North American Journal of Engineering and Research

Est. 2020

Cost-Effective Log Management Through Data Compression and Filtering

Aakash Aluwala
Email:akashaluwala@gmail.com

Abstract -

This paper aims at identifying a cost efficient approach to log management though data compression and filtration method. In doing so, the approach optimizes the system's storage effectiveness and the timely access to data, thereby increasing the system's efficiency as well as its security. The paper proves that storage cost is reduced and time for incident response is improved as the result of using effective log management in the current IT infrastructure.

Keywords– Log Management, Data Compression, Cost-Effective, Filtering, Log Management System, Log Analysis.

1. Introduction

It is important for organizations to effectively practice log management in order to ensure that security records of computers are properly documented enough to detail and retained for a required length of time. Also, the security assessment of simple log analysis can be helpful in detecting security breaches, compliance violations, scams and system issues. One of the major issues that many organizations face when implementing log management is to meet the demand of an unending stream of logs with a prescribed amount of resource for log management [1].

Several challenges may be associated with log generation and storage. for instance, an organization may have numerous log sources and logs which may have different content, formats, or timestamps across various sources and the size of log data increases gradually. The other aspect of log management is the security measures for the confidentiality, integrity and accessibility of the logs. The other issue with log management is to perform the security administrators, system and network administrators analyze logs adequately and proper consistently [2].

Some of the log management platforms have sophisticated features like data compression that extends the storage capacity and enables organizations to store much more data than would be conceivable in standard log management systems. In addition, the disk space required to store log files may be very large depending on the number of logs generated per instances and the retention period. However, if the same data are stored in an organisation's database. For example, compressing them instead of storing them in their raw form can save a great deal of storage space and it can presumed as cost-effecive.

Additionally, excluding unnecessary events and integrating noise reduction to essential events can assist in concentration [3]. Therefore, the aim of this paper is to analyze cost effective log management techniques through data compression and filtering by adopting secondary qualitative method. In addition, by reducing storage costs log data can concomitantly reduce the overall cost of storing the log management. Also, the objective is to evaluate compression algorithm and classify existing methods of data compression in order to to decrease the size of log management without large information loss.

2. Literature Review

The global community is advancing into the twenty-first century and with it, the use of electronic media is growing progressively each year. Every minute in 2015, there is upload of 300 hours of videos on YouTube, over 250000 pictures on Snapchat, 110000 call made on Skype, yet people see it to be a fraction of the total data generated and transmitted [4]. Since log management is a powerful method and is an amalgamation of Log analysis with event correlation which enlighten about the main cause of any attack and network can be safeguarded from security breaches.

Thus, there are two approaches in the Log Management – log analysis and event correlation. The log analysis and event correlation are feasible in the process of accumulating information in uncovering of insider hazard. For event correlation it is mandatory to profile a log analysis so that

it drops out all undesired information and performs one or many actions. For this, some of the used methods include data compression techniques and filtering out, as these methods help in the cost-effective log management to enable storage and transfer through restricted channels [5]. The common structural parts of a data compression system is depicted in Fig.1. The input data is handled by a Statistics Compressor where the latter typically contains two loops passing through the data. The first step contains an attempt at acquiring knowledge about the data for which compression is to be achieved in the subsequent step. The compressed data as well as the other data used in the effectual compression is then deposited or communicated to the receiver over a network [6].

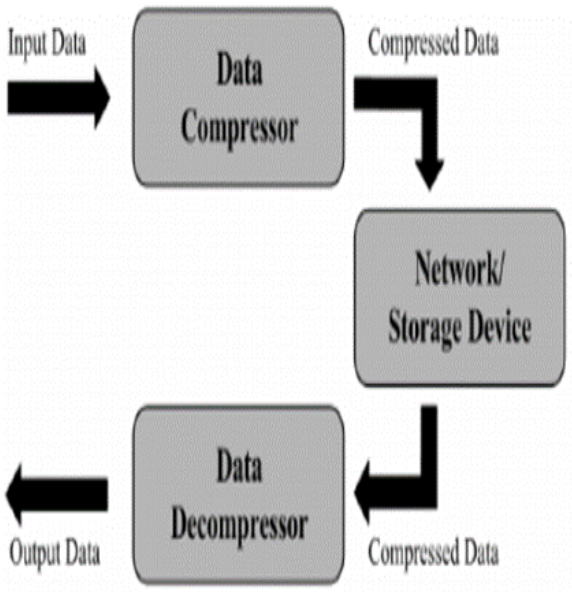

Figure 1:Constituents of Data Compression System (Source:Gupta et al., 2017) [6]

