Exploiting Flaws in Big Data Systems

Michael G. Brown, Paolina Centonze
Iona College 715 North Ave. New Rochelle, New York 10801
mbrown3@gaels.iona.edu
Iona College 715 North Ave. New Rochelle, New York 10801
pcentonze@iona.edu

ABSTRACT

This journal discusses the relevant security threats, vulnerabilities and prominent techniques to securing data lake [39]. Over the last few years, Big Data solutions to data processing have made remarkable strides in the how data is interpreted. Because of its ubiquitous nature and ability to handle various data types, it has become a valuable tool and asset to many other fields including bioinformatics solutions, marketing solutions, and social media. Ethical infractions also play a role in data processing and because of this, it is critical that comprehensive security solutions are determined and enforced under the appropriate conditions. Although modern security techniques are well developed, they still leave room for scrutiny as vulnerabilities continue to unfold. In this paper is the examination and comparison of modern ethical exploit techniques as well ass techniques to promote greater authentication and overall perimeter security. The scrutinization of the current methods met through exploitation, be it physical or conceptive, is imperative in defining the understanding and verification of definite secure solutions.

Indexing terms/Keywords

Big Data, Data Lake, Cloud Computing, Hadoop API, RESTful cluster, Hortonworks, Apache, Ambari, Perimeter Security, Authentication Security. Kali Linux, DDoS, Brute Force.

Academic Discipline And Sub-Disciplines

Computer Science, Information Systems, Data Analytics.

SUBJECT CLASSIFICATION

Computer Science Subject Classification.

TYPE (METHOD/APPROACH)

Provide examples of relevant research types, methods, and approaches for this field: E.g., Historical Inquiry; Quasi-Experimental; Literary Analysis; Survey/Interview

INTRODUCTION

The omnipresent field of Big Data raises ethical security concerns that should be considered during the creation and maintenance of analytic infrastructures. As the use of Big Data continues as a developing solution to consequential problems of modern analytics, security should be kept as an essential entity to avoid the misappropriation and abuse of private data. In doing so this grants safety and privacy to those who take part in analytic examinations. In order to better understand how to protect layers in a Hadoop environment, cross-examination is a key component to gaining insight into the cofunction of the appropriate security measures. This in turn demonstrates in exposition the issues with modern Hadoop security as well as the benefits and ways to promote greater privacy.

Big Data Privacy Background

Big Data and the Internet of Things (IoT) has long been an elusive yet omnipresent field that continues to contribute and alter the study of data computing. Most average individuals are not aware of its presence, yet are contently affected by it. Most social media websites including Facebook, Twitter, and Google+, all use Big Data analytic solutions. Companies that market through trending data such as advertising and news agencies also use Big Data to amend new products and content. User data has become a precious commodity among corporations and analysts that assess data. It is because of this that user data falls into a category of ethical infractions that requires continuous attention and is constantly updated. Because the field is still evolving and has ameliorated mainly over the past several years, handling user private data is a grey area in the larger picture. There are Information Privacy Laws that impose sanctions over private user Data and these laws are carried out amongst some of the European Union as well as the United States [40]. In the United States their are various acts which aim to assist in the regulation of private data, but whether or not these Acts are enforced is not apparent. Furthermore, handling and manipulating user private data is and always will be an ongoing concern that requires constant attention. When the average person signs up for a social media site or site that uses user accounts, rarely do they take into account that their private data can be at risk. Even more alarming is the fact that the user has no control over where their data is placed or whom it is available to. The user does not have any say as to where the data goes, what it is used for, or how it is retained. The user naively places trust in the company rather than themselves and depend on corporate data policy, which can change at any point in time at their discretion. For most instances this is acceptable as companies customarily handle user private data with care, but there are unfortunate outliers in which user private data has been managed irresponsibly and cause irreversible damages to the client's wellbeing.

Twitter is an excellent example of a networking service that carries real time trending data. This is a remarkable resource for trending news and topics that can be distinguished through different populations. This gives marketing agencies a chance to promote their products first hand to target demographic audiences with more refined holistic data. User data on social media contributes to which products are advertised and which advertisements are available in a specified area. Whether or not realized, user private data is being trafficked at all times. Companies such as Gartner Inc. offer data analysis and insights as a service. They are able to manipulate extremely large quantities of data and examine it for conclusions. A client company can then utilize these results effectively.

