Arithmetic Soundness of Deduplication Security Frameworks

Gagandeep Kaur

Research Scholar

Chandigarh University (Mohali), India

Abstract- The incomplete theory suggest that in real world conditions, contracts cannot specify what is to be done in every possible contingency. A contact may be a simple economic tie up or maybe a contacts between multiple software components .Future contingencies may not even be describable when the contact or software is been build . Same is true, when security frameworks are designed for Deduplication. Not all breaches in security structure of Deduplication System can be defined beforehand. However, the concept of provable security suggests that the adversary model and security model must be defined beforehand only then the security can be implemented. In this research work, we discuss how the axioms and postulates change and expand to satisfy new technical scenarios. As the expansion occurs the system can handle more security issues but at each stage there is a need to check the completeness, arithmetic soundness or arithmetic validity of the system in question. The computations of the checking the arithmetic validity and Proof of ownership schemes take some degree of overhead the work attempts to compute over head in computing the validity of provable security system of Deduplication in multiple Deduplication security scenario.

Keywords- Deduplication, Encryption, Arithmetic validity, Proof of ownership, Key Generation, Cloud Computing

I. INTRODUCTION

It has been found in the contemporary literature survey that there is no best Deduplication solution. It depends on the business and technical scenarios. In addition, it is self-evident that without the use of proof of ownership [1] schemes and key validity schemes it would be hard to run Deduplication Systems securely. This is due to the fact, that the conditions, assumptions change with time. Modular arithmetic is normally used to create groups of keys, rings and fields that are fundamental building blocks of most modern public-key cryptosystems for securing the computer files and systems. The reason is that modular arithmetic gives a chance to increase difficulty in guessing the keys if we introduce modular reduction methods in key management or proof of ownership schemes. The key management schemes include operations such as Key Generation, Key Distribution, Key size, Key Memory management, Key hardness, Key ownership proof workflows and many more operations that are designed based on the formal mathematical equations. These mathematical equations need to arithmetically sound. Which means in context of our work is soundness with respect to Key operations that make the system safe for transferring data from source to the target device for Deduplication. A fully automated backup and Deduplication system can be considered as mathematical logical systems that have order and logic in their workflow. Therefore, all the Deduplication system have inherently a soundness property that needs to be checked for it to be successful in real context.

The soundness of a logical system will work correctly only if the rules based on which it is build are valid with respect to its semantics. In simple words, this implies that if the semantics or the arrangements of things governing the Deduplication system under go some change the soundness of the operations, methods and mathematics will also undergo change. This would lead to unsecure and incomplete system. Which means that in all possible conditions, the mathematical rules must maintain the truth or valid conditions feedback from the current industry gives us hint that new forms of the attacks keep on challenge live to maintain, the security of the Deduplication System. Consequently, the semantics of the application may change and every time it must undergo the check to find the completeness and the soundness of the system.

The mathematical properties (soundness, completeness, arithmetic validity) helps to understand the usage of modular and formal logic in constructing the mathematical functionality of keys. Which would help to secure any system that may be Deduplication for example. But all these efforts may turn out to be worthless in cases we are unable to build a system that is not secure our privacy, integrity of data and safe guard our credentials in cross user environments. Feedback from the industry and contemporary literature shows that Deduplication application suffers from multiple attacks such as Insider attack, memory attack, side channel attacks etc. As the iterative progression is going on in context of Cloud based Deduplication systems .and at the same time the potential of such attacks is increasing because new and novel secure solutions are getting obsolete, as they do not incorporate new axioms, postulates and conditions. The security of a distributed Deduplication application starts from the user management and authorizations and it has

three levels at which security is required to be fortified of the Deduplication system [Figure 1]. The Figure 1 shows the levels at which security algorithms need to be implemented after the user has entered the cloud system. Nevertheless, a security process, as we know begins at the fuzzy boundary of the cloud itself and the system must be fortified with a fine grain approach. The next sections discuss the evolutions of the solutions that will be useful building a secure system.

