International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 3, Issue 2, Mar 2014, 47-66

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

ENTERPRISE IT LOGGING IN THE CLOUD: ANALYSIS AND APPROACHES

ADNAN SHAOUT & RYAN BANKSTO

Department of Electrical and Computer Engineering, University of Michigan, Dearborn, Michigan, United States

ABSTRACT

As the number of servers and the amount of data stored continues to rise for enterprise information technology (IT) organizations, the amount of data being logged by most organizations far out scales the resources available to store, analyze, and monitor those logs. With the rise of cloud computing as a cost effective infrastructure as a service provider for both storage and compute power, the creation of a widely used cloud based enterprise IT logging solution seems inevitable. In this paper, we identify the challenges of logging data in an enterprise IT environment, perform a literature survey of existing research in the field, identify key components of a successful service, review several existing commercial cloud based logging services and finally propose an architectural framework for our own cloud based enterprise IT logging service.

KEYWORDS: Cloud Computing, Logging, Hadoop, Cloud Architecture, Big Data

1. INTRODUCTION

In many information technology (IT) departments, logging events occurring within systems and network is a common practice. Logs are generated by workstations, servers, switches, routers, firewalls, web sites, databases, and nearly every hardware or software system in an IT environment. Logs may contain information relating to proper or improper execution, data access, or any other record of event occurring within a system or service. Depending on the organization, logging may be done for the benefit of the IT administrators or there may be legal requirements such as the Health Insurance Portability Accountability Act (HIPAA) [1], the Payment Card Industry Data Security Standard (PCI DSS) [2], the Sarbanes-Oxley Act (SOX) [3], the Federal Information Security Management Act (FISMA) [4], Gramm-Leach-Bliley [5] and others. In present day enterprise IT, there exist several mature logging services and protocols. Unfortunately, many of these logs are stored discretely and the existing framework for log collection and storage provides no analysis of the data. Many times this data is collected by a traditional log collecting server known as a sys log and is never looked at again. Gigabytes to terabytes of data are devoted to the storage and backup of these logs despite little or no use in many IT organizations. Even in organizations where logs are used for purposes such as troubleshooting or performance monitoring, the analysis is often done in an inefficient manner with a different tool for different logs. Separate and discrete systems are used to log and analyze different systems such as Microsoft Windows system events, Apache web server access logs, and firewall logs. As the number of servers has increased from 5 million in 1996 to over 30 million in 2008 [5] and more storage being consumed than ever, with a projected increase of data stored globally from 329 exa bytes in 2011 to 4.1 zetta bytes in 2016 [7] [8], the approach to logging in enterprise IT organizations has not caught up with the increases in storage and server technologies. In fact, the technologies for collecting logs have not progressed much since the 1980's when Eric Allmen first developed Sys log as a way to manage logs for his send mail program [9]. With the increase in the amount of computer users, servers and storage, a need for more efficient logging technologies is apparent.

Along with the rise of the number of servers per organization, cloud computing has also recently become an increasingly useful computing paradigm for affordable storage and compute power. In this paper we will present required and useful components of a cloud based logging and analysis server. In this approach, we hope to combine the storage efficiencies of the cloud to perform the basic sys log functions, while leveraging the compute utilities of the cloud to provide deeper levels of log analysis.

The paper is organized as follows: Section II present Background and Details of Problems in Existing Logging Environments, Section III presents Background to Cloud Computing and Applicability to Enterprise Logging, Section IV presents Literature Survey – Existing Research and Related Work on Cloud Based Logging. Section V presents Components of a Successful Cloud Based Logging Service, Section VI presents Review of Existing Commercial Solutions, Section VII presents Framework and Design Considerations for a Next Generation Cloud Logging Service, Section VIII presents Further Research and Evolution of Cloud Based Logging, and Section IX presents conclusion.

II. BACKGROUND AND DETAILS OF PROBLEMS IN EXISTING LOGGING ENVIRONMENTS

Logging is an integral part of an IT organization. Logs can be used to troubleshoot hardware failures, software problems, data access, and authorization. Challenges faced by IT organizations with log management include the need for long term storage, storage growth, analysis of the data, and a lack of consistent logging tools between IT systems.

A. Long Term Storage

Companies that process credit card information on their computer systems are required to meet a stringent set of rules related to their logs called PCI DSS. This standard states that companies who process credit card information must, “Retain audit trail history for at least one year, with a minimum of three months immediately available for analysis” [2]. Another industry regulation known as the Sarbanes Oxley Act (SOX) applies strict accountability and auditing requirements for all public companies in the U.S. as well as any international company that has registered with the U.S. Securities and Exchange Commission. Additionally, all accounting firms that provide auditing and accounting services to these companies must also be SOX compliant. In the SOX compliance guidelines, companies have to keep seven years with of relevant data related to the auditing of their accounting [3]. Because most, if not all, accounting is now done with computers, computer logs pertaining to both the accounting system as well as the auditing software need to be retained for seven years.

B. Storage Growth

Market research continues to show that the number of physical servers continues to increase, despite the increasing use of virtualization [6]. With the rise of Internet usage among developing countries, in addition to increases in developed countries, the total number of internet users has risen from just over 1 billion in 2005 to over 2.1 billion in 2010, a 50% increase in just 5 years [10]. This increase in the number of servers and users goes parallel to an increase in the size and amount of data being logged. Consider an average apache log entry:

127.0.0.1 - frank [10/Oct/2000:13:55:36 -0700] "GET /apache_pb.gif HTTP/1.0" 200 2326

This example is given in the official Apache documentation [11]. This file is 85 bytes, but typically many web hosts prefer to record additional data such as page being displayed and browser agent such as the following:

192.168.1.100 - - [03/Apr/2013:17:21:48 -0500] "GET /contact.html HTTP/1.1" 200 4268

"http://www.example.com/services.html" "Mozilla/5.0 (Linux; U; Android 4.2.2; en-us; Galaxy Nexus Build/JDQ39) Apple WebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari/534.30"

This information lets them collect data to decide which parts of their websites to tailor to mobile browsers, different operating systems, and also gives the content owners useful demographic data. This single log line is now 270 bytes in size. Given that each page view will generate several of these types of log lines, a large website such as Face book or Twitter with millions of page views per day will have very large log files. In June of 2011 Face book had over 1 trillion page views [12], which would mean if each page view generated the example 270 byte log entry to its web logs, Face book would have over 245 terabytes of web log data from June 2011 alone. Most websites are not as large as Face book, but it is easy to see how data logs can quickly grow, even with modest logging enabled.

Typically in organizations where there are not stringent PCI or SOX requirements, data is simply compressed or deleted to save space. This can waste valuable demographic information that may lead to better or more targeted sales approaches. It also discards useful data for detailed computer incident response or basic troubleshooting.