Rather than possessing uneasiness over the protection of private data, users should have the luxury of being at ease. This is the starting point, as well as the breaking point for user endpoint private data security. No matter how overwhelming a corporation may seem, they may still depend on data on an individual basis. It is and should be the responsibility of a corporation to handle user private data with finesse and it should be kept private under all circumstances.

Being that this information data is digital rather than physical, it is this characteristic that allows it to be more difficult to regulate. While some laws in effect help to control the flow of user private data, sanctions should be amended to ensure that policies are enforced. There are ethical infractions and questions that pose no clear meaning when it comes to user private data. Their should be strict set standards as to what exactly user private data entails, is private to whom, if anonymous data be anonymized if it can be retraced and retainment policies. Moreover, a universal definition and distinct guidelines should be set in place that companies must adhere to.

Big Data Background

The terminology of Big Data is used loosely; this is because the field is broad in the study, use, and analysis of large data sets. Big Data refers to extremely large quantities of digital data. Usually so large that it requires more resources than traditional network tools tend to offer. According to IBM, there are over 2.5 quintillion bytes of data created on a daily basis [22], [25]. That is 2.5 x 1018 which is the equivalent of 2,500 petabytes or 2,500,000 terabytes. Data is everywhere, it is all around us and in this day and age it would be difficult to escape. Data is largely accumulated on a regular basis. But utilizing this data is a different story. It would take an eternity to process such large amounts of data yet processing in its entirety would be a feat in itself. Most network infrastructures do not nearly have the capability to undergo such a large operation. When creating a capable infrastructure data governance, retainment, scalability, fault management processing power and security should betaken into account.

For most instances, data is not processed in quantities as large as this, but is in fact processing extremely large quantities of data. In order for large amounts of data to be distributed and processed, a network infrastructure standard had to be constructed as new tools that offer solutions are necessary. Big Data is usually comprised and classified by Volume, Variety, Velocity and Veracity [14], [19]. These four Vs (Figure 1) account for the components that make up the concept of Big Data. Volume referring to the quantity of data sets, Variety referring to the data type (i.e. Structured, Unstructured, etc...), and Velocity referring to the timing of data processing (i.e. Real-time, streaming job). This was more or less the initial conception until later Veracity was supplemented, accounting for the bias and uncertainty of data. An even superior conception comprising Big Data would derive from Davi Ottenheimer's Hadoop Security: Seven Ways to Kill an Elephant speech during the 2014 Black Hat Europe conference in which he discussed "Four Rs". Range of sources, Raw formats, Real time, and Re-ID [26]. Emphasizing on Re-ID is the added notion of identifying the value of secure data through the vulnerabilities it may exhibit. This added focal point on security is what is necessary to ensure the privacy of data.

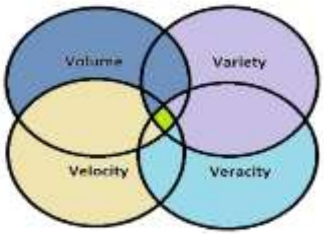


Figure1: The Four Vs of Big Data.

Hadoop is a powerful open source framework that is used for local as well as enterprise data processing solutions [1]. The Hadoop framework is largely customizable and capable of processing extremely large quantities of data in parallel. One might believe that expensive hardware is necessary to handle a data mining operation, however this is not the case. Hadoop was created with commodity hardware in mind, so it is capable of running on less costly equipment, however this is highly debatable on the circumstance. It allows even the common user to diagnose data from their commodity hardware at home. While running on less expensive commodity hardware does have its drawbacks, mainly quickness, it allows users to parse and identify data trends while reducing expenses.

The Hadoop framework was created by Douglass Read and Mike Cafarella but is now managed and developed by the Apache Software Foundation [14]. The Hadoop ecosystem utilizes various tools to maintain and perform operation on data sets. This includes Ambari, Zookeeper, Hbase, Hive, Yarn, HDFS (Hadoop Distributed File System), etc, [19]. These tools each have their own respective functions that contribute to the workflow of a Hadoop environment. MapReduce is an algorithm by Jeffrey Dean and Sanjay Ghemawat written in a Google research publication in 2004 [27]. The algorithm, written in Java, calls for a multi-stage process known as the "map" and "reduce" phase. These two phases operate in unison and are capable of parsing data and producing results. Input data is mapped to respective keys depending on the mapping algorithm and from that the key value pairs are aggregated into groups during the reducing phase. These groups are used to construct the results. Both the map and reduce functions are designed to operate in a distributed system and multi-node clusters allow an advantage in the available computing power. The ideology of the MapReduce functions also allows fault tolerance, which grants machine failures to be less of an issue. Now addressing some of the various tool functions in the Hadoop Ecosystem. The Hadoop Distributed File System (HDFS) is a file system used by Hadoop for application data that is capable of operating under batch configuration supporting large data sets. HDFS is also highly fault tolerant and capable of undergoing multiple hardware failures while maintaining data retention. HDFS can operate and be accessed across multiple nodes. There are two main types of nodes, name and data nodes. As you can see from the diagram below, Hadoop is a great way to parse, manage, analyze, and maintain data sets and has tremendous use.

