

An Effective Technique to evade the Flood Issue in Network Coding for Distributed Cloud Stockpiling Framework

Mangayarkarasi M¹, Jamunadevi C², Vishnu Priya S³, Hamsanandhini S⁴ & Deepa S⁵

^{1,2,4&5}Assistant Professor, Department of Computer Technology-PG, Kongu Engineering College, India.

³Currently pursuing Master's degree programme in Software Systems, Kongu Engineering College, India.

Email: ymmangai@gmail.com¹, jamu83@gmail.com², svishnupriyaay@gmail.com³, hamsanandhini.ctpg@kongu.edu⁴ & sdeepakec@gmail.com⁵

Received: 14 Feb 2020 Revised and Accepted: 25 March 2020

ABSTRACT: This paper investigates the flood issue of a Network coding stockpiling framework when the encoding specifications and the stockpiling specifications are crisscrossed. The flood issue of the Network Coding Stockpiling framework happens in light of the fact that the network coded encryption generates expanded coded information, bringing about high stockpiling and handling overhead. To stay away from the flood issue, we propose a flood evasion design that takes record of surveillance and stockpiling prerequisites in compressing, encoding, capacity techniques. We give the expository consequences of the most extreme passable put away encoded information under the ideal confidentiality rule. The structure rules to accomplish high coding productivity with the most reduced stockpiling cost are additionally introduced.

KEYWORDS: LZW; Cloud storage; information surveillance; compression.

I. INTRODUCTION

Network coding is appealing for its ability of accomplishing the unequivocal surveillance. On a basic level, organize coding basically blends information from various system hubs dependent on the well designed straight blend rules. For whatever length of time that fractional system coded information are verified, even with limitless registering force and time, a spy can't interpret the whole original information [1]. Another preferred situation of system coding is that no information move limit augmentation happens diverged from the cryptographic technique. As of late, network coding is acquainted with improve the surveillance of dispersed stockpiling in which clients redistribute their information to various clouds [2]. In spite of the fact that offering numerous focal points, distributed stockpiling definitely presents surveillance dangers on the re-appropriated information. In [4], it was indicated that network coding can be utilized to forestall listening stealthily in dispersed distributed storage. In [3], the issue of checking the uprightness of system coded information in a safe distributed storage framework was researched. In any case, from the part of usage, the performance problem of network coding for distributed cloud storage remains open. This propels us to investigate how to for all intents and purposes and cost adequately store coded information in different clouds. When encoding specifications are not

together planned with the stockpiling specifications, a protected network coding stockpiling framework may

experience that the size of put away coded information in the cloud repository can be greater than the size of the original information. This problem happens when system coded information are described in the form of digits this is known as the flood issue right now. The purpose of our work is to build up an efficient plan procedure of a network coded distributed cloud stockpiling framework.

II. PROPOSED SCHEME

In this proposed method, flood evasion design is introduced with the required surveillance and cloud stockpiling. Fig.1 describes the flow of proposed design. The key idea of the flood evasion design is to compress the original data then the compressed data are preprocessed and stored to the multiple cloud stockpiling frameworks. A total encoding strategies and information appropriation conspire are mutually intended for protected distributed stockpiling.

Our proposed design is executed in the four steps.

1. The Original file is compressed using the LZW algorithm.

2. A dynamic-length letters in order portrayal of network coded information is received.
3. The compressed information is splitted and pulled together.
4. The pulled together information are encoded and disseminated to the various cloud stockpiling frameworks.