C. Analysis of Data

One pitfall that besets many IT organizations is that they are collecting a vast amount of log data but have no time for analysis of that data. Often this is due to lack of resources, lack of knowledge, or lack of appropriate tools. One industry study [8] projects that the number of files will grow by 75 times while the number of IT staff will only grow by 1.5 times. With the increase in the amount of logs, the data can be overwhelming for many smaller organizations. Even larger organizations may struggle to extract useful information from a vast amount of log data. Using the Face book example above, many organizations do not have 245 terabytes of storage in their entire data center, nor the ability to perform analysis on that amount of data. Another problem is that in many instances data is divided between different IT units. For example, web logs may be owned and analyzed by a marketing team, database logs owned by a database administration team, while firewall and intrusion detection logs are owned by the security team. In larger IT organizations, this departmentalized approach means that several different systems, log types, and stakeholders need to be involved for a thorough analysis of any security breach. Frequently, when situations such as these occur, different teams have to get data with a myriad of tools which leads to wasted time and efficiency.

Consider the following scenario: a network administrator notices a high volume of traffic from a web server. Upon further inspection by the web team, a web server has been compromised and is sending spam. A computer incident response team would want to look at logs from the firewall and intrusion detection systems, the web server, any databases that may be connected and associated with that web server, and possibly the mail server logs to see if any administrative user had responded to a phishing attack. In this scenario, there are at least five different logs that would be useful for analysis.

D. Lack of Consisent Logging Tools

Using our intrusion scenario above, another problem in the enterprise IT logging landscape is that it is highly likely that each of these five systems have five different log formats and tools. Logs stored by databases, web servers, firewalls, and email servers are distinctly different types of logs. Even within the same manufacturer, there are multiple different logging formats. Microsoft's web server IIS and its email server Exchange have disparate types of logs and two distinct tools to analyze those log files. Although attempts have been made to standardize logging formats, there has been

no clear winner for a standard. There are presently two different standards and an additional proprietary format for logging that administrators can choose when configuring Microsoft's IIS web server. These include the World Wide Web Consortium's (W3C) extended log file format, the National Center for Supercomputing Applications (NCSA) common log format, and Microsoft's own IIS log file format. Without a clear standard chosen by the industry, there are there have not been standards for consistent analysis tools either.

The collection of logs has also not been standardized or consistent between the major operating system vendors. In UNIX and Linux operating systems, the sys log service has been the de facto standard for collecting log files across different daemons and services. Sys log was developed in the 1980's as part of the send mail system that collected and classified entries into a log file. As it was a simple and useful component for send mail, other services started to use sys log as their primary logging method. It has the ability to log events, debug information, across many types of systems into a central repository. Sys log has been retroactively described by RFC 3164 and 5424 [13] [14] and is implemented in all major versions of Linux, UNIX and Mac OS X. Microsoft Windows, the most used desktop computer operating system in the world, does not use sys log or have support for it by default. Windows operating systems use a proprietary Microsoft system and format known as the Windows Event Log. The Windows Event Log has been the primary source of application, system, and security logs on a Windows computer since 1993 and Windows NT. It was completely redesigned for Windows Vista in 2007, but is still not compatible with sys log without the help of third party applications.

E. Auditing and Security

A final problem in enterprise IT logging is that with the two most prominent solutions, sys log and the Windows Event Log, there do not exist sufficient controls to allow for a fine grained data access control. Typically, if a user has the ability to read a sys log folder or Windows Event Log, they can see all the information contained within that log. To be in compliance with laws such as the Gramm-Leach-Bliley act [5], one must not only know who has access to the data, but also when they accessed that data. Neither sys log nor Windows Event Log record when and what data has been accessed by what user. Windows does not allow for deletion of specific lines from the event viewer, but third party applications that use standard text files would allow for modification. Similarly, sys log logs changes to event files using standard Linux file modification attributes, but does not typically keep a history of file access to logs it has collected.

With storage capacity and computer power for thorough analysis two of the primary challenges in enterprise IT log management, using cloud computing to achieve better log management becomes an exceedingly attractive option.

III. BACKGROUND TO CLOUD COMPUTING AND APPLICABILITY TO ENTERPRISE LOGGING

Cloud computing has been a buzzword in the past few years. There are differing opinions on whether cloud computing is a new computing paradigm, a rehashing of existing ideals, or simply marketing spin. In their seminal paper on cloud computing Fox et al. [15] define cloud computing as both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. Cloud computing can refer to both the services that are provided via large scale data centers over the internet, the platform to build your own services on via large scale data centers, or direct usage of the hardware in a large scale datacenter. The National Institute of Standards has their own definition of Cloud Computing which sates, "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

This cloud model is composed of the following: five essential characteristics, three service models, and four deployment models" [16].

The roots of cloud computing derive from the concept of utility computing proposed in the 1960's by John McCarthy at MIT [17]. McCarthy suggested that in the future, computing would be available to the public similar to a telephone system. In essence, when a user needs more computational resources one can simply pay a fee and use more services. In the decades following McCarthy's suggestion, this has become a reality with infrastructure as a service (IaaS) being offered by several different companies including Amazon, Microsoft, Rack space, and others. Extensions to this idea include the notion that both an underlying application framework now called platform as a service (PaaS) and the services themselves (software as a service or SaaS) are now also realities. There are several different and useful use-cases for cloud based computing. Many commercial applications of cloud computing utilize the scalability feature to their advantage. Dropbox, the world's most popular file storage service, is cloud based software as a service solution that builds upon Amazon's hardware as a service EC2 for multi-terabytes of storage. Amazon is able to scale its storage and charge at a rate where Dropbox still makes money so the relationship is beneficial to both.

With scalable and long term storage readily available in the cloud, the storage aspect of enterprise IT log management are easily solved via cloud computing. Similarly, with vast compute resources readily available and scalable, the cloud also provides a solution for the analysis of these logs. The rest of this paper will focus on the remaining issues: extracting useful information across multiple log file formats and types with the cloud as our engine for analysis.

IV. LITERATURE SURVEY – EXISTING RESEARCH AND RELATED WORK ON CLOUD BASED LOGGING

A. Mass Log Data Processing and Mining Based on Hadoop and Cloud Computing

A framework for mass log data analysis based on Hadoop and cloud computing was proposed in 2012 by Yu and Wang [18]. This paper focuses on data mining logs from large scale SaaS applications. In SaaS, applications typically run on server clusters and log data is created at execution. The data is stored and processed according to the specific requirements of the SaaS application. Yu and Wang are quick to point out that as applications scale in size, the typical relational database management system (RDMS) is not an efficient method for analysis when data sizes are larger than 1TB. This is especially true when attempting to glean real-time statistics from a SaaS application. Even a modern server class computer is unable to run multiple queries against a 1TB database quickly.