Big Data Security Overview

When dealing with sensitive data it is important that the necessary steps be taken to secure the data set. Both physical and software measures must be set up in order to maximize data privacy and decrease the attack surface area. Because the data sets can be extremely large they more than likely require additional security enforcement. Security concerns were not initially taken into account during the creation and development of the framework. It is because of this that the Hadoop ecosystem lacks the interdisciplinary security that it requires it to be most efficient. Security approaches were later supplemented over the pre-existing framework, which attempted to solve and provide a comprehensive perimeter security solution. From a security point of view, it would be best to innovate security solutions during the creation and development of software because this can heavily decrease the vulnerabilities and susceptibility to ongoing attacks. Big Data infrastructures have long suffered from vulnerable security concerns and attacks that caused data hemorrhaging and private data to be unwilling shared with an attacker. This is especially a worrisome concern to health care providers and the data they possess on individual patients. Records such as these and other user private data should have strict security policies and be managed accordingly. Who has access, can view, delete, and distribute data should all be managed on a trusted network.

Starting from the most basic approach, the first way to managing a secure network would be to trust the individuals who have access to it. Every single person that is able to manipulate or view the data should be trusted. If their is a possibility that someone who accesses the data is not trusted, measures should be put into place to enforce policy/access control as well as the balance of power over the network. All part of the network infrastructure should be trusted as well which ensures that only trusted hardware can access the data sets. Lastly a comprehensive security mechanism should be set in place that addresses both policy control and authentication to minimize the attack surface of the data. Cloud infrastructures on the other hand are not localized and therefore more difficult to maintain security. Big Data networks would be increasingly more difficult to secure over WAN. Because Big Data infrastructures do not use the traditional network tools, it can be even more susceptible to attacks when configured adequately. Thinking of the transmission of large quantities of data across a network, it would be clear to say that there can be several points of weakness. How data is transmitted, stored, and accessed are all areas that require several layers of protection.

Testing and Analysis

During testing experiments, analysis gained insight into the route of exploits and attack vectors that a hacker may use to gain access, disrupt services or modify data using open source technologies. This promotes modern perimeter, authentication/authorization techniques as well as data governance when securing enterprise systems. This results in a comprehensive overview of system security and integrity that is beneficial to Hadoop infrastructures and data lakes during configuration.

Testing Configuration Specifications

Two hardware machines were used for a test setup. Test configuration setup utilizes Hortonworks HDP 2.3 as a virtual cluster configuration with Ambari. Machine A was imaged to a Windows 10 OS and VirtualBox was installed to mediate a virtual machine with Kali Linux. Machine B was also imaged with Windows 10 and Virtual Box was installed to mediate a virtual machine with Hortonworks Data Platform. The data connection between the VM and physical machine were then bridged to give the VM access to my Ethernet connection. Specifications for both physical and virtual machines are shown below in Figure2.

Machine A (HP Desktop)	Machine B(Dell Desktop)
Specifications: • Windows 10 Home V1511 Build 10586.122 • X64 Intel Xeon E5-2609 @2.4GHz • 250Gb HDD • 16Gb DDR3 Ram • Virtual Machine Specifications: • Kali Linux • 4096Mb • Processors: 2 • 30Gb Virtual HDD	Specifications: • Windows 10 Home V1511 Build 10586.122 • x86 Intel Core i7 @2.93GHz • 12Gb DDR3 Virtual Machine Specifications: • Hortonworks Virtual Box. • 40Gb HDD • 12Gb DDR3

Figure2: Physical and virtual machine device and configuration information.

Assuming that an attack can potentially occur on sight and within the same LAN as the nodes on a cluster, an attacker may be able to interact directly with these nodes, therefore it would be accurate to test all security interactions between the client and a server. Hadoop is supported on Linux based kernels although it is also capable of running on FreeBSD, Mac OS X and Windows. It would be safe to assume that it also inherits the security vulnerabilities of the distribution that it is configured on. Hadoop is commonly configured on Linux based systems so for this experiment RHEL will be used.