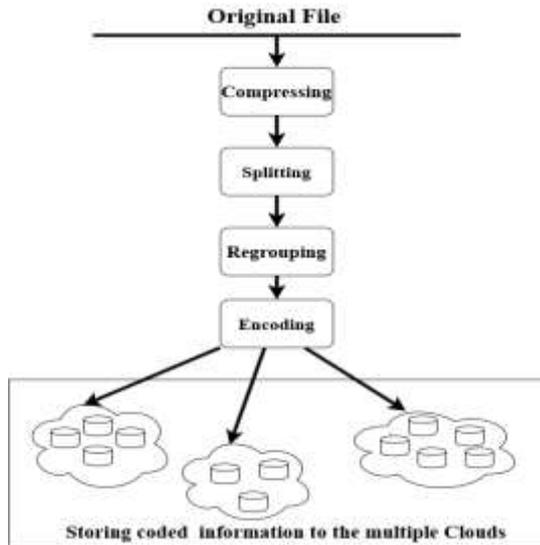


Fig.1: Flow of the Proposed Design

III. Flood Issue

The flood issue right now that the length of the put away coded information in the cloud repository can be greater than the length of the original information. It tends to be characterized as

- Closely Non-Flood
- α -bounded Non-Flood

Closely Non-Flood

The size of the original information is equivalent to the size of the system coded information. This is a process of closely non-flood encoding.

α -bounded Non-Flood

In [9], a bit of encoded information is α -bounded Non-Overflow if and onlyif

$$\sum_{j=1}^{|\tilde{c}_i|} l_d(c_j) \leq |\tilde{c}_i| \alpha l_d(b_i)$$

for $1 \leq i \leq p$.

where,

\tilde{c}_i =Encoded information vector that stockpiled in the i -th cloud repository

$|\tilde{c}_i|$ =Number of components in \tilde{c}_i

$l_d(a)$ =Number of a digit that speaks to a in base d

b =Original information cluster

c =Encoded information vector

α =Number of encoding activities

IV. Compression

For compressing the original file Lempel–Ziv–Welch (LZW) is used. Lempel–Ziv–Welch (LZW) is a compression algorithm introduced by Abraham Lempel, Jacob Ziv, and Terry Welch. In [11], Xie and Kuo proposed a protected LZ78 calculation by utilizing various lexicons and arbitrarily choosing one mystery word reference for each encoded string. In [12], Zhou et al. proposed a protected LZW calculation dependent on arbitrary lexicon inclusion and change. Contrasted and innocent encryption of packed bit stream created by a word reference encoder, the safe lexicon encoders proposed in [11, 12] are relied upon to be less computationally substantial (or at least with a similar computational unpredictability) in light of the fact that the encryption is consistently implanted into the lexicon encoding process. The two plans are likewise planned not to bargain the coding proficiency while as yet keeping up an elevated level of protection from cutting edge assaults particularly picked plain text and picked cipher text assaults. In [10], LZW calculation, the index of another section added to the dictionary is consistently $d + 1$, where d is the index of the last included section. This permits coding the dictionary indices with a steady number of bits as indicated by the present number of legitimate dictionary sections. Since the LZW decoder consistently attempts to develop the same dictionary as the encoder, the two sides are synchronized in the manner how they interpret each coded dictionary index. By using this algorithm, the original file is compressed and it helps to reduce the storage space in the distributed cloud stockpiling framework. Fig.2 shows the storage size of compressed file and the original file in the cloud. Clearly, this idea would not take the large storage space in the cloud stockpiling framework.

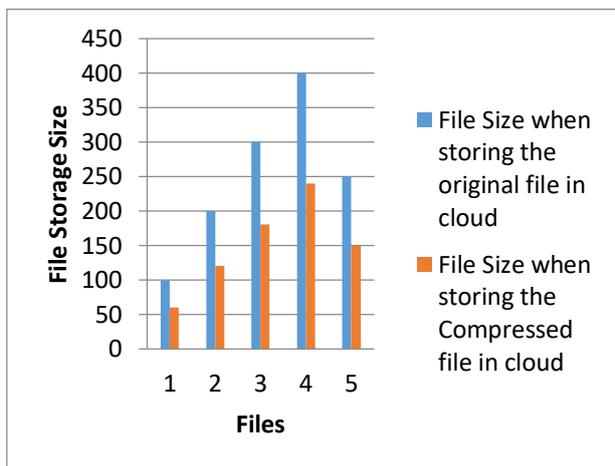


Fig.2: Example diagram for storing Original file versus Compressed file in cloud

3. A dynamic-length letters in order portrayal of network coded information

In [9], Yu-Jia Chen and Li-Chun Wang proposed two theorems for taking the dynamic length alphabet representation of network-coded data. Theorem 1 for clearly non-flood and Theorem 2 for the α -bounded Non-flood. Based on these two theorems, the length of the compressed information is taken. After taking the dynamic length from theorems, the compressed information is splitted and regrouped.