The receiver on the other hand uses a decompression algorithm to decompress the log management data. While data compression has been found to lower the consumption of a network's band width and the storage space, the method incurs more computational resources in the formation of the compressed and decompressed data through computation and this may be detrimental to application such as the embedded devices with extremely restricted computation ability [7].

Thus, there is always a trade between the compression ratio and the time taken for compression/decompression of the data using a compression algorithm. The techniques of compressor can be categorized into two main groups, (a) Lossless, (b) Lossy. Essentially, fixing the lossless compression means that no data or information is lost in the compression and the filtering processes. This is done by encoding the file in fewer bit where it is not possible to lose information during compression [6].

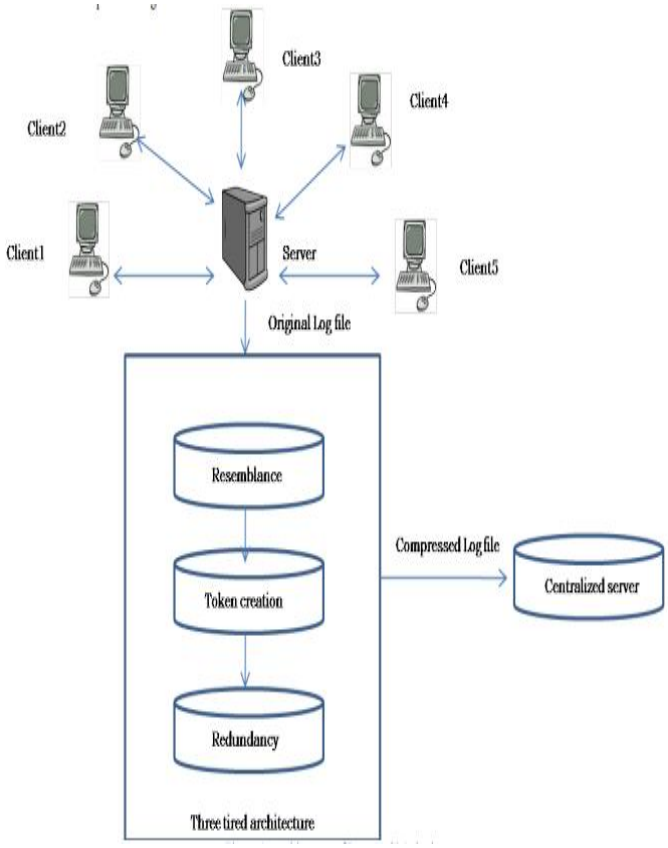

Figure 2: Architecture of Proposed Method (Source, Gupta et al., 2012) [8]

On the other hand, J-bit encoding (JBE) functions changes bits of data in order to minimize the extent and enhance the effort fed to additional algorithm. The general concept of this algorithm is that they divide the input data into two data and where the first data have holded the new nonzero byte while the second data contain the bit value referring the position of nonzero and zero bytes. Both of the data can then be compress separately with other data compress procedure to get the highest ratio of data firmness [9].

