Distributed Digital Forensics File Carving using Hadoop HDFS and MapReduce

Darren C. Sherratt
School of Design and Informatics
Abertay University
Dundee, Scotland
2018/2019

Abstract

Context

As files grow larger, and the cost per GB of storage shrinks, the amount of storage capacity possessed by the average person has continued to grow. In a digital forensics setting, this presents a problem; the time required to perform a forensic analysis is increasing. This paper serves to develop and investigate a digital forensic file carving system using methods designed for "Big Data".

Aim

To develop a Hadoop cluster capable of performing digital forensic file carving for the purposes of assessing viability for local deployment and compare file carving times against single machine methods such as Autopsy.

Method

The cluster will be created using individual computers in an isolated intranet in a lab environment; failing this, virtual machines will be used to simulate a Hadoop cluster. A program will be created to take a dd disk image and, using MapReduce and HDFS, efficiently perform file carving.

Results

Results will be measured as time to completion for varying sizes of disk image. While the finished system should be capable of handling any size of dataset, it may be necessary to use smaller sets due to a lack of cumulative hard-drive capacity. The same dataset will be supplied to both the cluster system and a traditional single computer system for metrics.

Additionally, any discrepancies between the baseline files found by traditional methods and the cluster setup will be scrutinised as this would risk invalidating the cluster method as a viable digital forensic tool. As with a digital forensics investigation, maintaining data integrity is crucial.

Conclusion

Consumer datasets are bigger than they've ever been and digital forensic investigators are in need of a system which can alleviate their already over-encumbered digital forensic laboratories. A system capable of scalable processing power would go a long way to reduce the time it takes to process a disk image and complete a digital forensic investigation.

Keywords

Big Data, Digital Forensics, Hadoop, HDFS, MapReduce

1 Introduction

A multitude of factors come into play in when evaluating the increase in average volume of data. Smartphone cameras take increasingly high-resolution pictures with increasing data size, movies are now available in 4K resolution with individual file sizes above 10GB and with more content available to download - both legal and illegal - it has never been easier to gradually accumulate terabytes of data for personal use. To ascertain if a suspect is in possession of illicit materials, a complete evaluation of all data storage devices must be carried out and because it is possible to hide data outside of partitioned space, the full capacity of the device must be scanned.

In addition, the price per gigabyte of hard disk space has continued to fall. As shown below, the HDD price per GB has slowed in recent years but continues to fall.

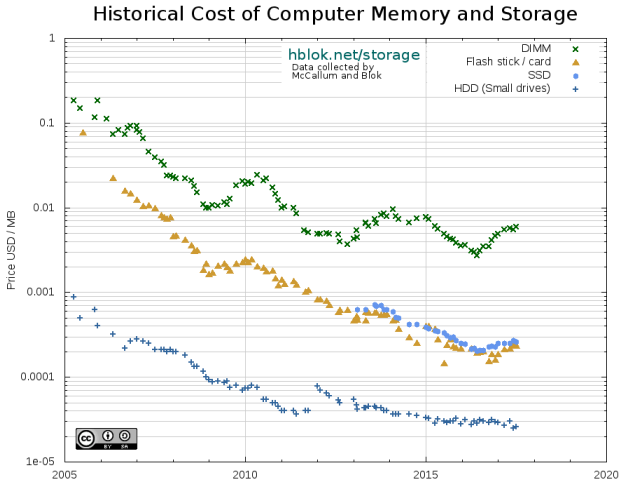

Figure 1: HDD Price per MB (*hblok.net - Freedom, Electronics and Tech*, 2017)

The average person in 2016 owned more than three connected devices (Buckle, 2016), and that number is set to increase. Most personal computers have at least 1TB of hard drive storage and the storage capacity of mobile devices is also increasing, with up to 1TB of storage available in a mobile phone.

Digital forensics investigators have been facing increasingly tight deadlines for years. In 2016, the U.S. Department of Justice

[1] A service request is a request for forensic examination, device imaging, or search warrant assistance. This request will be submitted to the laboratory from a law enforcement agency

completed an audit of a New Jersey Regional Computer Forensic Laboratory. This facility was found to have 194 service requests[1] that were not closed within 60 days, including 39 that had remained open for more than a year (U.S. Department of Justice, 2016*a*). An earlier 2015 audit found that across all 16 regional forensic laboratories operating under the FBI, 893 of 1,566 open service requests had been open for more than 90 days (U.S. Department of Justice, 2015).