V. Encoding and Storing

Right now, the coding activities are implemented in Galois Field $GF(2^3)$, with the basic polynomial $P(x) = x^3+x+1$. After implementing the encoding, the coded information are stockpiled to the multiple clouds.

An approach for united coding and arrangement issue can be spotted in [5]–[8]. The creators of [5] proposed an encoding-mindful information arrangement plan to accomplish throughput increases of encoding tasks by considering the relation among various clouds during the encoding procedures. A versatile network coding stockpiling plan was suggested in [6]. Encoding technique is balanced by the transportation conditions. In any case, stockpiling cost of coded information isn't designed. In [7], the creators suggested to encode pieces utilizing paired expansion so as to reduce encoding multifaceted nature. It is appeared that the ideal tradeoff between capacity limit and fix data transfer capacity are accomplished. The most significant to our effort is [8]. It researched how to stockpile information dependably to the different clouds and gave the ideal measure of information to be put away

in the clouds. The stockpile costs are demonstrated exceptionally influenced by capability number of distributed cloud repositories. In [9], the encoding procedure and the information situation are together structured for network coding system with the stockpiling cost just as surveillance necessities.

Different feature of our work by comparing with these previous works,

- i) Researched that the stockpiling cost reduction of cloud stockpiling framework. In the investigation, the splitting of original file is a big reason for stockpiling cost.
- ii) Unique in relation to the past work [9] granting the reduction of storage costs, we extended that the cost of storage is reduced by compressing the original file.

VI. STORAGE ANALYSIS

We are investigated that, measure of stockpiled information with surveillance necessity regarding the likelihood that a spy could acquire first information. That is on the grounds that solitary a specific measure of coded information parts are stockpiled in neighbourhood machines that could improve the surveillance level. As the necessary surveillance level builds, the proportion of encoded data set aside at the neighbourhood site increases. Notwithstanding the coded information, the customer needs to keep the encoding lattice for deciphering so it dwells in the neighbourhood machine. The measure of the stockpiled encoded information is the important design parameter for the proposed Network Coding Cloud Capacity Framework. In practice, the system with the large encoded information requires a huge memory to store all the coding coefficients. In the proposed design, the storage of the large encoded information is reduced by 40% because of applying the compression in original file.

VII. RESULTS AND DISCUSSION

To execute the client application and distributed storage, we build up the coding layer and capacity layer of Network coding capacity Framework. Every unique record is related with the information which incorporates the coding data. The objective of our investigation is to investigate the coding performance of our created Network coding stockpiling framework in terms of the stockpiling cost. Fig.3 shows that the storage cost is reduced by 40% of the Original file storage.

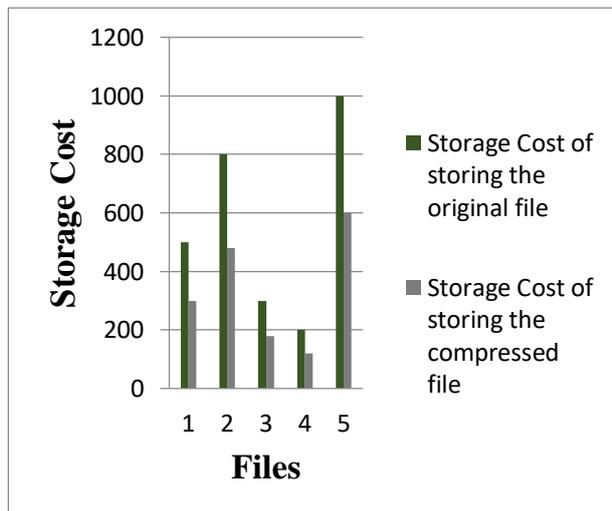


Fig.3: Storage cost of original file and compressed file

VIII. CONCLUSION

Right now investigated that the flood issue in a network coding distributed stockpiling framework. Flood issue generates new extra space and builds coding pace. We built up the flood evasion network coding based protected stockpile design. Precise methodology for the ideal encoding and stockpile specifications were given to solve the flood issue and reduces the stockpile cost.