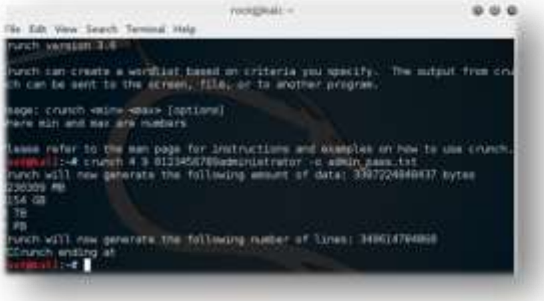
Rather than utilizing a large-scale data lake, HDP 2.3 will play the role of a Hadoop cluster with Hadoop secure off. With Hadoop secure mode not in effect, it mitigates the potential authentication safety that Kerberos principals commit. This entails that end users are not authenticated before directly accessing Hadoop services. Exact configuration of setup is seen in Figure 1. Kali Linux was chosen as the platform to dispatch and identify potential exploits based on its availability and wide array of tools[3]. With Machine A running Kali Linux and Machine B running HDP 2.3, attacks were invoked on Machine B from Machine A using multiple penetration testing tools. The purpose of this was to engage in live attacks to further develop understanding of the routes a malicious user may take to compromise a system.

Nmap is a network mapper that allows the user to view hosts and services on a network [6]. Using Nmap, network configuration such as IP addresses and open ports can be identified in real time. In order to locate Machine B on our LAN, Nmap host discovery can be used to find its IP, open ports, and which services are currently running. On Machine A executed from the terminal: nmap –sP xxx.xxx.x.x/xx. This command will return live IP addresses in the specified range assigned with the same subnet using SYN packets to ping. The command nmap xxx.xxx.xxx.* can also be used to scan a range of IP addresses. Once the IP address of Machine B is identified, the open ports can be examined with the command: nmap –sP xxx.xxx.xxx.xxx. This returns a list of open ports and services for the specified IP address. Depending on how Hadoop services are configured by the administrator, services may run on particular portsThe scan shows that Ambari-server services are running on port 8080. It was most likely chosen to operate on port 8080 due to the fact that it is out of the range of most services. There are also various TCP ports open as well as SQL and Postgres on some other ports. Ambari Server is an influential service due to the fact that it grants user access to the cluster management through a web portal. From here, various services and configuration can be managed making it a valuable asset to cluster configuration and running jobs.

If a cluster were to be configured on separate physical machines, this would also increase the attack surface area of the services. Due to Hadoops scalability, enterprise configurations are usually mapped across several layers of hardware and software causing security concerns to become even more critical, especially if nodes are spread off site. Theoretically if an attacker were able to attain the network information of independent name and data nodes, then they would be able to perform a Man in the Middle attack and without encryption, this would compromise data security. This type of attack has occurred in the past and would work in theory, but since cluster nodes use SSH for remote network services, this would be incredibly difficult today unless an exploit in SSH can be found or the attacker gets the target machine to accept their public key. Man in the middle attacks may also occur between the cluster and other end systems but RESful architectures using HTTPS mitigate this attack. Knowing the IP addresses and port configuration of a cluster leaves options for an attacker. Attacks such as DDoS, password attacks, OS attacks, eavesdropping, and sniffer attacks are real world attacks that plague large data lakes and there is always constant threat of breach.

