Security Threats of Large Data (December 2017)

|  |  |  |
| --- | --- | --- |
| Gerardo Garza  [Mail.garza@gmila.com](mailto:Mail.garza@gmila.com) | Ekaterina Pirogova  [epirogova@smu.edu](mailto:epirogova@smu.edu) | Scott Gozdzialski  [Sgozdzialski@smu.edu](mailto:Sgozdzialski@smu.edu) |

***Abstract*—This paper presents the computer and network security threats to individuals inherent to large data, or big data, and some theoretical fixes.  The world today is looking for faster and cheaper ways to handle big data, but they have not focused on the risks involved. A simple breach can expose confidential business information or even millions upon millions of individuals to risk, as was the case in the Equifax break exposing 140 million individuals' information.  For all of this, it is becoming more important that as we grow, we develop security procedures for big data and we build security into future tools and systems**.

.

**I. Introduction**

Reading the news today it is not hard to see violations of computer and network security everywhere. Whether it is an article about Kapersky Antivirus and Security Suite having a relationship with the Russian intelligence agency, the FSB, or the thousands of cell phones infected by fake apps on the Android play store. Perhaps most worrying, the security breach at Equifax which exposed 140 million individuals’ social security numbers, addresses, and other personally identifiable information. These breaches show the dangers to individuals caused by untrusted relationships with seemingly trusted agents, nefarious individuals finding loop holes in the system, and lax security practices at large corporations. What if there were some systems that had inherent security issues? What if they were prominent technologies in their infancy but with such explosive growth that newer security could not be implemented? One of these technologies is big data, whose growth has been so pervasive that most of the technology has focused on how to keep up and not how to secure.

The exponential growth of big data into and beyond exabytes [1,2] is reminiscent of the growth of computers in the late 1970s and early 1980s [3], as well as the growth of the internet in the late 1990s and early 2000s [4]. There is a global explosion of technology and need, with everyone from individuals, small companies, and large corporations, to ever larger governments using big data. As such, there is a need for many different technologies to handle these security problems.

The different technologies all have different foundations in the control and manipulation of the data. There is no underlying standard to secure each instance of different security risks and fixes. The nature of the differences in tech build in securities risks. This will be one of the focuses of this paper: what security has been baked into the technology and where can we build from there.

We are in a time where everyone is watching everyone else. We have television shows where we watch other peoples’ lives, called reality television, and we live stream everything. As creatures, we are both voyeur and exhibitionist, but we like our privacy all the same. When we hear things like our phone company has super cookies that track everything we do [5], we protest in the street because this violation of privacy is not granted but taken. With big data, there is also concern of anonymity being taken. If I have enough data, we can in theory parse that data down to a single person and take the privacy away from the individual.

As big data becomes more and more used in individuals’ lives by the companies we trust, companies we don’t trust, and even governments, the risk to individuals is being removed from their control to the control of other entities. With this hand-over of control of individuals’ information, the risks are not always controlled in a proper fashion. Minor infractions in security policies can cause large incidents, as is seen in the US government Office of Personnel Management losing millions of members’ information with security classification and in what organizations they work [6]. The Equifax incident was caused by delaying the implementation of a security patch for a known issue [7].

We will be focusing on the inherent risks of big data. One of the major points will be a look at the different systems out there to store and manipulate big data, as well as some of the security concerns with these systems. Additionally, we will focus on some historic examples of security breaches and what went wrong, and lessons learned.

Finally, we will be laying on an innovative approach to security in Big Data. Big Data has some inherent challenges associated with it [8]. When data is so big it bogs down a business’s IT resources it is hard to imagine a way to properly secure it from intruders while maintaining confidentiality. We will do this with a four-prong approach, the Gerardo, Pirogova, Gozdzialski (GPG) method, with a focus on storage, transmission, and finally processing of the data.

**II. Big Data systems and Security**

**III. Historic Examples /Lessons Learned**

**IV.Gerado Pirogova Gozdzialski (GPG) method**

a. The focus of the GPG method for protect big data, of petabytes or greater size, will focus upon data at rest or storage, data transmission, and finally data processing. The key to the entire method is simple as data gets bigger it bogs down resources [8], whether this is storage, processing, or even transmission capabilities. Thus, the need for new approaches or systems to protect our information. With data growing at such a fast rate we need to be able to protect it. To do this, we need to change our thoughts of security, security is currently thought of as a locked box where a key opens the entire system to compromise. We need to think of it more like a house. In your house you have many rooms, some are more secure, like an office or bedroom, while others are open, lie a living room or kitchen, finally, some areas are open and not as protect, like the patio and porch. But, all has some expected over-arching security.