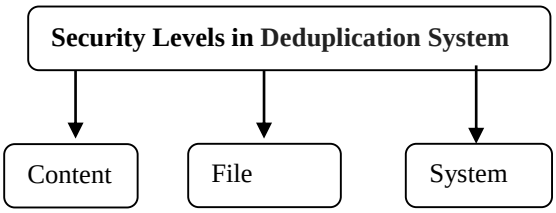


Figure 1. Level of Security of Deduplication.

Evolution & Implementation of Solutions for securing Deduplication Systems:-

Beyond user name and password management systems. Today, the need of the hour is to build multi-step security systems or system that check vulnerable points that each point of life cycle of system. Moreover, it is was desired that an arrangement beyond the Key Management having addition security features is better. One of the solutions that has come long way is making the system secure the use of concept called "provability" . This means that the security of the system is based the adversity model and is clearly defined and can be build using set of rules of "inference" in objective manner. The system should be able securely realize an ideal security functionality, as it build on axioms clearly defined axioms of adversity model. However, in real life, the constructs keep on changings for adversity model and consequently the security model constructs continue to change. Hence, the solution to security would remain in evolutionary process. The Figure 2 shows the basic flow of the Deduplication applications and its security components.

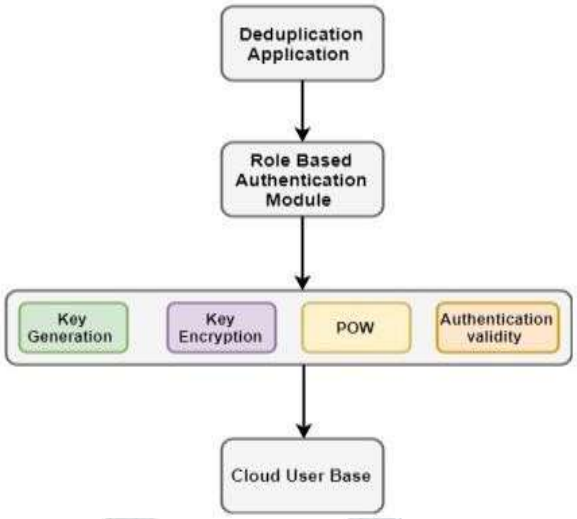


Figure 2. Flow of the Deduplication System

The word implementation is a connotation of a source code .While the implementation may not have flaws (buffer overflow, etc.) that influence security. The protocol design is secure, but the implementation may not be due to reasons discussed in the Problem section. Hence, the realization of the solutions depends on multiple technical conditions and level of security expected from the system. Nevertheless, it can be seen that the main components of the Deduplication System include:

1) **Core Deduplication Application or Algorithm :** It is prone to memory and cross user abuse issue
2) **Role Based User Management:** It is prone weak passwords & password leakage kind of issues that may lead to insider attack and sometimes even a rouge client may get change to peep into the system.
3) **Key Generation & Encryption :** This part of the Deduplication Applications is prone to multiple issues such as Key refreshes , complexity, distribution and storage of Keys and attacks that can guess the Keys (Key tempering , Encryption attacks , Cipher text attacks) etc. . ,
4) **Proof of Ownership (POW):** The attacker can deceive the server to obtain ownership of the file by get the hash value of original file.
5) **Arithmetic Validity /Soundness:** The attacker may be able to observe the pattern and monitor to do intelligent conjectures to estimate and obtain key values and get ownership.
6) **Cloud User:** There might be possibility legitimate user may become malicious insider and create unwanted conditions such

cross Virtual Machine attacks etc.