Testing N.1: Dictionary Password Attacks

On Kali Linux, there are tools such as CeWL, CUPP and Crunch that are capable of generating arbitrary usernames and passwords [28], [29], [30]. There are also word lists that exist online tailored to common user credentials. Brute force dictionary and exhaustive key search attacks can then be triggering through various end systems allowing a user unauthorized access into an account. Using Crunch, password lists were generated that would evaluate user access into the Ambari Web console. With Crunch initialized in the Kali terminal a command can be formatted to generate a password list specifying password length, uppercase, lowercase, and character sets: crunch <min-len> <max-len> [options]. A test username and password was created for Ambari as well as a password list for that user with: crunch 4 9 0123456789administrator -o out.txt. This command will generate a password list in file out.txt with passwords ranging from four to nine characters with a character set of numbers and letters in the word administrator (Figure 2). Depending on the parameters set by the Crunch command, time to complete the password list is a function of password complexity and hardware. Passwords lists generated with increasing character count, character set size and thoroughness, increase the time of generation. This is no surprise and one of the reasons that brute force attacks are incredibly time consuming. The less complex the password, the more likely a password match due to their being less combinations and probabilities. Using a combination of password generation tools would be the most efficient way to create a tailored list of plausible passwords. CeWL is a password generator by DigiNinja, which given URL will generate a word list using the unique words on the page. CUPP is another password generator but instead of generating passwords through character sets, it will ask the user pertinent information about the victim and use those responses as parameters. Using the THC Hydra command to brute force remote authentication services on Ambari: hydra xxx.xxx.xxx.xxx:8080 -l <username> -P /path/passwds.txt and a combination of CeWL and CUPP password lists, access was able to be gained to the Ambari user test account. This entails that the shorter the mock password, the more insecure and the easier access can be gained. Passwords are one of the most vulnerable areas of any infrastructure and safe guarding them should be a top priority. Password complexity is a significant factor in safeguarding an account and rather lengthy passwords enforce security measures. Longer passwords with greater character sets increase the string matching complexity. In this case the total number of passwords in a generated list for a set of characters S would be $\sum_n^x S^n$ where n is the minimum password length and x is the max password length. Time complexity for linear string matching would then be a max of O(n) and O(1) in the best case scenario. Depending on the chosen character set, because there is only a 1/S chance of a string match if $pass \in P$, there is a relatively low probability of a password match given the time and spatial complexity of brute force algorithms and the password meets security requirements. This test also consists of a bias being that the password list and the test user account were created by the same individual, but proves brute force is a viable method of approach to spoofing authentication given that there is relatively low password complexity or even contains duplicates.



```
runch version 3.6

runch can create a wordlist based on criteria you specify.  The output from cru
ch can be sent to the screen, file, or to another program.

sage: crunch <min> <max> [options]
here min and max are numbers

lease refer to the man page for instructions and examples on how to use crunch.
[illegible]:~# crunch 4 3 0123456789administrator -c admin_pass.txt
runch will now generate the following amount of data: 3397224840437 bytes
230309 MB
154 GB
 TB
 PB
runch will now generate the following number of lines: 348614784868
^Crunch ending at
[illegible]:~# 
```

Figure3: Password Generation with Crunch.

On Linux systems it is possible that an attacker may chose to target the location of stored user passwords to target a specific user account. Theoretically if an attacker were able to gain access control of a Linux environment, they may choose to target the /etc/shadow file which contains a list of hashed passwords. Taking a look at the /etc/passwd file to view login information, there is various account information such as usernames and passwords however, the password is most likely represented by an "x" which signifies that it is encrypted. Once a user password is created, it is hashed and then stored in /etc/shadow. Only the hashed version of the password and salt is saved to increase security efforts due to the fact that hashes cannot be reversed but theoretically there are ways to uncover the shadow file. Rainbow tables are notable for their ability to precompile and check match hashes in a table look up. John the Ripper is a password cracker available for various operating systems widely used for discovering weak passwords [31]. Using John the ripper on Kali terminal, the command: john unshadow <passwd> <shadow> > <out_file> will attempt to join the information from both files and save it in one output file. John contains a default password list that is relevant for popular passwords although other password lists can also be used. The John command: john --wordlist=/usr/share/john/password.lst <out_file>. Hash and salt information may then be displayed as well as status information. If John was able to successfully crack the passwords, it can be viewed with: john --show <out_file>. This was run this previously on a separate Linux installation with success, but the shadow file and passwd file on the machine with this virtual cluster did not return cracked passwords. Rainbow tables are also an alternative method for password cracking hashes.