Yu and Wang propose a framework for large scale log analysis based on Apache Hadoop. Hadoop is a highly scalable framework that is built for clusters of computers to distribute processing of large data using a simple programming model. Hadoop uses a programming paradigm Google designed called Map Reduce [19]. In Map Reduce, the first step is Map. In this step, input is initially divided into many smaller fragments and distributed to worker nodes. Worker nodes may also divide their input into smaller fragments and distribute the work to other workers. The answers obtained by the worker nodes are passed back to their master nodes.

In the Reduce step, the master node collects all of the worker nodes answers and combines them to form the output. Hadoop also utilizes a specialized file system called the Hadoop File System (HDFS) which is also scalable and highly distributed. In HDFS, each node containing data serves its data up using an HDFS block protocol. Files are distributed across the Hadoop cluster system, creating built-in fault tolerance in clusters of sufficient size. Because the

processing and data storage processes are very parallelizable, Hadoop scales extremely well and is well suited for Cloud computing, and specifically text analysis.

Yu and Wang's research is specific to logging data in the cloud that was also generated in the cloud. As such, the data they are looking for is related to properties such as service, tenant, and user of a cloud service. They use a transaction set, which is generated from the original records of log data as the target, and a constraint function set which is the usage records which satisfy certain conditions. In this way, only useful data is used for further evaluation of market forecasting and business intelligence. The specific architecture of the system is shown in Figure 1.

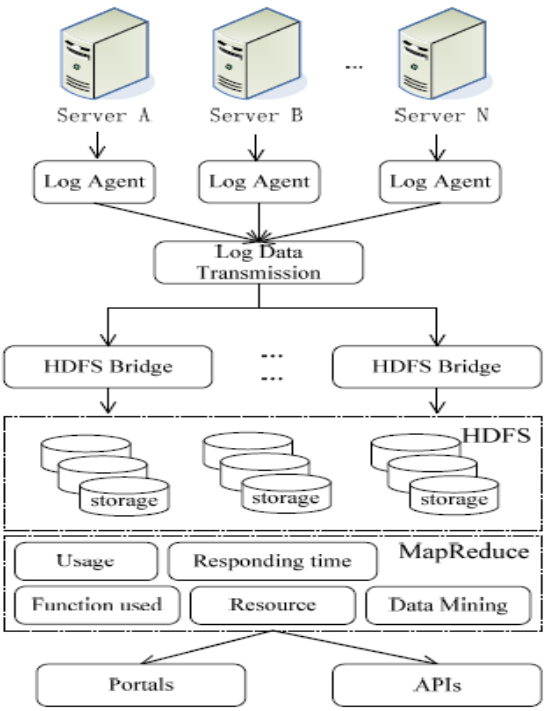

Figure 1: Yu and Wang Architectural Proposal

In this implementation, a specialized log agent must be used to generate the proper log of execution information on the Cloud platform running the SaaS applications. Using this Hadoop architecture, Yu and Wang performed an experiment with a Hadoop cluster of 10 computers. The log consisted of simulation SaaS application execution code and was stored in 10,000 files each 100M, for a total of 1TB of data. The dataset was also loaded into a traditional RDBMS system for comparison. Querying statistics such as usage, resources, response time, and others, the experiment showed that the Hadoop based system performed at least 12% to at most 41% better than a the RDBMS. They also compare their Hadoop based system against the classical Apriori algorithm often used in existing data mining techniques. The Apriori is a bottom up breadth-first search structure that can find association rules and common data groupings. Against the classical Apriori algorithm, their Hadoop implementation improves performance by an average of 45% and reduces the number of relation rules by nearly 60% making more efficient use of the data for further analysis.

B. Secure Logging as a Service – Delegating Log Management to the Cloud

A cloud based logging system focused on security, confidentiality and privacy has been proposed by Ray et al. in "Secure Logging As a Service -- Delegating Log Management to the Cloud" [20]. This approach is targeted towards organizations that fall under strict data control policies such as SOX or HIPAA. This paper assumes that data logged to a

cloud service may fall victim to a 'curious cloud provider' who may look at the log data being saved. They identify that in a service that provides secure logging, correctness, tamper resistance, verifiability, confidentiality, and privacy are the main concerns. Their framework proposes that servers send their data to an intermediary called a logging client/relay.

A logging client/relay is essentially a sys log server that performs some operations on the data before sending it to the cloud. In their example, the logging client/relay can encrypt, anonymize, and batch data together to be pushed to the cloud. The cloud in this scheme is a SaaS provider that offers storage space for the secure logging system. Essentially, the cloud can only accept data and delete data in this proposal. Ray does not utilize the cloud for data processing because the paper assumes that cloud operators cannot be trusted with private data. The final part of this system is a log monitor, a system used to monitor that logs are still available and intact on the cloud, as well as perform certain analysis on that data.

This paper is somewhat hampered by the use and focus on Dolev-Yao attacker model [21] in which they assume present in a cloud infrastructure. In this proposal, the cloud only functions as a storage location, but any analysis must be done by systems owned by the subscribers. In most cloud storage scenarios, this is a terribly inefficient method to use the cloud on many levels. The first issue is that it often costs more for storage bandwidth than it does for storage space; that is, storage providers charge for both data access and data storage. If the analysis for all of the cloud based logs must be done by in-house servers, it becomes prohibitively expensive to continuously transfer data in and out of the cloud. For example, if this infrastructure were built with Amazon S3 storage as its cloud storage provider, storing 1TB of data costs $97.28. Retrieving 1TB of data from S3 costs $122.76. A system that needs to download the data locally is more than twice as expensive as just storing the data. Adding the costs for the servers to do the local analysis and it may well be over three times more expensive than doing the analysis in the cloud as well.

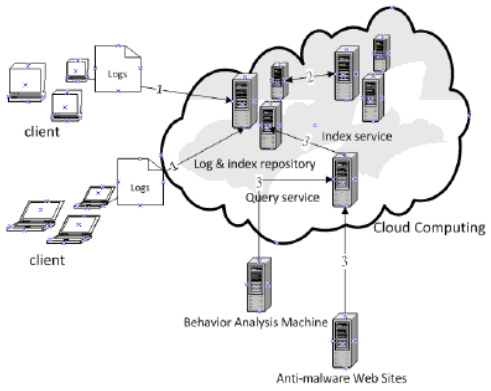

Figure 2: Liu and Chen Proposal Architecture for Detection of Malware via Cloud Computing

C. Retrospective Detection of Malware Attacks by Cloud Computing

An outline for a cloud based approach to retrospective detection of malware is proposed in [22] by Liu and Chen. Their proposal is based on a novel portable executable (PE) watchdog service that generates a log of all new PE files and relationships amongst those files to other PE files. It utilizes a lightweight log collector designed to push these logs to cloud storage at periodic intervals. The logs are processed by two separate Hadoop engines that are used to create file indexes and relationship indexes. Both behavior analysis systems and information from leading anti-malware sites are used as queries to find malware on log files in the system. A working concept of their proposal is shown in Figure 2.