We exhibited that encoding proficiency as far as preparing time can be improved by mutually structuring the encoding and the capacity framework parameters. All the more critically, we proposed the design for Network Coding Stockpile Framework for streamline the performance tradeoff among surveillance prerequisite, stockpile cost, and coding preparing pace. Our continuous work focuses on the augmentation of recovering the nodes and files, which is a fascinating topic to examine further in the future.

IX. REFERENCES

- [1] P. F. Oliveira, L. Lima, T. T. V. Vinhoza, J. Barros, and M. Medard, "Trusted storage over untrusted networks," IEEE Global Communication Conference, 2010.
- [2] H. C. H. Chen, Y. Hu, P. P. C. Lee, and Y. Tang, "NCCloud: A networkcoding-based storage system in a cloud-of-clouds," IEEE Transactions on Computers, vol. 63, no. 1, pp. 31–44, 2014.
- [3] F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow, "Secure cloud storage meets with secure network coding," IEEE Transactions on Computers, vol. 65, no. 6, pp. 1936–1948, 2016.
- [4] P. F. Oliveira, L. Lima, T. T. Vinhoza, J. Barros, and M. Medard, "Coding for trusted storage in untrusted networks," IEEE Transactions on Information Forensics and Security, vol. 7, no. 6, pp. 1890–1899, 2012.
- [5] R. Li, Y. Hu, and P. P. Lee, "Enabling efficient and reliable transition from replication to erasure coding for clustered file systems," IEEE Transactions on Parallel and Distributed Systems, vol. PP, no. 99, pp.1–1, 2017.
- [6] J. Li, Y. Liu, Z. Zhang, J. Ren, and N. Zhao, "Towards green IoT networking: Performance optimization of network coding based communication and reliable storage," IEEE Access, vol. 5, pp. 8780–8791, 2017.
- [7] H. Hou, K. W. Shum, M. Chen, and H. Li, "BASIC codes: Low complexity regenerating codes for distributed storage systems," IEEE Transactions on Information Theory, vol. 62, no. 6, pp. 3053–3069, 2016.
- [8] P. Hu, C. W. Sung, S.-W. Ho, and T. H. Chan, "Optimal coding and allocation for perfect secrecy in multiple clouds," IEEE Transactions on Information Forensics and Security, vol. 11, no. 2, pp. 388–399, 2016.
- [9] Yu-Jia Chen and Li-Chun Wang, "An Overflow Problem in Network Coding for Secure Cloud Storage", IEEE Transactions on Parallel and Distributed Systems, vol.30, no. 4, pp. 789-799, 2019.
- [10] [Shujun Li](#), [Chengqing Li](#) and [Jay C.-C. Kuo](#), "On the security of a secure Lempel-Ziv-Welch (LZW) algorithm", [IEEE International Conference on Multimedia and Expo](#), 2011.
- [11] DahuaXie and C.-C.Jay Kuo, "Secure Lempel-Ziv compression with embedded encryption", Security, Steganography, and Watermarking of Multimedia Contents, vol. 5681 of Proceedings of the SPIE, pp. 318–327, 2005.
- [12] Jiantao Zhou, Oscar C. Au, Xiaopeng Fan, and Peter Hon-Wah Wong, "Secure Lempel-Ziv-Welch (LZW) algorithm with random dictionary insertion and permutation", IEEE International Conference on Multimedia and Expo, pp.245–248, 2008.