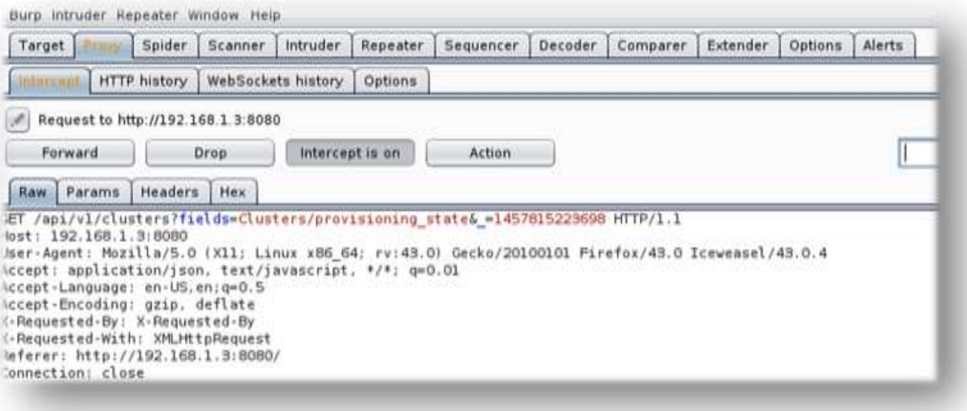
Metasploit is an open source network penetration tool designed with the ability to verify and dispatch exploits on desired hosts [4]. Payload software, such as Meterpreter, can also be utilized within Metasploit to grant functionality and network control of a host once exploited. This may allow an attacker to upload backdoor software granting them future accessibility compromising security of the system. Armitage is an open source front-end graphical user interface run in unison with Metasploit to identify host machines, appropriate exploits based on the host machines environment, and dispatch selected exploits. First initializing Metasploit and then Armitage, Armitage will prompt the user to accept initialization settings and start the Metasploit's RPC server. Once Armitage is done loading, the user is presented with the main UI that they may coordinate attacks from. The target machine IP or range can then be added through Hosts > Add Hosts and and Nmap scan may also be run from this menu with Hosts > Nmap Scan, Intense/Quick scan, assuming that the target machine is on the same network. Selecting Machine B from the available hosts, Armitage can distinguish its native OS as Linux. Now that the OS has been identified, Armitage can query the Metasploit database for corresponding exploits that may be pertinent to an attacker. A service scan can be invoked to identify which protocols or services are operating on which ports. Using the folder browser exploits can then be selected and invoked on the target machine. Through Metasploit there are various SQL, SSH and HTTP attacks that are identified as being provocative to the target machine. Using Armitage an SSH brute force attack was invoked with various password lists created with Crunch. This method will also work depending on root password complexity. Once the Armitage is able to link a connection with SSH, a payload can then be uploaded to the target and the targets file hierarchy can be browsed. This is clearly an invasive practice that would allow an attacker direct access and control over a host machine. With respect to a Hadoop cluster, this would grant an attacker the ability to control or disrupt services and even pilfer information. Aside from SSH brute force, Armitage also has a feature that would allow an attacker to invoke multiple matched exploits in tandem. Weather or not Metasploit will find an entry point into the target machine depends on a great deal of security factors such as if the attacks are OS specific, protocol specific and even version specific. This is known as a Hail Mary attack. While Triggering a Hail Mary on my target machine I was expecting it to find a way to infect the target machine but it did not. An attacker may also be able to dispatch a reverse TCP payload, which is a reflective injection attack.

Testing N.2: DDoS

DDoS are some of the most wide spread attacks that occur on large-scale infrastructures. HPing3 is a pen-testing tool native to Kali Linux that is capable of dispatching custom TCP packets for reflection attacks [11]. It also will return ICMP responses if the target sends one. Using HPing3, SYN packets could be flooded to the port of a specific IP address. If an attacker is unable to gain entry into a protected cluster, they may opt to other means of disabling it. Using HPing3 in the Kali terminal, executing hping3 -S –flood -V –p 8080 <Machine B IP>. -S will set the SYN Flag and -V will run the command in verbose mode. Attackers that use this method will usually spoof their IP and send from roaming IP addresses making it increasingly difficult to track them. While configuring a Hadoop cluster, in order to have separate hardware nodes communicate seamlessly the cluster administrator may choose to disable SELinux and IPTables to make file exchange easier. This can be partly dangerous and cause the nodes to become more susceptible to attacks when ELinux policy is not enforced. Clusters that disable this security feature should require another means of role based access and policy control.

Testing N.3: Proxy Listeners

Burp Suite by Port Swigger is a proxy tool that allows the operator to intercept HTTP requests between a client and server [8]. Burp is also widely known for having other features such as a crawler, repeater and sequencer that are used for penetration testing web applications. With Burp, it is possible to intercept and modify the HTTP POST and GET protocols between a client and a server. With Burp initialized and running on Machine A, in Proxy Options Tab, I enabled the proxy listener on interface 127.0.0.1:8080. This will route HTTP web traffic through this port, allowing Burp to intercept it. Using the Ice Weasel Kali Linux native browser, the network settings were then edited to route traffic through the new proxy. Hortonworks HDP and Ambari Server services were started on Machine B. Then from Machine A, I navigated to the Ambari web login and proceeded to input administration credentials. As a control, with intercept off, Machine A can freely sign in and out of the Ambari Web portal on Machine B with the correct credentials (User: “Admin” Pass: “admin”), the default. Now with intercept on, the Ambari web portal is visited again this time allowing the interception of HTTP code passed. When at the login, the correct user name is added, “Admin”, but an incorrect password is then substituted. After submitting the sign in form, forward was pressed in Burp Suite allowing the request to proceed. The HTTP code received back from Ambari was a GET request that contained arbitrary information about the connection pertaining to port and agent. The request also contained the GET request itself.