Therefore, it sufficiently clear that Deduplication is prone to many issues at multiple points. As mentioned in the last paragraph, that if the adversity model changes the axiomatic system of securely needs to changes or it may simply defeat the purpose of securing, for example, if it is assumed that the interaction between the cloud user [5] and cloud storage provides is as simple as logging into the system with user name, then the system can be presented as follows:
These are four parties (p, q, r, s.) involved in the process of securing the 'd' Deduplication system.

d is the Deduplication application 'p' is the cloud service provider

'q' is the cloud user subscription to the cloud storage service from 'p'

Set of rules between

P, q

If 'q' is a authorized user then 'p' allows 'q' to use 'd' application.

Table 1. Truth table of user management and authorization

User	Password	Authorization (Allowed to use Deduplication Function)
T	T	T
F	T	F
T	F	F
F	F	F

With the passage of time the trust level of people using cloud services has declined and uses have been demanding anonymous alternatives as no one want to take risks. The cloud user does not trusts the cloud service provider nor does the cloud service provider trusts the cloud user. To solve this issue a third party is engaged and both the participating parties take a trust worthiness certificate from them and continue with their business of Deduplication. The truth table now need to have additional conditions, assuming that both parties will behave in good sprit.

Table 2. Truth Table with Third Party Verification

User	Password	Third Party Verification	Authorization (allowed to use Deduplication Function)
T	T	T	T
F	T	F	F
T	F	F	F
F	F	F	F

The cloud user 'q' does not trust the cloud service provider 'p'. There is fair chances that the credentials of cloud user 'u' may share or sell to other unwanted elements. If 'q' is authorized user 'p' may use credentials for some other purposes. Hence a third party vertically is introduced to maintain trust 't'. If 'p' gets verified by 't' then cloud service provides is (trust worth) 'e' 'q' is ready to use cloud services. If 'p' is trust worthy, then 'q' may not be trusty. Therefore, hence 'q' needs to get verified. If 'q' does not get verified, then 'p' is not authorized user.

As the cloud services matured. Multiple points in the workflow were found to be prone to the issues of security. A need arises to do security audits either on the fly or periodically for maintaining every ready security. The cloud service provider either involves its own resources to do so or finds another third party to do so for conducting the security audits. Hence, again this solutions tries to overcome the issue of "Quality" of security being implemented. The third Party verifier was able to build the trust between the interacting parties but fortification need to be checked every moment. Hence, now new conditions and new axioms need to design just to enter the Deduplication system.

Table 3. Truth Table with Third Party Verification & Security Audits

User	Password	Third Party Verification	Security Audits	Authorization (allowed to use Deduplication Function)
T	T	T	T	T
F	T	F	F	F
T	F	F	F	F
F	F	F	F	F

Taking the case further, Let's us say - Party A wants to optimize storage space and have taken storage and Deduplication services from cloud services [6] provides as add in its subscription. At the same, the CSP also wants to optimize its distributed storage. The parties A is sharing its file and follow with other cloud users with whom its shares database on the participation by management. It was found that simple username /password [7] system does not suffice the role-based security because now the need is to give permissions of read/write at object level or fine grain level. For this Key management system is introduced. Moreover, the axioms, postulates and conditions change as shown in the table below:

Table 4. Truth Table based on Key interaction between the Parties

Server Key(s) Public/Private	Cloud User Key Public/Private	Third Party Verification using Public /Private Keys	Security Audits	Authorization (allowed to use Deduplication Function)
T	T	T	T	T
F	T	F	F	F
T	F	F	F	F
F	F	F	F	F

The further constructs in pursuance of high degree of security involves enhancing the Deduplication algorithmic process along with Proof of ownership [8] (POW). Security Organizations have shown incidence reports where the Keys are hacked / leaked and then the Deduplication application is at the mercy of the intruder. The axioms of the system and adversity model again have to undergo expansion. Now the server, the cloud user and all other parties need to proof the owner ship of the Key they are using to check into the Deduplication application.