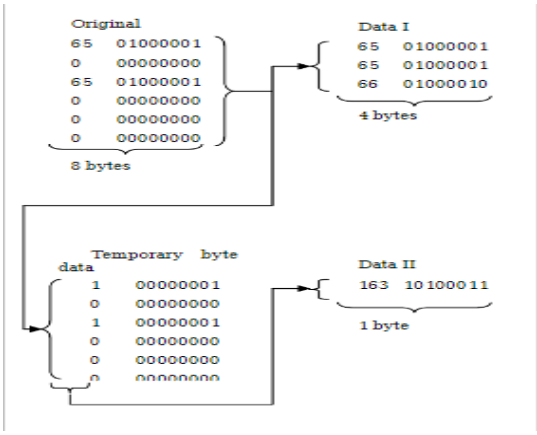

Figure 3: J-bit Encoding process
(Source: Suarjaya, 2012)[9]

As it is depicted in Figure, J-bit encoding following the compression method of the given encoding technique. The introduced original inputs starting index of length has been used as the data about data one and two. If the result of a recited bit operation is negative or positive determine, it can be read bit as '0' or '1'.If bit 'read' is '1' then read data 'I' and store it to output, if the 'read' bit is '0' then inscribe information to the output [9].

On the other hand, gathering information from numerous sources enables system administrators and security specialists to get an understanding of the existing condition of the system. In this projected technique it has utilized log file as the fundamental statistics feed to the occurrence correlation system. Log file is made up of discrete log entries which are records and this is made up of a solitary line of text. At server side, these log histories are composed from diverse clients. Above figure shows the real log assortment scenario.

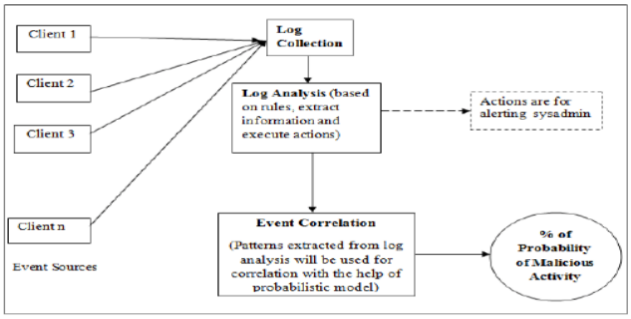

Figure 4: Proposed System
(Source: Ambre, 2015) [5]

3. Monitoring Tools Impacted

The impact of data compression and filtering out tool can be measure when organizations provides a real time view of protecting networks, system activities and invaluable in maintaining security, availability, and stability of information technology assets. Though, the monitoring tools with the log management system ensures that the process of the collection, analysis and the storage process of the data is conducted effectively and efficiently.

The benefit of monitoring tools is that there is time for responding to threats being able to take the necessary measures. Sophisticated technologies are employed to analyze log information in present in order to define the presence of possible threats, or violations of existing regulations. This capability enables the Information and Communication Technology (ICT) infrastructure and security personnel to be in a position to manage risks appropriately hence reducing on incidents such as data leakage or unlawful access [10].

Furthermore, they can effectively filter out many unnecessary logs and thus, the volume of occurrences to be stored and analyzed is decreased and in this way, cost-effectice technique adopted. As there is success in improving the measurability of the performance of the system, the monitoring tools are constantly observing CPU, memory, network and other metrics. By incorporating them with logs allows the administrators to study performance trends, analyze issues, and manage resource distribution. This way an organization can avoid breakdowns of its systems which is very vital for the business undertakings in an organization.

Furthermore, the monitoring tools assist in meeting regulatory requirement by ensuring that log information is well captured, archived and easily retrievable incase of audit. Many industries have laid down certain standard procedure that has to be followed in handling the data, its retention and security and one of the ways the monitoring tools help in following these standards is by providing the features such as logging and reporting. Apart from avoiding legal repercussions and financial losses, it also enhances the image of an organization especially when dealing with its consumers and other administrative units.

4. Tasks

As organizations continue to produce ever-increasing amounts of machine log data, it becomes increasingly important to find ways to reduce log management costs, such as through compression and filtering. This will involve an evaluation process and planning to ensure that these strategies are implemented appropriately. The first activity involves a review of the current state of the organization with regards to log management. This assessment should also consider all systems and applications that produce logs, the type and format of logs; and the amount of logs over time. Understanding how logs are currently being collected, transported, and stored is valuable knowledge that should be obtained [11].