Request to http://192.168.1.3:8080

```
ET /api/v1/clusters?fields=Clusters/provisioning_state&_=1457815223698 HTTP/1.1
lost: 192.168.1.3:8080
Jser-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:43.0) Gecko/20100101 Firefox/43.0 Iceweasel/43.0.4
\ccept: application/json, text/javascript, */*; q=0.01
\ccept-Language: en-US,en;q=0.5
\ccept-Encoding: gzip, deflate
(-Requested-By: X-Requested-By
(-Requested-With: XMLHttpRequest
leferer: http://192.168.1.3:8080/
Connection: close
```

Figure4: HTTP GET request viewed thorough Burp Suite proxy
returned to Machine A from Ambari web portal on Machine B.

The information fetched from the host does not appear to contain login credentials such as username or password information, but it does the cluster state request noted by:
/api/v1/clusters?fields=Clusters/provisioning_state&_=1457815223698.
Leaving this state the same and proceeding to forward would cause the login to fail. Modifying the GET request so that it incorporates a not equals in place of equals will cause the portal to login provided the incorrect password:
/api/v1/clusters?fields=Clusters/provisioning_state&_!=1457815223698. Allowing the modification of information passed between a host and client should not allow an attacker to inadvertently login to an account given the wrong password.

Security Considerations for Hadoop Cluster

There are several software technologies that promise to provide a comprehensive secure network perimeter for the Hadoop ecosystem. Among these tools popularity resides among Apache Knox, Ranger, Sentry, IBM Guardium, and Project Rhino [22]. This area of study is widely controversial as the resulting effects are scarce. This document will also serve as an introductory comparison for popular Big Data security software through both methodological and theoretical comparisons. Understanding how hardware components communicate on a network cluster. When pertaining to security there are several affected areas which should be protected for both software and hardware components. Taking the property security measures and distributing reputable security software solutions on a network cluster will decrease the attack surface area and the minimize risk of sensitive data. It is also imperative that the cluster does not lack efficiency while implementing security modules, as its efficiency will impact the cost of computing. Security should be taken into consideration during the creation and configuration of nodes to best suite running job purposes. Some security setups will limit compatible software. When taking security into consideration, most ideally would be an implementation that is centralized, has policy/access control, granular auditing, perimeter security, SSL, authentication as well as protect from various attacks [13].

Security considerations may depend on if the cluster was designed for standard map-red or streaming jobs. Apache Knox is a REST compliant gateway that serves as a single point of authentication while interacting with Hadoop. They offer the services of Active directory and LDAP Authentication, Service Level Authorization, SSO/Federation, Auditing. Apache Knox also integrates with the already built in Kerberos from Hadoop Secure mode. While this appears to be the ideal solution, the drawback is that only certain services are supported. Among these are YARN, Hive, Storm, Oozie, HBase, HCatalog, and HDFS. For the most part these are a complete set of services but limit the possibilities of your cluster setup constricting the layout an administrator can create. One would also believe that a smaller set of supported services would also decrease the attack surface area and security vulnerabilities, however this is debatable. This solution would be beneficial to budgeted enterprise integration due to its support for LDAP. Granular access/policy control is also necessary for data governance security. Administrators need to control the flow of sensitive data with policy management designed for individual user groups. Apache Ranger for security administration specializes in centralized auditing and authorization methods while also providing audit tracking management. Apache Ranger is supports HDFS, Apache HBASE, Apache

Hive, Apache Storm, Apache Solr, Apache Knox, Apache YARN and Apache Kafka [33]. Both Ranger and Knox should offer optimal open source solutions. In recent years, streaming jobs has become widely popular with the increasing trend of real time data. Apache has continued to offer security solutions tailored more to the capabilities of MapReduce 2.0 with Apache Storm. Storm integrates into YARN and is not only scalable but fault tolerant and guarantees processing, which is crucial for live data. Storm has only recently been undergoing security development with Authentication (Kerberos), Authorization (The SimpleACLAuthorizer), automatic credential push and renewal, OS security and other configurations [34]. For additional distributed monitoring Nagios and Ganglia can also be installed to assist in predicting unforeseeable security concerns [36], [37].