The authors went on to design a prototype system and noted that although their system was somewhat more efficient than existing methods to detect malware, they experienced inefficiencies in their Hadoop Map Reduce processes

when indexing a large number of small files. In practice, building and querying on one large index file was faster than building and querying on thousands of smaller file-relation files. This insight can be important when creating a larger scale cloud based log analysis service. Despite this system being primarily focused on detection of malware, this proposal lends itself well to extensions to add other types of logs to support other retrospective queries that can be useful to IT administrators and IT security personnel.

V. COMPONENTS OF A SUCCESSFUL CLOUD BASED LOGGING SERVICE

In order to create a more successful cloud based logging service, we must identify the key components and needs of such a system. From this we can create a design framework for a successful cloud based logging service.

The challenges outlined in section II must be addressed, along with some unique issues related to the cloud.

A. Long Term Storage

A cloud based logging service must provide service level agreements that allow for long term archival of the logs in their original and unchanged format. The system needs to keep both the original copy of the logs, which can be compressed but otherwise unchanged for legal compliance for certain regulations, as well as the post processed logs used for analysis. Additionally, there needs to be a way to easily export all of the original unchanged data in the event a customer wishes to move providers or needs to reproduce the logs in original format for a court or legal proceeding.

B. Storage Growth

As scalability and growth is one of cloud based storage's strengths, this should be an easy part of the system to address. The only challenge here is that the cost of the system may be heavily dictated by the cost to store data. As such, efficient compression algorithms should be used to keep storage usage to a minimum for the original logs. Thankfully, text files lend themselves well to standard compression algorithms. We have conducted a compression experiment on 20MB and a 1.6GB apache web logs covering several months [23]. The compression was performed using a standard AMD Turion II P520 Dual-Core Processor in a system with 8GB of RAM. The freely available 7-zip file archive tool [24] was used to test different compression types, speeds, and ratios. Space savings were at best 99.45% and at worst 89.64%. However, the time and space used to compress files for the larger log file was not consistent at all. The standard deviation for file size was nearly 39430k with a standard deviation for time to compress nearing 150 seconds. The experiments results can be seen in Table 1. Although this is not a rigorous test of compression algorithm space savings, it does show that there exists a tradeoff between cost savings and processing time. A further analysis between space savings cost versus processing time and cost needs to be performed but is beyond the scope of this paper.

C. Storage Access

The service should not utilize cloud storage as proposed by Ray et al, in a manner that all data must be continuously uploaded and downloaded. The cloud storage needs to be used in conjunction with cloud compute power to make the most efficient use of the system.

D. Consistent Log Collection Tools

The service should be able to accept all text logs as input. A bridge should be written that allows for existing sys log servers to upload data en masse to this service. Similarly, a lightweight client should be installed that allows for log

shipping from Windows, Mac OS X, and popular Linux distributions. Furthermore, this agent should allow for easy shipping of user defined text logs as well. Input parsers will need to be created that are written with rules specific to different types of common log files. The service should allow for both user choice for log file type, as well as some intelligence to automatically detect a log type. Once properly classified and categorized, more specific rules can be generated that will extract the useful parts of the logs and discard what is not useful for analysis.

The input parsers on a BIND DNS server log will be substantially different from that used for a Cisco ASA Firewall. A default generic 'text log' format should be available for logs from custom or niche applications. The parser for the generic log format should still be able to identify things such as IP and MAC addresses, timestamps, URLs and filenames. Additionally, these parsers should be able to be used as templates for a user to define their own parser rules so that they can define things such as error code meanings and debug information.

E. Auditing and Security

Table 1: Analysis of Compression on Apache Log Files

Compresion method	Size after compression 21257k log	Avg. time (in sec.)	Avg. space savings
7z	116k	6.6	99.45%
zip	1487k	3.74	93.01%
bz2	10003k	8.65	95.29%
gzip	14859k	4.12	93.01%
Compresion method	**Size after compression 1589644k log**	**Avg. time (in sec.)**	**Avg. space savings**
7z	105792k	646	93.34%
zip	164749k	352	89.64%
bz2	89159k	474	94.39%
gzip	164749k	313	89.64%

The system should send logs in a secure fashion using SSL or some sort of encrypted connection. The system should allow for fine grained access control policies so that users can be given authorization to only data they are authorized to see. This will need to be set up by the users of the system, but the service should provide the ability to categorize and restrict logs on a per user basis.

Analysis within this system will abstract the previous one-to-one relationship between a user and a log. However, the system should be able to reconstruct log access back to the original log files for auditing and accountability. In a scenario where a user queries for information across logs for a web server, a database, and a file server, the system should record this as if the user had accessed each of these three logs. Additionally, access to logs and queries should be stored per user and be easy to summarize in a reporting tool.

The threat model proposed by Ray et al. where a cloud provider may be curious and surreptitiously read or modify data will not be honored or explored. Service level agreements will be used to ensure the level of trust between the users, the service providers of a cloud based logging service, and the cloud storage providers. Cloud storage providers such as Amazon and Microsoft are enormous corporations who would not risk litigation or market share loss by randomly reading a customer's data.

F. Analysis of Data

The analysis of the data should be both retrospective and in real time. The exact analysis performed should allow for querying traditional IT attributes such as IP addresses, URLs, and error messages. It should allow this analysis over all types of logs without hindrance. Non-real-time queries should involve searching for known vulnerabilities and errors with suggested fixes. Visualizations are an important part of analysis as presenting the users with a text interface does not give the much value over a traditional sys log. Both keyword search and regular expression searches should be available across multiple logs.

VI. REVIEW OF EXISTING COMMERCIAL SOLUTIONS

A. Papertrail

Paper trail [25] is a service primarily based on the standard sys log framework. Each customer is given a port which they use to send sys log data. This is not the most secure form of logging, since by default the connection is not secured with SSL/TLS and all data is sent over the internet unencrypted. An additional flaw in the security is that devices are not authorized before being able to log data.

If a typo is made in a configuration file, a Paper trail user may very well be sending their log data to another customer. Paper trail allows for logging via sys log for UNIX, Linux and Mac OS X based systems. It functions as a sys log server and can log any data that can natively be sent to sys log such as from firewalls, switches, or other network equipment. Paper trail requires the use a Windows event viewer to sys log third party application, and offer instructions on one written by Purdue University to log Windows based computer information. Additionally, for flat text file logs that are not able to use sys log natively, they provide their own remote_syslog server.