The key to this is the data, called set A has some subset B, where if B is secure A is expected to be reasonably secure. Without B the content and context of A cannot reasonably be ascertained through analysis or brute force. Set B can also be split up into subset C, D, E, F and so on, so that with only part of each subsets of B (called set CDEFtoN from now on) you cannot determine the nature of set B. Finally, without complete knowledge of any one subset CDEFtoN you cannot determine the content or context of the subset. As we continue the foundation upon which this is built will become clearer.

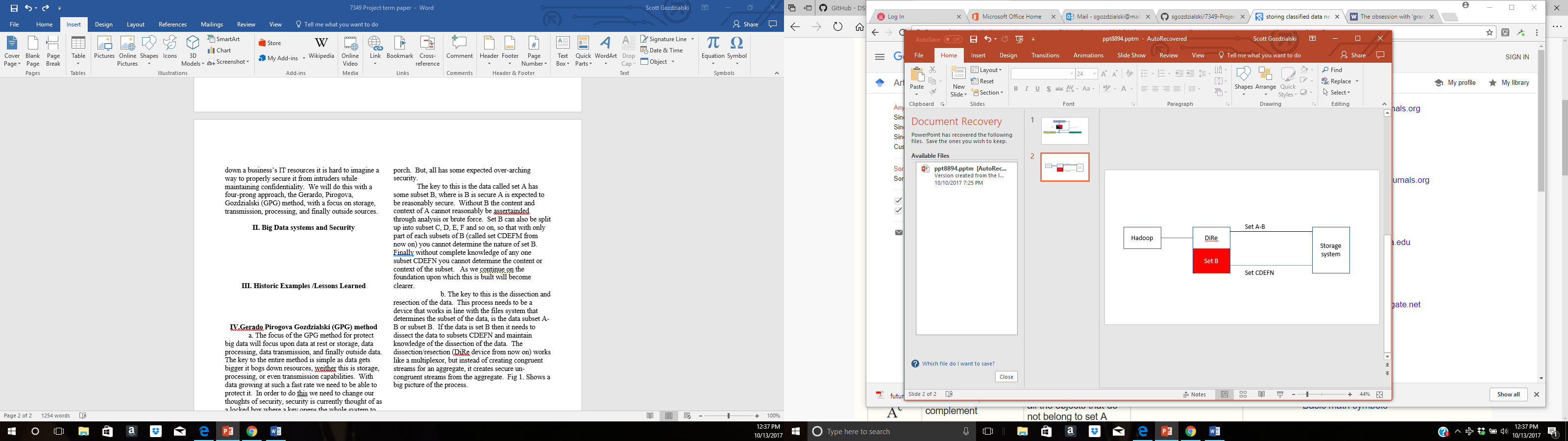
b. The key to this is the dissection and resection of the data. This process needs to be a device that works in line with the files system that determines the subset of the data, is the data subset A-B or subset B. If the data is set B then it needs to dissect the data to subsets CDEFtoN and maintain knowledge of the dissection of the data. The dissection/resection (DiRe device from now on) works like a multiplexor, but instead of creating congruent streams for an aggregate, it creates secure un-congruent streams from the aggregate. Fig 1. Shows a big picture of the process. 

Fig 1

c. The storage of the data in the file system is a big security concern as the data sits in a static position there is the potential for outsiders to capture and analyze the data. As your data product increases in size you have a greater risk to security by the sear volume. To mitigate the risks the GPG method will store the multiple subsets of data (A-B, C, D, E, F,..N) next to each other as one data file with different encryption processes run on each subset in a random order. Without knowledge of the encryption for each piece or the order stored it will be infeasible to gather any actionable information on the static data. The best way to think of this is like encrypted DNA. We can read the proteins of DNA, but without the order the proteins it is nearly impossible to determine the purpose of that section.