The assessment should also evaluate stakeholders who frequently review logs for activities such as monitoring, auditing, and troubleshooting. Once the as is landscape has been documented, the log data volumes and growth projections have to be estimated. This will give insight of future infrastructure scaling requirements and its costs in case efficient optimization techniques are not incorporated. The comparisons with the industry averages and standards can identify areas of inefficiency to strive for [12]. The assessment results should be organized in a comprehensive report format and disseminated to the stakeholders as a way of raising awareness on the need to undertake an optimization project.

Once the logistic needs are understood, the next step is to choose the relevant data compression algorithms. Various types of algorithms can provide different degrees of compression ratio and the utilization of CPU. Algorithms such as ZIP, Snappy, and LZ4 are suitable for simple machine-generated log data with recurrent patterns and are widely applied in log management tools [13]. It should be

useful to perform a proof-of-concept to evaluate and compare these algorithms on a sample of production logs. Parameters like compression speed, compression ratio, and decompression speed should be considered to determine the most suitable approach [14].

At the same time, the process of defining what filtering rules are necessary, should begin as well. This is done by defining the core and shared objects which are evaluated by the stakeholders under regular workshops and surveys [15]. An example of fields that are not often used can become candidates for exclusion. Filtering rules will also have to use log segmentation patterns where only the type or source of logs needs more depth of retention and analysis. It will be necessary to update filtering rules on a regular basis as analytic demands and legislation evolve. To systematically manage this, a centralized rules configuration interface should be built. This enables the stakeholders to easily change the filter criteria after the review and approval process has been completed. Automated testing makes it possible to check that rules work correctly before being released.

The next task is integration of the optimized compression and filtering chain into the log management framework [16]. This involves integrating the selected compression algorithms into Log collection tools, transportation agents, and the backend storage systems. Filtering rules must be coded and implemented in log parsing as well as in the storage tier of the system. Rollout is the process of pushing configuration updates to development, staging and production environments [17]. It also involves equipping the appropriate stakeholders with knowledge on how to sustain and update filters in the future. Additional analytics and visualization glue logic must be addressed to retain the usefulness of the filtered data. Pre-production testing ensures that strategic and technical specifications have been met before full production is embarked on.

5. Solution and Implementation

Many companies have implemented the Log Management System (LMS) technique for collecting, categorizing, and archiving log data and event logs originating from multiple sources into a single center. Reserachers by implementing log management software systems enabled the IT staff, DevOps as well as SecOps personnel to define a starting point through which all network and application data is retrievable. Usually this logbook file stands searchable to such an extent that the IT group can get results they require in order to see the health check of the Network, prioritizing of resources or security measures [18].

In addition, sophisticated logs are critical in the detection of security threats and preventing of risks. Since such tools analyze log data in real-time, they are also capable of recognizing systemic threats, unauthorized attempts, or potential weaknesses, and immediately mitigate risks to the Company's information. However, many security tools have found problem by limiting the log sources that emerge with many non-security tools. Wrong logs create noise in the security instruments, which leads to alert fatigue. So as it accumulates, data storage costs that can rise sharply when working with gargantuan figures of log information.

With data compression and filtering technique cost-effective log management, organizations can lessen the amount of data they need to store and still maintain the essential information. Therefore, the log management tool does the work of Cribl. Thus, by implementing a data filter, data compression, and data enhancement before storing data at the beginning of a business process organizations could save large sums of exertion while still being capable of accessing important data in the future for analysis or compliance. Each case will be different, but the early findings show 20%-45% log ingest savings.

6. Results

The data compression and filtering of log management enabled storage and performance. Through the help of some modern techniques in data compression, it was found that the size of the log data could be compressed and reduced by about 60 percent in average, which in-effect affected the storage cost. These means of filtration were found to be effective in removal of other information that is included in the logs which was not required hence simplifying the amount of information that needs to be processed. Comparisons of the performance indicated that the new system of log management to give data three- to fourfold faster. This strengthening response to incidents and diagnostics of systems. Secondly, study have also included the aspect of security which entail aspect of data's integrity, availability and confidentiality to meet regulatory standards and policies regarding the data.