When a digital forensics investigation is required, there is the potential for the suspect to be involved in or conducting illicit activities potentially involving the harm or exploitation of other people or children. Therefore, it is imperative to conduct and report on an investigation in a short enough time frame so as to reduce the risk of the suspect re-offending. A suspect can only be held without charge for a maximum of 24 hours. Therefore, while an investigation is underway or a HDD is waiting to be processed, a suspect could commit further crimes, attempt to evade the course of justice, or destroy other evidence. Therefore, it is imperative to complete digital forensic investigations in as little time as possible. However, with the rapidly increasing quantity of data available to consumers, investigations are taking too long, and some digital forensic laboratories are facing backlogs of six months to a year (Casey et al., 2009), and a year is more than enough time for a suspect to create the same amount of data again, if not more. Regardless of the size of the data or the number of suspect devices, completed analysis of the evidence is required as soon as possible and while available tools have improved, the average size of the data sets has increased by a significant margin in recent years. Between 2008 and 2016, the Regional Computer Forensics Laboratory's processed data total increased from 1,756 terabytes of data to 5,667 terabytes. (U.S. Department of Justice, 2008) (U.S. Department of Justice, 2016*b*).

2 Background

The increasing size problem in digital forensics has been known for several years and as such, solutions have been developed and promising results shown. However, as mentioned, digital forensic laboratories are still severely backlogged. The assumption that has to be made is that either none of these solutions has been suitable, or none have lived up to the promised performance. This project intends to adapt established frameworks and concepts to provide a solution.

As a solution, file carving using local Hadoop clusters is not an exhausted area of study. Research for this paper found more papers specialising in performing digital forensics on the cluster itself, rather that using the cluster for compute power.

Recently, companies are being founded on the principle of digital forensics as a service, the purpose of which is to outsource digital forensics to more powerful computers in datacenters. While a potential solution, this adds another point of failure and requires the storage and transport of potentially illegal and dangerous images to a company who may be providing services to a number of clients. Additionally, digital forensics as a service requires a forensic laboratory to pay for services which could be brought in-house.

2.1 Literature Review

The World Academy of Science paper "Digital Forensics Compute Cluster: A High Speed Distributed Computing Capability for Digital Forensics" (Gonzales et al., 2017) details a proprietary compute cluster software suite which can run on either Amazon Web Services (AWS) or a local computer cluster. This project takes the open-source digital forensics package Autopsy and gives it the capability of distributed computing with a focus on cloud-based computation. The finished result achieved an eight-times speed increase over a single computer report on a 232GB disk image. While this is reminiscent of the work proposed in this document, this project will be open source and focus on using Hadoop to handle the cluster compute specifics. Additionally, the finished product will use local clusters and all data will remain local, or on another owned cluster. This brings all variables and potential risk in-house.

In the Vassil Roussev journal entry "A Cloud Computing Platform for Large-Scale Forensic Computing" (Roussev et al., 2009), Roussev et al. discusses the alternative MapReduce algorithm MPI MapReduce (MMR) and the viability and benefits of performing digital forensic using a data-centre centric distributed compute cluster. In the paper, it is discussed that before digital forensic laboratories can evaluate the use of cloud-based methods, a purpose-built software suite first needs to be developed; this paper then describes a proof-of-concept software infrastructure which could be developed and implemented for use by digital forensic laboratories. Roussev et al. found that the traditional Hadoop MapReduce did not lend itself particularly well to digital forensic applications, this prompted their research and development of MMR.

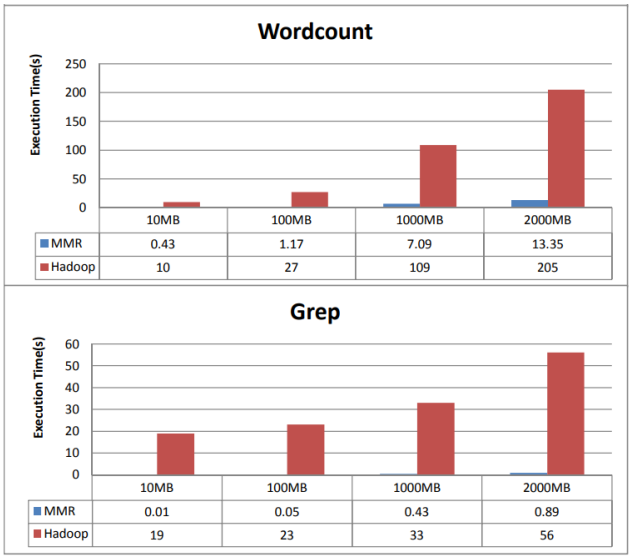

Figure 2: Roussev et al. MPI MapReduce vs MapReduce (Roussev et al., 2009)

MPI MapReduce will be looked into for this project and evaluated. As shown in Figure 2, the results displayed in the paper show a potential speed increase of 20x over traditional MapReduce but without repeatable findings, these figures are meaningless. MMR is open-source which allows for ease of access but is potentially more complicated to implement as it is a newer implementation with less documentation. In the paper, Roussev et al. discuss the benefits of MMR when using CPU computation. This would prove beneficial as it is unlikely that a rudimentary forensic laboratory cluster would have access to computers equipped with GPUs. The use of MMR will be evaluated in the development stage of the project to determine viability for this project.