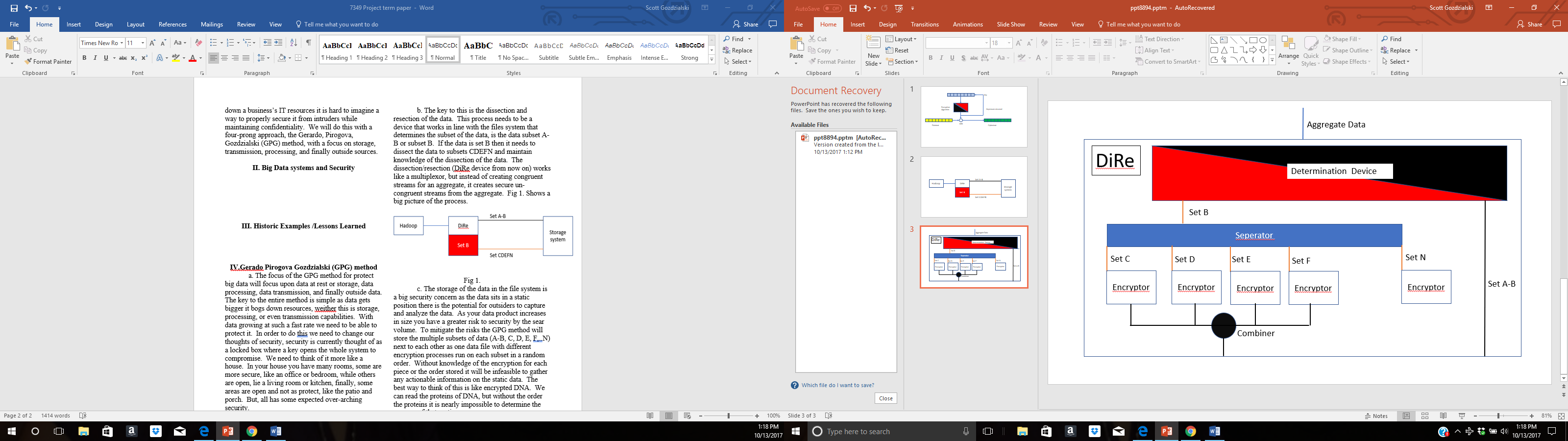


Fig 2

As is displayed in fig 2. The first step of the DiRe is to determine if the data is part of set A-B or set B. this will be done based upon a set of rules set up before. If the data is of set A-B then it is sent straight to the file storage system. If the data is of set B, it is then sent to the next step which is a separate where set B will be separated out to different subsets of B. The set CDEFtoN is then encrypted. Each set of data is encrypted by a separate unique encryption algorithm. The key and algorithm for each set is unique during each session so each set of say C is not encrypted with the same key or even the same algorithm adding another layer to the process. The different sets CDEFtoN data are combined in a random fashion into an aggregate to be set to storage with each aggregate file uniquely and randomly tagged for tracking purposes. The tag data and encryption process are stored on onboard storage on a separated stored/bus system used only for Command and control of the device, like an isolated Master Boot Record. When the data comes in from the file storage it comes through the process again. The runs through the determination device. Set A is passed through, Set CDEFtoN is separated based upon the stored tag file, decrypted, and finally combined back to set B. With this process even if a whole file output by the DiRe is completely captured, different pieces are encrypted randomly by different algorithms and different keys creating a multidimensional problem for anyone trying to crack the file, making it nearly impossible to determine the content or context of the file.

d. The next step is the transmission of the data from the processing area, which contains the DiRe to the file storage area. If the data is stored in the cloud then we lose control of the process and must trust the process up to this point. If the storage area is controlled then we want to fragment the overall data set A, so each part is stored in multiple locations, whether it is in completely different SANs or different racks within the same SAN.