Table 5. Truth Table based on Key interaction & POW between the Parties

Server Key(s) Public/Private	Cloud User Key Public/Private	Third Party Verification using Public /Private Keys	Security Audits	Proof Of Ownership	Authorization (allowed to use Deduplication Function)
T	T	T	T	T	T
F	T	F	F	F	F
T	F	F	F	F	F
F	F	F	F	F	F

II. INFRENCES AND RESULTS

Arithmetic Soundness/Validity of Deduplication System: It is clear from the above constructs /cases of security systems for Deduplication process. The axioms and the postulates with when we start build the system keep on expanding to match the possible adversities that may be involved in running the Deduplication system. We can see the Deduplication System, ultimately comes as system of proofing system of true or false multiple conditions. We are now left with understanding we need to prove anything that's right (completeness) and at the same we cannot prove anything that's wrong (Soundness).

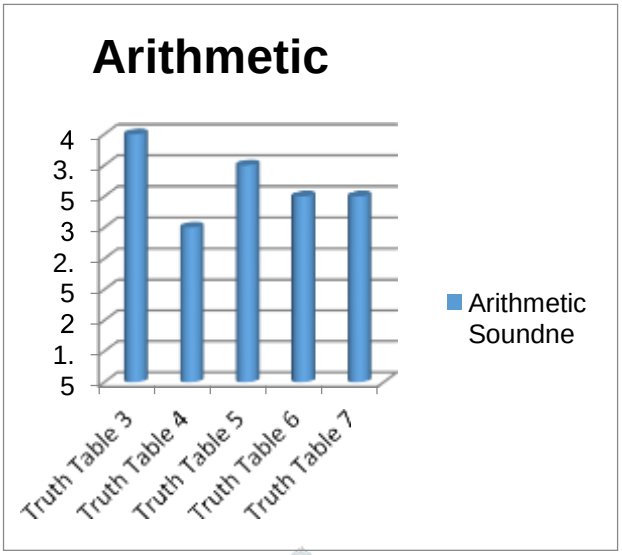
Which also means that of the mathematics (modular arithmetic or cryptographic mathematics) with which the interaction between the parties is done is also required to be check, especially. When the zero knowledge proof algorithms [9] are introduced into the system for providing anonymous credential checking at File /Block levels.

By definition when a formal system for example Deduplication system will be called complete, if the with respect to all the properties or formulae based on which the system has been build can be derived from the system itself. But, we see in almost all the cases the system becomes incomplete and expansion is required. The moment the system is expanded the arithmetic validity or the arithmetic soundness of the systems also becomes questionable.

To validate this conclusion we can take the example of "zero knowledge proof algorithm" in the Deduplication System. It has also been reported that certain observable patterns (numbers) can be deduced from the process of Key Management and POW to arrive at fair guess even if we use modular arithmetic. These rules or mathematical equation will only be valid if and only if it is true under every interpretation (interaction between the parties, data servers and other software components), and an argument (a number representing the Key value) form (or schema) is valid if and only if every argument (Key Value) of that form is valid and does not come from outside the system (Hacker/attacker/intruder).

Therefore, we consider all the cases, the present in the arithmetic soundness of the system in terms of 10-point scale. We shall see as the system becomes complex and bigger. The arithmetic soundness of the increases but it cannot be 100% as certain axioms will be left; change or new axioms and conditions need to add. A 5 point scale was developed to rate the arithmetic soundness based on the number of operations which a system need to proof to become secure and safe. Higher the scoreless it is likely to be complete and sound. The following bar graph emerge. might just happen that the cloud storage user is mindlessly shares the authentication credentials with other people and create problem for her/himself. Then, there might be case, where the data is important for some anti-social elements and cyber criminals target it day in and day out. There looking at the bar graph we can make following observations:

1. As the number of operations increase the completeness and arithmetic soundness decreases as more assumptions and conditions need to be proved and derived. New factors and variables give birth to new axioms and postulates.
2. Authorization/actions true that are valid/true, but may not provable as they work in that logical manner only.
3. Trust cannot necessary be proved, but can deducted or inferred from actions defined by provable mathematical function.
4. Axiomatic system based adversity model or security model may require additional axioms for defining high order of security logic [10].
5. Keys/actions may be provable but may not be arithmetically sound due to intervention of adversities.