A container-based method for scalable digital forensics is examined in the 2017 paper, "SCARF: A container-based approach to cloud-scale digital forensic processing" (Stelly and Roussev, 2017). In this article, Stelly and Roussev investigate the potential for coping with the growth of volume of data by using a container based software framework capable of large-scale deployment. The approach developed in this paper focuses on processing speed and pays little heed to functionality, preferring instead to develop a system designed for performance from the ground up while enabling third-party compatibility to integrate and develop the SCAlable Realtime Forensics solution (SCARF) for their own purposes. Additionally, the container-based approach developed is language agnostic, allowing for high, or low level languages to be used for third-party development. For the purposes of this project, a container-based approach would not be fitting; if an approach was to be developed from the ground up, containers would be considered but, based on the timescale available developing a complete solution is infeasible.

3 Method

3.1 Research

Shown in Figure 3, after a suspects devices have been catalogued and logically imaged, the disk images are processed and analysed. File carving is part of the processing stage, raw byte-by-byte copies of the suspects drives are read in, and files are fed into the analysis stage which, in the case of Autopsy, structures a timeline of events and provides a GUI for browsing and investigating the retrieved files. During these processes, it is vital to maintain integrity of the data. If the data is changed then it is inadmissible as evidence. Traditionally, the files are hashed at regular intervals and the hashes compared.

The solution created in this project will fit within the processing stage. The scalable, distributed computing approach used in this project will provide a solution which will be capable of integration into any digital forensic laboratory at little cost thanks to the open-source nature of Hadoop, and the compatibility and ease of integration with off-the-shelf hardware.

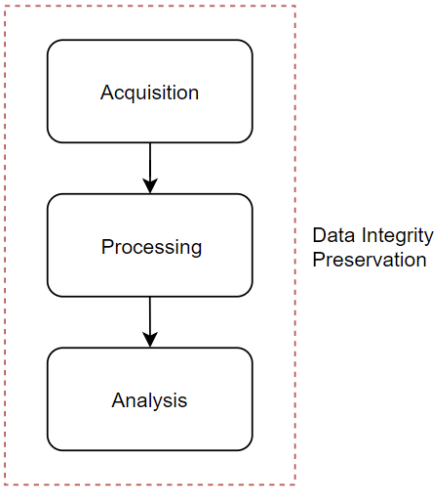

Figure 3: Digital Forensics Data Stages

Before a Hadoop cluster is implemented it will be required to understand how a cluster operates.

Additionally, the finished files will be required to be in a format which can be used as an input to the analysis stage. This will necessitate research into the structure that existing methods use. Due to the focused functionality of the cluster, for an analysis of the results to be possible, the files must be compatible with an existing method or in a state that a method could be purpose-built for the analysis stage.

3.2 Development

Programs indented to run with MapReduce can be written in a multitude of languages including Python, Java, and Scala. Of these languages, this project will be written using Scala for the MapReduce functionality. This language is not as efficient as Java in its execution but is far easier to program in and this will aid in programming complex functions and allow for far easier troubleshooting and modular programming. Code written in Scala is also easier to document and read by other parties. During the development stage, MapReduce and MPI MapReduce will be tested to determine if the speed increase granted by MPI MapReduce is significant. If so, it may be required to change to another language such as Java or C++ to be able to implement MPI MapReduce; this will be established early in the development stage.

The DataNodes will operate using the Ubuntu Linux distribution. This operating system will be used because Hadoop can be installed easily using pre-installed packages and Additionally, Ubuntu is a very commonly used Linux distribution and has an extensive online community; this provides a wide support network if issues should arrive with the operating system or the integration of Hadoop.

MapReduce is fault-tolerant with each node periodically reporting back to the DataNode with a status check. If no status is received, or the status is unexpected, MapReduce will reassign the job in progress to another node. This way, even on inexpensive - and potentially prone to failure - hardware, MapReduce can maintain up-time without user interaction.

For the purposes of this project, Hadoop was chosen over distributed computing due to the requirement for the capacity of dealing with large datasets. In distributed computing, resources are dealt out to the nodes as required. This creates a bottleneck at the host as each node attempts to read from a local source. Hadoop, on the other hand, uses HDFS which distributes the files in chunks across the cluster. When a task is assigned, the NameNode attempts to pair compute functions with nodes which already possess the required data; this way, the data access is local and fast.