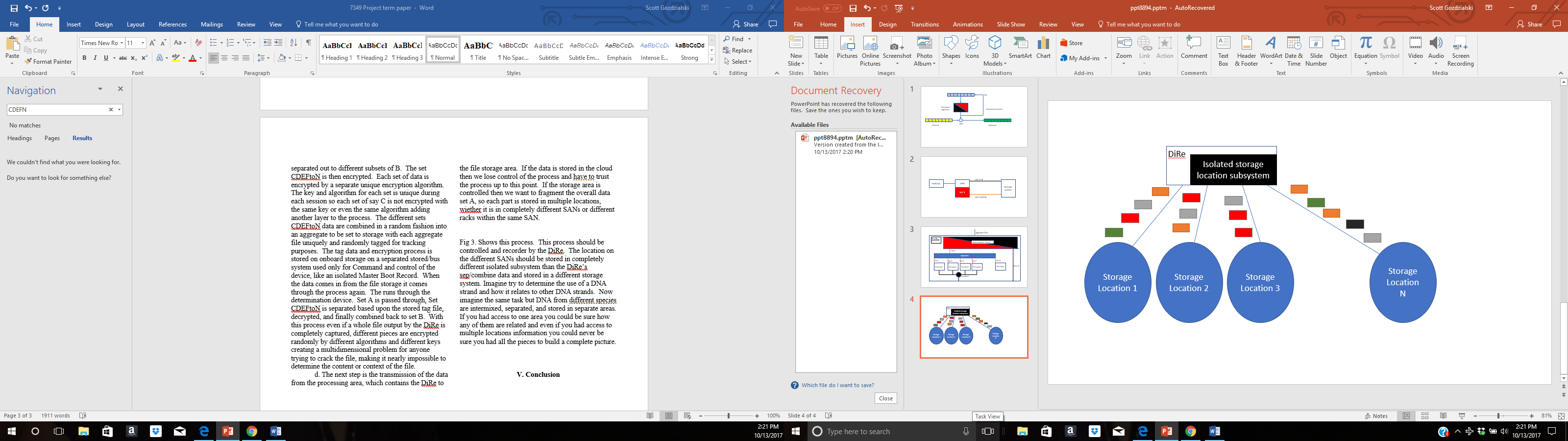


Fig 3

Fig 3. Shows this process. This process should be controlled and recorder by the DiRe. The location on the different SANs should be stored in completely different isolated subsystem than the DiRe’s separated combined data and stored in a different storage system. Imagine try to determine the use of a DNA strand and how it relates to other DNA strands. Now imagine the same task but DNA from distinct species are intermixed, separated, and stored in separate areas. If you had access to one area you could be sure how any of them are related and even if you had access to multiple locations information you could never be sure you had all the pieces to build a complete picture.

e. Finally, the GPG method will control the processes run during data processing. To protect our data from erroneous and malicious applications we would incorporate an application whitelist (AWL) in the data processing step. The focus on this prong of the GPG method for securing big data is to limit the ability to deduce the content of the data through a file hidden within the data. As the data becomes bigger and bigger it will be harder to sift thru all of it to determine if it can be trusted. As we process the data it will be easier in our data processing step to run only dedicated processes. When we limit the processes running on the CPU we drastically reduce the vulnerability to our processing system. We also, reduce the likelihood of an adversary deducing the content or the context of our data. As multiple parts of the GPG method need to have different control of the different processes. For the Actual data processing we will need to control the distinct functions very precisely because this is the one area where the data is raw. Malware or control of this process will open your entire system to corruption. The whitelist at this section will be the most locked down to limit the applications run. If the processing of the data is done on a cloud system, it is recommended to use the next severely controlled while list will be inside the DiRe control system. Even though it is strongly recommended to run the DiRe control system information on an isolated network and subsystem, it is an added step to only allow only trusted processes from and to trusted locations. The information inside the DiRe is the keys to the entire process.

**V. Conclusion**

**VI. Related Work**

**VII. References**

[1] Chaowei Yang, Qunying Huang, Zhenlong Li, Kai Liu & Fei Hu (2017). Big Data and cloud computing: innovation opportunities and challenges. *International Journal of Digital Earth, 10*, 13-53.

[2] R.L .Villars, C.W. Olofson, M. Eastwood, *Big data: what it is and why you should care*, White Paper, IDC, 2011, MA, USA

[3] Emma Kavanagh. 2017. *The Evolution of Computers.*

<http://www.nortonsecurityonline.com/security-center/evolution-of-computers.html>

[4] 2017 *Internet Growth Statistics: Today’s road to e-Commerce and Global Trade Internet Technology Reports.* <http://www.internetworldstats.com/emarketing.htm>

[5] Definition Super cookie. <http://searchsecurity.techtarget.com/definition/supercookie>

[6] Cybersecurity Resource Center.

<https://www.opm.gov/cybersecurity/cybersecurity-incidents/>

[7] Cybersecurity Incident & Important Consumer Inforamtion.

<https://www.equifaxsecurity2017.com/consumer-notice/>

[8] Avita Katal, Mohammad Wazid, RH Gourder, (2013) *Big data: Issues, Challenges, Tools and Good Practices.*

<https://pdfs.semanticscholar.org/34ad/09cda075101dc4ce3c04006ff804aca3ebf8.pdf>