Paper trail succeeds in creating a single pane of glass to view all log files by use of sys log. However, it provides no data analysis capabilities other than search. There are no visualizations of the data, as the interface is very plain as seen in Figure 3. In fact, Paper trail does not have much beyond the ability to offer a web browser based version of the UNIX ‘tail’ command and easily search across logs.

It uses Amazon S3 as its storage backend, and does allow users to directly utilize an archived version of their logs for Hadoop analysis. Unfortunately, beyond the instructions on how to setup an Amazon instance of Hadoop, there is no automatic analysis or Hadoop integration. It is obvious that for any large amounts of data, the service is not effective as an analysis tool. Finally, there are some primitive alert capabilities that can email users or perform a custom HTTP post operation to interact with a website.

Paper trail’s cost structure is based on how long logs are actively searchable as well as the amount of data, with no per user or per system charges. Data can only be kept in the real-time search space for a maximum of 4 weeks. An example plan in their system costs $75 per month which gives a user 8GB of log storage, two weeks of search, and one year of log archive.

Figure 3: Paper Trail Interface

To store 100GB of logging data per month with Paper trail and have that data be searchable for 4 weeks costs $875 per month, or $10,500 per year. 100GB of data is not a large amount of data for medium sized business running email, web, and file services. For the functionality that Paper trail provides, $10,500 per year is not a good value for most organizations.

B. Loggly

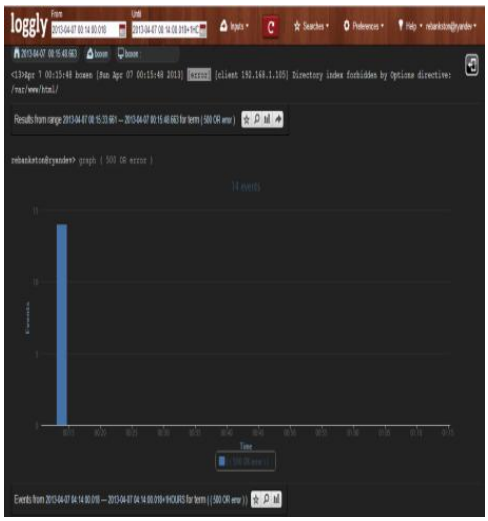

Figure 4: Loggly Interface

Loggly [26] is a service similar to Paper trail. It is based on the sys log-ng framework, but can also work with rsys log. It can use sys log-ng or rsys log to log data directly from Linux, UNIX, and OS X computers. For it to work with Windows a third party event viewer to sysl og tool needs to be installed. Loggly suggests a standard sys log server be used as a proxy to upload data that is not from a Linux computer. Similar to Paper trail, SSL/TLS can be configured but is not the default. However, for a device to log to a Paper trail account, devices must be registered via IP on the website before they are allowed to send data. Alternatively, data can be logged directly using an HTTP post method and a provided GUID for web data. Loggly offers more than just a search interface, and allows for searching, graphing, finding unique strings, and creating filters as seen in Figure 4. Unfortunately, most of these tools are done through a web based command line

interface. Alerting based on matching string criteria is available through Alert Birds and requires the use of a Google email account. Alerts can be sent to an email address, or to other monitoring systems such as Cloud kick, Nagios, or Pingdom. Loggly also allows its data to be accessed via various API for custom visualizations, but not are provided by Loggly itself.

As a comparison to Paper trail, logging costs are similar with a 100GB of data per month with 2 weeks of search plan costing $139 dollars. Loggly can keep logs searchable for up to 90 days, but that increases the price to over $500 per month for 100GB of logs per month. Loggly does not allow for archive or retention past 90 days, but offers a service to dump data to Amazon S3 for long term storage. The costs for S3 storage are not included in the Loggly pricing structure.

C. Log Entries

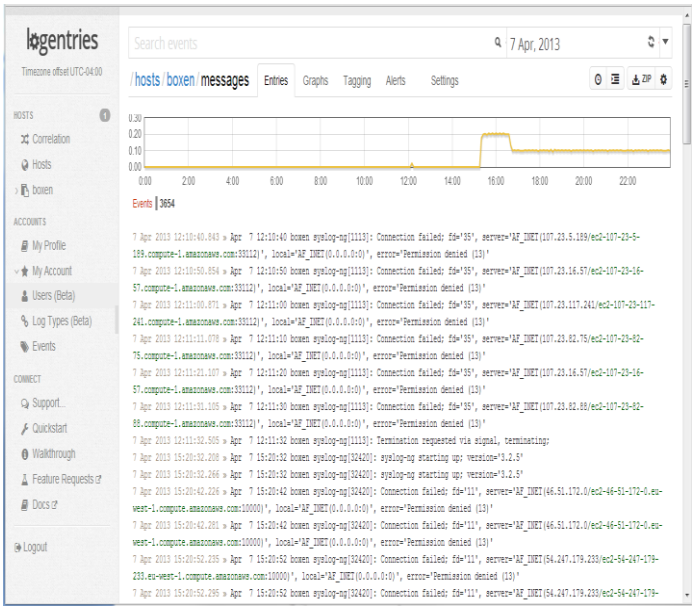

Figure 5: Log Entries Interface

Log entries [27] is yet another cloud based log service. It has custom agents for Mac OS X, Windows, and various distributions of Linux that allow for default HTTPS secure communications. Each system must be registered to an account which can be done via a specialized GUID account key or via username/password. It is also compatible with sys log and includes support for HTTP PUT logging. Log entries has a very simple command line setup, and any standard text log can be added to the system with a single command.

In addition, Log entries can monitor CPU, memory, network, and disk utilization for any hosts using the Log entries agent. Unfortunately, the search functionality does not appear to work properly across different hosts or even across different logs on the same host. This severely limits the usefulness of aggregated logging. Additionally, the visualizations via graphing were rudimentary and not entirely useful. A sample interface can be seen in Figure 5. Standard alerting is configurable via email or via I Phone application.

Figure 6: Splunk Storm Interface

Pricing for the service is in two parts. They charge $0.99/GB for indexing data, and $0.99/GB per month up to 100GB where it drops down to $0.49/GB per month. So for indexing and storing 100GB a month, it would cost $200. They do not have a limit on the amount of storage you can index or purchase, which makes it usable for large scale environments but the costs may be prohibitive.

D. Splunk Storm

Splunk Storm [28] is the cloud hosted version of the popular Splunk log searching tool. Data can be sent in a myriad of ways including Splunk API, a custom splunk log forwarder, standard sys log, sys log-ng, rsys log, snare, Heroku drain, and even manual upload of logs. Hosts being monitored must be registered with Splunk Storm via IP address, and by default using the Splunk forwarder, data is encrypted when sent to Splunk Storm.