7. Conclusion

In conclusion, the study reported through secomdary qualitative method have reported the approaches of data compression and filtering in log management demonstrate a valuable solution for the organizations that struggle with the problem of the increasing amounts of log data and the shortage of storage capacity. Besides, this approach minimizes the storage costs in addition to orienting the data access and analysis to the speed that is useful forsecurity incident identification and system performance monitoring. Thus, the implementation of these techniques show that data integrity and data availability objectives can be achieved along with significant cost reduction. Further, by accomplishing the filtering of data, the number of data to be analyzed and processed is substantially lowered, which benefits the overall activity of security systems and prevents excessive alerting, leaving them more effective. More research can be carried out in the future to determine improved ways of increasing the efficiency of the application of compression algorithms and the selection of the filter criteria for log management systems.

References

[1] K. A. Kent and M. Souppaya, "Guide to Computer Security Log Management," 2006.

[2] A. Oliner, A. Ganapathi, and W. Xu, "Advances and challenges in log analysis," *Communications of the ACM*, vol. 55, no. 2, pp. 55-61, 2012.

[3] O. Söderström and E. Moradian, "Secure audit log management," *Procedia Computer Science*, vol. 22, pp. 1249-1258, 2013.

[4] J. Josh, "Data never sleeps 3.0," *Domosphere blog*, 2015.

[5] A. Ambre and N. Shekokar, "Insider threat detection using log analysis and event correlation," *Procedia Computer Science*, vol. 45, pp. 436-445, 2015.

[6] A. Gupta, A. Bansal, and V. Khanduja, "Modern lossless compression techniques: Review, comparison and analysis," in *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 2017: IEEE, pp. 1-8.

[7] K. Sayood, *Introduction to data compression*. Morgan Kaufmann, 2017.

[8] R. Gupta and R. K. Gupta, "A modified efficient log file compression mechanism for digital forensic in web environment," *International Journal of Computer Science and Information Technologies*, vol. 3, no. 4, pp. 4878-4882, 2012.

[9] I. Suarjaya, "A new algorithm for data compression optimization," *arXiv preprint arXiv:1209.1045*, 2012.

[10] M. Cheminod, L. Durante, and A. Valenzano, "Review of security issues in industrial networks," *IEEE transactions on industrial informatics*, vol. 9, no. 1, pp. 277-293, 2012.

[11] R. Casado and M. Younas, "Emerging trends and technologies in big data processing," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 8, pp. 2078-2091, 2015.

[12] L. Wang *et al.*, "Bigdatabench: A big data benchmark suite from internet services," in *2014 IEEE 20th international symposium on high performance computer architecture (HPCA)*, 2014: IEEE, pp. 488-499.

[13] H. Annangi, "Security Log Analysis Using Hadoop," 2017.

[14] J. Kunkel, A. Novikova, E. Betke, and A. Schaare, "Toward decoupling the selection of compression algorithms from quality constraints," in *High Performance Computing: ISC High Performance 2017 International Workshops, DRBSD, ExaComm, HCPM, HPC-IODC, IWOPH, IXPUG, P^ 3MA, VHPC, Visualization at Scale, WOPSSS, Frankfurt, Germany, June 18-22, 2017, Revised Selected Papers 32*, 2017: Springer, pp. 3-14.

[15] D. E. White, N. D. Oelke, and S. Friesen, "Management of a large qualitative data set: Establishing trustworthiness of the data," *International journal of qualitative methods*, vol. 11, no. 3, pp. 244-258, 2012.

[16] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," *IEEE access*, vol. 2, pp. 652-687, 2014.

[17] C. Tang *et al.*, "Holistic configuration management at facebook," in *Proceedings of the 25th symposium on operating systems principles*, 2015, pp. 328-343.

[18] Y. Tian, X. Li, and Z. Yang, "The Research and Design of Log Management System Based on Struts Frame," in *2008 International Symposium on Computer Science and Computational Technology*, 2008, vol. 2: IEEE, pp. 694-697.