3.3 Evaluation

The intent of this project is to provide a fast, efficient method of preparing data for analysis in a digital forensic investigation. As such, the difference in processing speed between the developed solution and currently used methods is of the utmost importance. If, after investigation, the developed solution is faster only by a small margin, the project will not be deemed to be viable. The set-up and potential cost of a Hadoop cluster solution would outweigh the benefit of a small efficiency increase.

The finished algorithm implemented using MapReduce may not have the same functionality as a finished tool such as Autopsy. E.g. the ability to recover files required to display emails sent from and received by the suspect may not be implemented in the final product; this would not necessarily be seen as a point of failure because this is functionality which could be implemented at a later date. At the end of the project, the functionality of the final artefact will be evaluated and discussed to ascertain if the required functionality is present.

The integrity of the files from beginning to end must remain fully intact. Hashes of the files will be compared at the end of the process to determine the preservation of integrity as if these hashes are found to be different then the data has been changed and cannot be used for further analysis or a court report. This should be tested and addressed throughout project development.

The disk image that will be used for testing will be either acquired through available resources such as the university or online repositories or will be purpose-built for this project. Depending on available disk size, available datasets, and available time, the size of the test dataset is subject to change. In order to extrapolate the time required to file carve very large datasets, multiple datasets of varying sizes will be tested and graphed.

4 Summary

Digital forensics backlogs are ever-present in digital forensics laboratories and despite some action being taken, cases are still being kept open for over 90 days. By increasing the speed of digital forensics file carving, cases can be completed faster. The finished piece of work will be capable of fast and efficient file carving on run-of-the-mill computers, providing a solution that can be implemented in a digital forensic laboratory. By increasing the available computing power for file carving, files can be analysed faster and any backlog can be alleviated. While work of this nature has been completed to a professional degree previously, there does not exist a Hadoop HDFS and MapReduce solution designed for file carving in a digital forensics environment. By making the project open-source from day one, modifications to increase efficiency or add capabilities can be undertaken by anyone. By not restricting access, the code can be further refined by those who want to use such a solution. With increasing dataset sizes, a big data solution to a previously small data problem could provide a useful addition to digital forensics.

References

Buckle, C. (2016), 'Digital consumers own 3.64 connected devices'.
URL: *https://blog.globalwebindex.com/chart-of-the-day/digital-consumers-own-3-64-connected-devices/*
(Accessed: 27 September, 2018)

Casey, E., Ferraro, M. and Nguyen, L. (2009), 'Investigation delayed is justice denied: Proposals for expediting forensic examinations of digital evidence*', *Journal of Forensic Sciences* **54**(6), 1353–1364.

Gonzales, D., Winkelman, Z., Tran, T., Sanchez, R., Woods, D. and Hollywood, J. (2017), 'Digital forensics compute cluster: A high speed distributed computing capability for digital forensics', *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering* **11**(8), 944–951.

hblok.net - Freedom, Electronics and Tech (2017).
URL: *https://hblok.net/blog/storage/*
(Accessed: 09 October, 2018)

Roussev, V., Wang, L., Richard, G. and Marziale, L. (2009), *A Cloud Computing Platform for Large-Scale Forensic Computing*, Vol. 306 of *Advances in Digital Forensics V*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 201–214.
URL: *https://link.springer.com/chapter/10.1007/978-3-642-04155-6_15*
(Accessed: 03 October, 2018)

Stelly, C. and Roussev, V. (2017), 'Scarf: A container-based approach to cloud-scale digital forensic processing', *Digital Investigation* **22**(S), S39–S47.

U.S. Department of Justice (2008), 'Regional computer forensics laboratory annual report for fiscal year 2016', p. 6.
URL: *https://www.rcfl.gov/downloads/documents/2008-rcfl-national-report*
(Accessed: 30 September, 2018)

U.S. Department of Justice (2015), 'Audit of the federal bureau of investigations philadelphia regional computer forensic laboratory radnor, pennsylvania'.
URL: *https://oig.justice.gov/reports/2015/a1514.pdf*
(Accessed: 28 September, 2018)

U.S. Department of Justice (2016*a*), 'Audit of the federal bureau of investigations new jersey regional computer forensic laboratory hamilton, new jersey'.
URL: *https://oig.justice.gov/reports/2016/a1611.pdf*
(Accessed: 28 September, 2018)

U.S. Department of Justice (2016*b*), 'Regional computer forensics laboratory annual report for fiscal year 2016', p. 12.
URL: *https://www.rcfl.gov/downloads/documents/Fiscal%20Year%202016*
(Accessed: 30 September, 2018)