Splunk Storm is able to automatically categorize data and isolate fields such as host or source, and suggests interesting fields which can be errors or other noteworthy events such as service starting and stopping. It can detect a log type and automatically extract relevant field data for dozens of different types of logs. It seamlessly searches across hosts and log types, and can display sections of every log for a given period of time with a few clicks of the mouse. Alternatively, the search bar is also interactive and providers type-ahead suggestions and instant matches Inside each log entry, the user can click on any word and Splunk Storm will perform a search for that word across the rest of the log. These searches can be saved for later review. Graphs are interactive and allow users to click or select a range to focus and zoom in on. Granularity of the graphs allow for each line to represent a second, or a year. An example interface can be seen in Figure 6.

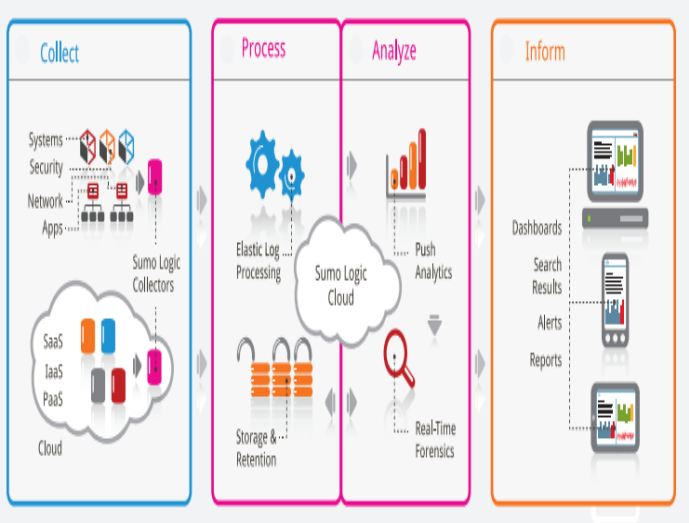

Figure 7: High Level Overview of Sumo Logic

There are presently no alerting capabilities, or anyway to look at logs in near-real time. Their plans are based on total amount stored, but still have a monthly charge. Storing 100GB of data, the costs are $400.00 per month. Users can set a deletion policy so that after a certain number of days, logs can be deleted. This service does not offer longer term storage. The largest plan offered is 1TB of data which costs $3000 per month.

Sumo Logic

Sumo Logic is a cloud based log management and analytics solution. They bill themselves as the next generation of log management. They have agents known as collectors available to all major operating systems. Each collector can act as a relay for other servers who do not have the collector installed. According to Sumo Logic's manual, each collector instance can manage 500 files combined at a rate no greater than 15,000 events/second. Collectors can manage local log files, remove files via cifs/ssh, act as a sys log, or collect logs by running an executable at a defined interval using cron. They have a patent pending process known as Log Reduce which reduces log logs into more human readable patterns. In addition, they have a feature that will use fuzzy logic and soft matching to group similar messages together to provide a quicker way for administrators to read through data.

These summarizations are rated using predictive analysis for what it believes the user will find useful based on a scale from 1-10. These summarizations can be voted on to get better results next time, and it will influence both the summarizing and prediction analysis ratings. Searches can be saved and turned into alert event emails. In addition to user search, Sumo Logic offers what they called "Push Analytics" which generate information and are proactively delivered to technical staff. These are retrospective analysis performed via data mining that can find errors, inconsistencies, or summarization of customer behavior. This is a unique feature to Sumo Logic.

Real time data and historical data are easily searchable. In addition, they are the only provider who offers role based administration to restrict users view and access to logs. Based on OS or log type, Sumo Logic provides default dashboards that show useful information specific to the view of the customer. Their interface can be seen in Figure 7.

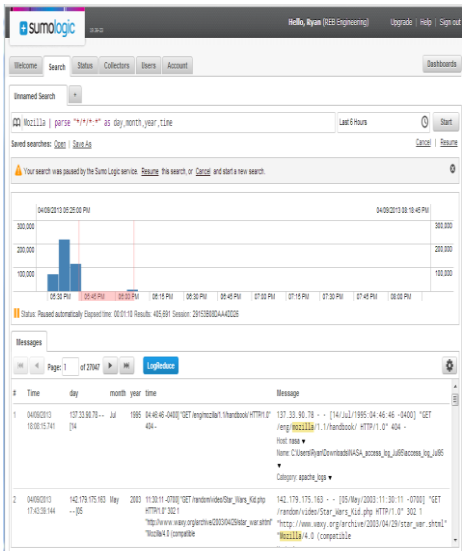

Figure 8: Sumo Logic Interface

They are a recent startup, having only been in production since 2012 but already have large customers such as Net Flix and Pager Duty, a SaaS based approach to being on-call. A high level overview of their architecture is provided in Figure 8. Pricing is based on daily log volume and retention. Their base plan allows for 5GB of logging per day (150 per month) with 30 days of retention for $400 per month. Retention of 90 days increases that cost to $600 per month, with longer retention periods available but require a sales engagement for a price quote. The largest plan they list on their website includes 150GB of data a day (4.4TB per month) with 90 days of retention for $12,999 per month.

E. Summary of Analysis and Suggestions for Improvement

A summary of the analysis for these commercial solutions can be found in Table 2. Based on this survey of existing cloud based logging and log analysis services, there are no services that meet all of the requirements for a next generation cloud based logging service. Only Sumo Logic offered any methods to authorize or audit user access to log files. There is only a meager attempt at role based accessed by the other services surveyed. Most only had granularity for read only or full access. Additionally, only Log entries and Sumo Logic are suitable centralized logging service due to the ability to scale size easily. The other services have limited storage capabilities that make them unable to be used for full replacements to a standard sys log server.

Splunk Storm's ability to analyze data across logs and systems appears to perform the best out of those surveyed. It also offers one of the easiest methods for collecting logs across multiple platforms. Unfortunately, with a 1TB for $3000 per month, it is not a sustainable service for long term storage or retrieval of the original data. It offers log type detection and automatic categorization of log files, but does not automatically detect errors or have any alerting mechanisms. Of the services surveyed, all except for Splunk Storm appear to be running on Amazon storage services. Due to each vendor being closed source and proprietary, it is hard to say for sure what techniques are being used for data analysis. Splunk Storm is most likely using a proprietary Splunk Search Language and Map Reduce implementation that they use in their standalone product [29]. Sumo Logic is a closed source solution, but admits to being deployed on Amazon AWS, using Hadoop, the programming language Scala, the Apache Cassandra no SQL solution originally designed by Face book, and an open-source graph database Neo4j. Sumo Logic also has a patent pending Log Reduce algorithm that is probably similar in some respects to Map Reduce.

