable on various format of file on public cloud, secure cloud and local cloud. The parameters such as various file formats and message digest algorithm were studied. It has been found that the data stored on a cloud has to be verified by a message digest algorithm at local cloud, public cloud and secure cloud where file and the server has been created. On comparing the data, we found that precompressed data's that is the data which is designed by the precompressed technique like jpeg, mpeg, pdf, txt has their difference in message digest present at local connection side and secure connection side at both secure and insecure server. This has been because the file is precompressed and message digest has been recomputed and recomputed which is going to give difference hash code.

Conclusion

The security and privacy issues are significant obstacles towards the cloud storage. With the emergence of cloud storage services, data integrity has become one of the most important challenges. Earlier works have shown various provable data possession methodologies. But we have not found any analysis on the data possession methodology applicable on various types of files on local cloud, secure cloud and public cloud. In cloud storage systems, the server that stores the client's data is not necessarily trusted. Provable data possession (PDP) model is used to check that data has been tampered with or deleted.

Acknowledgement

I would like to express my deep gratitude and thanks to Dr. Vibhakar Pathak Computer Science Engineering Department, Arya College of Engineering and IT, Kukus Jaipur, India for giving me an opportunity to work under his guidance and also for his consistent motivation & direction in this regard.

References

[1] Giuseppe Ateniese, Roberto Di Pietro, Luigi V. Mancini, Gene Tsudik, "Scalable and Efficient Provable Data Possession", Published in International Journal of Advanced Trends in Computer Science and Engineering, 2008.

[2] Kai Hwang, Yue Hu ," Cloud Security with Virtualized Defense and Reputation-based Trust Management", Published in Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009.

[3] Yuesheng Tan, Jingyu Wang,"CC-VIT: Virtualization Intrusion Tolerance Based on Cloud Computing", Published in IEEE, 2010.

[4] M.R.Tribhuvan,V.A bhuyar, "Ensuring data storage security in cloud computing through two-way handshake based on token management", Published in International Conference on Advances in Recent Technologies in Communication and Computing,2010

[5] Dzmitry Kliazovich, Yury Audzevich, Samee Ullah Khan ," GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers", Published in IEEE,2010

[6] Ian Gorton, Yan Liu, Jian Yin, "Exploring architecture options for a Federated, Cloud-based System Knowledgebase", Published in 2nd IEEE International Conference on Cloud Computing Technology and Science,2010

[7] Antonio Celesti, Francesco Tusa, Massimo Villari, Antonio Puliafito, "Three-Phase Cross-Cloud Federation Model: The Cloud SSO Authentication", Published in Second International Conference on Advances in Future Internet, 2010

[8] Zhitao Wan," A Network Virtualization Approach in Many-core Processor Based Cloud Computing Environment", Published in Third International Conference on Computational Intelligence, Communication Systems and Networks, 2011

[9] Mohammed A, AlZain, Ben Soh and Eric Pardede," MCDB: Using Multi-Clouds to Ensure Security in Cloud Computing", Published in Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2011

[10] Mohemed Almorsy, John Grundy and Amani S. Ibrahim," Collaboration-Based Cloud Computing Security Management Framework", Published in IEEE International Conference on Cloud Computing , 2011

[11] Karin Bernsmed, Martin Gilje Jaatun, Per Hakon Meland, Astrid Undheimy ," Security SLAs for Federated Cloud Services", Published in Sixth International Conference on Availability, Reliability and Security, 2011

[12] John Buford, Kshiteej Mahajan, Venkatesh Krishnaswamy," Federated Enterprise and Cloud-based Collaboration Services", IEEE, 2011

[13] Deyan Chen, Hong Zhao,"Data Security and Privacy Protection Issues in Cloud Computing", Published in International Conference on Computer Science and Electronics Engineering,2012

[14] Liuyang WANG, Yangxin YU, Huai ZHOU," Design of Intelligence Multi-agent for Virtualization Resource in Cloud Computing", Published in International Conference on Computer Science and Service System, 2012

[15]C. Chris Erway, Charalampos Papamanthou, Alptekin Kupcu, Roberto Tamassia, "Dynamic Provable Data Possession", Published in IEEE,2012

[16] Zhang Xin, Lai Song-qing , Liu Nai-wen,” Research on Cloud Computing Data Security Model Based on Multi-dimension”, Published in International Symposium On Information Technology In Medicine And Education, 2012

[17]Veerraju Gampala, Srilakshmi Inuganti,Satish Muppidi, ”Data Security in Cloud Computing with Elliptic Curve Cryptography”, published in International Journal of Soft Computing and Engineering (IJSCE), 2012

[18] Yahya Al-Hazmi, Konrad Campowskyy and Thomas Magedanz,” A Monitoring System Federated Clouds”, Published in IEEE 1st International Conference on cloud Networking, 2012

[19] Zhiyong Xu, Wansheng Kang, Ruixuan Li, KinChoong Yow, and Cheng- Zhong Xu,” Efficient Multi-Keyword Ranked Query on Encrypted Data in the Cloud”, Published in IEEE 18th International Conference on Parallel and Distributed Systems, 2012

[20] Jun Xu, Weiming Zhang, Ce Yang, Jiajia Xu, Nenghai Yu,” Two-Step-Ranking Secure Multi-Keyword Search Over Encrypted Cloud Data”, Published in International Conference on Cloud Computing and Service Computing, 2012

[21] Maicon Stihler, Altair Olivo Santin, Arlindo L. Marcon Jr, “Integral Federated Identity Management for Cloud Computing”, Published in IEEE, 2012

[22] Sushmita Ruj, Milos Stojmenovic, Amiya Nayak ” Privacy Preserving Access Control with Authentication for Securing Data in Clouds”, Published in 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012

[23] G.Rakesh Reddy, Dr. M. Bal Raju, “Augmentation Data Security Aspects of Cloud Computing”, Published in International Journal of Advanced Trends in Computer Science and Engineering, 2013

[24] Paridhi Singhal, Alok Garg, Manoj Diwakar” Security in Cloud Computing- Hash Function”, Published in International Journal of Computer Applications, 2013

[25] O. Rahamathunisa Begam1, T. Manjula2, T. Bharath Manohar3, B. Susrutha,“ Cooperative Schedule Data Possession for Integrity Verification in Multi-Cloud Storage”, International Journal of Modern Engineering Research Vol. 3, Issue. 5, Sep - Oct. 2013 pp-2726-2741,2013

[26] Yongjun Ren1,2, Jiang Xu1, Jin Wang1 and Jeong-Uk Kim3, “Designated-Verifier Provable Data Possession in Public Cloud Storage”, International Journal of Security and Its Applications Vol.7, No.6 (2013), pp.11-20, 2013