III. CONCLUSION

There are many kinds of attacks possible on Deduplication systems. To build a highly secure system, the first steps is to consider the type of axioms which are taking to build the system. These axioms if considered on single adversity model, will be inadequate and will have some degrees of contradictions due to level of resources available for securing the system. The behavior of the client is hard to observe with high precision, especially when the client is an insider and may have access to cloud services may have malicious intent to steal the files content or there might be someone just wasting the resources by adding duplicate files in storage mindlessly

REFRENCES

[1] J. Li, X. Chen, M. Li, J. Li, P. P. Lee and W. Lou, "Secure deduplication with efficient and reliable convergent key management," *IEEE transactions on parallel and distributed systems*, vol. 25, pp. 1615--1625, 2015.

[2] H. Riesel, "Prime numbers and computer methods for factorization," vol. 126, Springer Science \& Business Media, 2012

[3] R. A. Patel, M. Benaissa, N. Powell and S. Boussakta, "Novel power- delay-area-efficient approach to generic modular addition," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, pp. 1279--1292, 2007.

[4] "Repeated Squaring," Wednesday March 2017. [Online]. Available: http://www.algorithmist.com/index.php/Repeated_Squaring. [Accessed Wednesday March 2017].

[5] R. K. Banyal, P. Jain and V. K. Jain, "Multi-factor authentication framework for cloud computing," in *Computational Intelligence, Modelling and Simulation (CIMSim), 2013 Fifth International Conference on*, 2013, pp. 105-110.

[6] C. Wang, K. Ren, W. Lou and J. Li, "Toward publicly auditable secure cloud data storage services," *IEEE network*, vol. 24, 2010.

[7] J. Xu, W. T. Zhu and D. G. Feng, "An improved smart card based password authentication scheme with provable security," *Computer Standards \& Interfaces*, vol. 31, 2009.

[8] L. G.-. Manzano and . A. Orfila, "An efficient confidentiality-preserving Proof of Ownership for deduplication," *Journal of Network and Computer Applications*, vol. 50, no. 1084-8045, pp. 49-59, 2015.

[9] Q. Zheng and S. Xu, "Secure and Efficient Proof of Storage with Deduplication," pp. 1-12, 2012.

[10] D. Geneiatakis, C. Lambrinoudakis, G. Kambourakis, A. Kafkalas and . S. Ehlert, "A first order logic security verification model for SIP," in *Communications, 2009. ICC'09. IEEE International Conference on*, 2009, pp. 1--6.

[11] "Calculating Powers Near a Base Number," Wednesday March 2017. [Online]. Available: http://www.vedicmaths.com/18-calculating-powers-near-a-base-number. [Accessed Wednesday March 2017].

[12] Gagandeep Kaur, Mandeep Singh Devgan, "Data Deduplication Methods: A Review", International Journal of Information Technology and Computer Science(IJITCS), Vol.9, No.10, pp.29-36, 2017. DOI: 10.5815/ijitcs.2017.10.03.

[13] Gagandeep Kaur, Mandeep Singh Devgan, "Identification of Duplicate Chunks Using Content Approach," *International Journal of Computer Sciences and Engineering*, Vol.5, Issue.10, pp.110-117, 2017

[14] Deduplication of Cloud Computing", International Journal of Emerging Technologies and Innovative Research (www.jetir.org), ISSN:2349-5162, Vol.5, Issue 10, page no.564-569, October-2018

[15] Deduplication in Databases using Pattern Matching", International Journal of Emerging Technologies and Innovative Research (www.jetir.org), ISSN:2349-5162, Vol.5, Issue 3, page no.281-283, March-2018