VII. FRAMEWORK AND DESIGN CONSIDERATIONS FOR A NEXT GENERATION CLOUD LOGGING SERVICE

If the services outlined in section VI are the current generation of cloud based logging services, in the next section of this paper will suggest a framework which a next generation cloud based logging service. This service builds upon features of existing services while adding features to satisfy the goals as outlined in sections V and II.

A. Log Retrieval

A custom agent based approach along with support for traditional sys log such as Splunk Storm or Sumo Logic is the best approach for a consistent logging tool. The service should support a tiered approach to data input. Critical data such as firewall or IDS logs should be pushed in real time, while many other systems could be compressed and pushed on a schedule. In either case, all data should be sent to the service via SSL by default.

B. Log Storage on the Cloud

The logs should be fed into Hadoop cluster for HDFS storage as well as kept in the original format but saved in a highly compressed state for later retrieval for audit purposes. Upon transfer to HDFS, the data should have some automatic analysis of data or log type and issue customer parsers to extract the relevant data. Splunk Storm provides a similar pre-classification for uploaded log files.

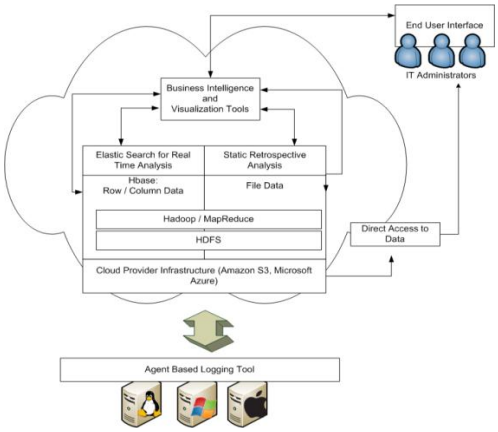

Figure 9: Proposed Architecture for Cloud Based Logging Service

C. Big Data Analytics

Table 2: Analysis of Commercial Cloud Based Logging Services

	Long term	Storage Growth	Storage Access	Consistent Logging Tools	Security and Auditability	Data Analysis
Papertrail	No, searchable for only 4 weeks max	Up to 500GB/month	Archive available to Amazon S3	Compatible with syslog, custom syslog daemon available. 3rd party tools for Windows required	SSL not default. No authorization required for systems. No granular access control	Little to none
Loggly	No, searchable for only up to 90 days max	Plans beyond 360GB/month require a sales engagement	Archive available to Amazon S3	Based on syslog, third party tools for Windows integration	Not SSL by default. Authorization required. No granular access control	Graphing available via cli
Logentries	Yes, unlimited storage and indexing	Yes, unlimited storage.	Downloading individual logs, but not all log data.	Custom logging agent available for all platforms. Syslog compatible.	SSL by default. Authorization required. No granular access control.	Graphing available but limited use.
SplunkStorm	No, 1TB max data.	No, 1TB max data.	Data is unavailable to download or archive once uploaded.	Custom logging agent available. Compatible with many logging types.	SSL by default. Authorization required. No granular access control.	Graphing available, useful interface and analysis tools
SumoLogic	Yes, but costly	Yes, allows for over 1TB of data per day	Data is unavailable to download or archive once uploaded.	Custom logging agent available for all platforms. Syslog compatible.	SSL by default. Authorization required. Somewhat granular access control.	Graphing available, useful interface and analysis tools

Similar to Liu's proposal, Hadoop should be used for Map Reduce analysis using static and defined criteria to classify and find data. Some of the approaches that Liu uses to detect malware could even be implemented in such a system if client based logging were enabled. Retrospective analysis against static rules could be continuously run and performed against the dataset. In addition to malware detection, business intelligence, market forecasting, and consumer behavioral analysis are all types of data that could be performed using this part of the system. Sumo Logic has a similar featured with their Push Analytics. Intrusion detection could also be incorporated retrospectively or actively with appropriate intrusion detection logs. For example, an IDS could enable the coordination of 'actionable' analysis that would kick off automated searches/analysis based on seeing a specialized input parameter from certain sources. In such a system, when spam was detected outbound from an IDS, the system could perform analysis and identify which server or users have been compromised by searching for any unexpected IP addresses accessing accounts or systems. Additional retrospective data mining rules would be determined based on the user's needs and the types of logs being uploaded.

D. Real Time Analysis

To perform real time data analysis, other parts of the Hadoop ecosystem need to be in place to meet all necessary requirements from IT administrators. First, the Hadoop implementation should use Hive for ad-hoc query and analysis of the data stored on Hadoop. Secondly, a no SQL Database system such as H Base should be implemented as well to allow for random reads. This is critical, as a traditional Hadoop Map Reduce job would require searching across all documents in the system when it would be much faster to only search the logs that the end user is requesting. While experiments such as Yu's work show that Hadoop Map Reduce can outperform a traditional RDBMS, with the potential for several hundred terabytes of data, a Hadoop Map Reduce job could take hours or days to perform and finish. Additionally, a distributed search index should be in place as well to increase the real time querying capability even further. If being built on Amazon, the new Amazon Cloud Search service [30] could be used as a search platform. Otherwise, the use of Elastic Search would provide acceptable real-time search capabilities [31].

E. Interface and Visualizations

The ease of text based search driven by a web browser such as that by Splunk Storm is ideal. As both Splunk Storm and Sumo Logic have similar usable interfaces, design features from both should be considered and heavily borrowed for the combination of aesthetics and utility. Utilizing the cloud compute resources to display nice graphs, charts, and visualizations on both traditional desktops and mobile clients is important. Visualizations are more useful than raw text data for quick data analysis and event correlation, and can help eliminate the problem of information overload. Useful and interactive graphs and charts can allow for deeper exploration of data while not overwhelming the user. An example of this would be the sunburst visualization of security data as proposed by Patton et al. Graphics can be used to show known problems in when scrolling through log files on the interface. When known errors are logged, a graphic could be shown that links the administrators to known fixes to common error problems. Data should be made available via cloud storage such as S3 for additional analysis and for long term archival of data. The data should be kept in its original form as well as in the indexed form.

F. Proposal Framework

An architecture framework proposal for such a system is provided in Figure 7. It includes client based agents across multiple platforms. Data is securely transferred to the cloud provider and saved in both original and compressed

form, and in HDFS. To mitigate the potential negative impacts of running Map Reduce jobs against thousands of small files as seen in Liu's research [22], files of similar types would be appended whenever possible. Data is stored and processed on both standard HDFS Map Reduce analysis as well as stored via H Base with Elastic Search for real time data analysis. Specialized rules are run retrospectively using standard Map Reduce against the data. These include IDS and Malware analysis as well as common error correcting. Both sets of data are pulled into visualization and business analysis tools and presented to the end user via a website or mobile application. Data can be extracted in its original form for legal purposes or to move providers. The Hadoop implementation must contain separate HDFS instances for each customer to protect privacy. This would also allow the services to open up standard APIs to let the users run their own custom Map Reduce jobs against their data. This could be a higher tier of payment or the users could pay for the compute cycles that they consume.

VIII. FURTHER RESEARCH AND EVOLUTION OF CLOUD BASED LOGGING

With the introduction of five cloud based logging services within the past several years, this is a definite area for growth and investment. These providers should work on reducing costs associated with their services; only providing a centralized search mechanism does not seem to warrant several thousand dollars per year. The real value of such a system comes from the analysis that the services can build into it. Many of these services are more similar to frameworks and expect the end user to customize and create their own intelligence in the system. However, if the users have enough skill and knowledge to create such a system, they hardly need the frameworks that many of these services provide. In addition to relying on the cloud for quickly scalable and usable solutions, many smaller and medium sized businesses will be turning to the cloud to provide skilled personnel who understand business intelligence.

XI. CONCLUSIONS

In this paper we have provided a detailed analysis of the problem within enterprise IT organizations for maintaining and analyzing log files. We have outlined and detailed several issues within the present state of enterprise logging and suggested that a cloud based service would be ideal to solve the problem. We have given a survey of current scholarly research in the field of cloud based logging systems. Additionally we have surveyed several existing commercial solutions and outlined their weaknesses and strengths. Finally, we have proposed an infrastructure and implementation recommendations for a next generation cloud based service for enterprise IT logging.

REFERENCES

1. U.S. Department of Health and Human Services. (2013, April). *Understanding Health Information Privacy* [Online]. Available: http://www.hhs.gov/ocr/privacy/hipaa/understanding/
2. PCI Security Standards Council. (2010, October). *Payment Card Industry (PCI) Data Security Standard Requirements and Security Assessment Procedures Version 2.0* [Online]. Available:

   https://www.pcisecuritystandards.org/documents/pci_dss_v2.pdf
3. Sarbanes-Oxley Act of 2002, Pub. L. No. 107-204, 116 Stat. 745 (codified as amended in scattered sections of 15 U.S.C.).
4. National Institute of Standards and Technology (2013, March). *Federal Information Security Management Act (FISMA) Implementation Project* [Online]. Available: http://csrc.nist.gov/groups/SMA/fisma/index.html

5. Gramm-Leach-Bliley Act of 1999, Pub. L. 106-102, S. 900--106th Congress.
6. M. Bailey, *The Economics of Virtualization: Moving toward an Application-Based Cost Model*, (2009, Nov.). IDC and VMware Whitepaper [Online]. Available:http://www.vmware.com/files/pdf/Virtualization-application-based-cost-model-WP-EN.pdf
7. Gartner, Inc. (2012). *Forecast: Consumer Digital Storage Needs, 2010-2016* [Press release]. Available: http://www.gartner.com/newsroom/id/2060215
8. EMC Corporation. (2011, June). *The 2011 IDC Digital Universe Study* [Online]. Available:

   http://www.emc.com/collateral/about/news/idc-emc-digital-universe-2011-infographic.pdf
9. G. Shields (2007, Aug.) "*Sys log 20 Years Later*," Redmond Mag [Online]. Available:

   http://redmondmag.com/articles/2007/08/01/syslog--20-years-later.aspx
10. Central Intelligence Agency. (2013). The World Fact book, Section: World Communications. [Online database]. Available: https://www.cia.gov/library/publications/the-world-factbook/geos/xx.html
11. The Apache Software Foundation. (2013). Apache HTTP Server Version 2.4, Log Files [Online]. Available: http://httpd.apache.org/docs/2.4/logs.html
12. N. Olivarez-Giles. (2013, August 25). Face book had 1 trillion page views in June, according to Google. *The Los Angeles Times*, [Online]. Available: http://latimesblogs.latimes.com/technology/2011/08/facebook-1-trillion-hits-google.html
13. C. Lonvick (2001, August). The BSD syslog Protocol. IETF Network Working Group. RFC2164 [Online]. Available: http://www.ietf.org/rfc/rfc3164.txt
14. R. Gerhards (2009, March). The Syslog Protocol. IETF Network Working Group. RFC5424 [Online]. Available: http://www.ietf.org/rfc/rfc5424.txt
15. A. Fox, R. Griffith, et al. Above the Clouds: A Berkeley view of Cloud Computing. *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley*, Rep. UCB/EECS, 28, 2009.
16. M. Badger, T. Grance, R. Patt-Corner, J. Voas (2009, May). Cloud Computing Synopsis and Recommendations. *National Institute of Standards and Technology* [Online]. Available:

   http://www.nist.gov/customcf/get_pdf.cfm?pub_id=911075
17. S. Garfinkel, *Architects of the Information Society: Thirty-Five Years of the Laboratory for Computer Science at MIT*. Cambridge, MA: The MIT Press, 1999.
18. H. Yu and D. Wang. (2012, July). Mass log data processing and mining based on Hadoop and cloud computing. In Computer Science & Education (ICCSE), 2012 7th International Conference on (pp. 197-202).
19. J. Dean and S. Ghemawat. (2008). Map Reduce: Simplified Data Processing on Large Clusters. Communications of the ACM, 51(1), 107-113.

20. I. Ray, K. Belyaev, M. Strizhov, D. Mulamba, and M. Rajaram. (2012). Secure Logging As a Service—Delegating Log Management to the Cloud. In IEEE Systems Journal, Issue 99.
21. D. Dolev and A. Yao, "On the Security of Public Key Protocols," IEEE Trans. Inform. Theory, vol. 29, no. 2, pp. 198–208, Mar. 1983.
22. S. T. Liu, and Y. M. Chen (2011). Retrospective Detection of Malware Attacks by Cloud Computing. International Journal of Information Technology, Communications and Convergence, 1(3), 280-296.
23. Logs provided by Andy Baio of Waxy.org. Available: http://waxy.org/2008/05/star_wars_kid_the_data_dump/
24. 7zip file archiver. http://www.7-zip.org/
25. Papertrail. https://papertrailapp.com/
26. Loggly. http://loggly.com/
27. Log entries. https://logentries.com/
28. Splunk Storm. https://www.splunkstorm.com
29. S. Sorkin. (2011). Large-Scale, Unstructured Data Retrieval and Analysis Using Splunk. An Easier, More Productive Way to Leverage the Proven Map Reduce Paradigm. Splunk Technical Paper [Online]. Available: http://www.splunk.com/web_assets/pdfs/secure/Splunk_and_MapReduce.pdf
30. Amazon Cloud Search. http://aws.amazon.com/cloudsearch/
31. Elastic Search http://www.elasticsearch.org