IBM Guardium is a proprietary product marketed to protect sensitive data in a variety of environments. The key point of Guardium is to analyze sensitive data, offer activity monitoring, identify usage patterns and protect sensitive data while remaining efficient [35]. It also offers authentication, Access/Policy Control, Encryption, Auditing, encryption etc… With all that Guardium has to offer it is seemingly one of the best tools available for a proprietary solution. There is also the debate of open source vs proprietary technologies. While each has their respective downsides and benefits with regards to the field of Security, the beneficial solution should not be chosen solely on this factor. The security solution chosen should pertain to what hardware is being implemented on, the network architecture, which services need to be implemented, and the budget. The expense of computing should be taken into account during budget instances, as this will also decide the following support that will be received.

With all of the widely available software currently available to mitigate vulnerabilities, it becomes essential that they remain compatible. Version control becomes an issue when updating segmented software solutions. Various bugs seldom arise that may cause issues justifying the need for support services through companies such as IBM, Hortonworks and Cloudera. Cloudera makes ease of use of data management analytics through integrating many of these software platforms including The Enterprise Data Hub offering timely updates for version control and bug fixes while providing governance, security and management controls [38].

Conclusion

During the creation of a Hadoop cluster, security should not be underestimated but should be of high priority in a secure setting. The most important segment of creating a secure network cluster is its configuration. Simple solutions such as secure passwords can determine the difference between data privacy and a breach. To best ensure that configuration of nodes is done properly it should be done by professional administrators equipped with the skill to operate in a secure setting. Another concerning factor should be the balance of power between administrators. Their should be implications allowing administrators to balance out each others policy/access control privileges creating a more careful control of sensitive information flow. There are several security threats that tend to plague large data lakes including DDoS, password attacks, eavesdropping, sniffer and OS attacks. Through brute force, password attacks are inevitable and should be expected; so weak password testing should always be in effect when creating new users. Through tools such as Crunch, Hydra and John the Ripper, arbitrary password lists can be created and check-match them in real time. If an attacker were able to spoof into a user account granting them authentication and policy privileges, this can spell disaster due to the fact that it would be more difficult to identify as a security threat. Using THC Hydra and Crunch successfully demonstrated that brute force techniques are a real possibility. One way that perimeter security systems can counter a brute force attack is through account lockouts. DDoS attacks are also a concern depending on how the OS handles SYN packet floods. Newer Linux systems should be able to reject and ignore SYN floods but older systems may require an implementation that rejects the packets automatically. An attacker may also choose to flood SYN packets from a range of IP addresses in an attempt to create obfuscation and conceal their identity. Proxy listeners are a non-invasive approach to examining the web communication between a client and server. Leaving Ambari open on a port can lead to an unauthorized attacker attempting to spoof into an account if no perimeter security is configured. Burp Suite proxy allowed the successful modification of the GET request that was sent between the host and the server to accommodate a new GET request that would allow user sign in given an incorrect password.

Future Directions

Linux on its own is secure to an extent and modern security approaches for Hadoop that take into account these types of attacks possess a great potential and should always be installed when operating in a secure setting. Rather than introducing a layered scheme of implementing Hadoop Security, Enterprise efforts should be made which greater integrate these security features directly into Hadoop in a new standard Hadoop distribution. Rather than having the option of installing distinct components to incorporate a wide variety of security features, an integrated and centralized approach in a new distribution should be the main focus of leading industry efforts. This would not only ensure more seamless security reliability, but more efficient operation between components. Companies such as Hortonworks and Cloudera offer data processing solutions that assist in ensuring stable environments but further security integration is required in order to undermine vulnerabilities. An open source software distribution created with security in mind integrating SSL, authentication/authorization, encryption, real time activity monitoring, SSO, granular access/policy control designed for enterprise solutions would better fortify data governance.

ACKNOWLEDGMENTS

I would also like to thank Dr. Glen Mules for his encouragement and answering my many questions. I thank Iona College as well as my friends and family for granting me the opportunity to have a successful career as a student.

REFERENCES

[1] Apache Hadoop; http://hadoop.apache.org/
[2] Hortonworks HDP; http://hortonworks.com/products/hdp/
[3] Kali Linux; https://www.kali.org/
[4] Metasploit; https://www.metasploit.com/
[5] Armitage; http://www.fastandeasyhacking.com/
[6] NMap; https://nmap.org/
[7] Nmap Reference Guide; https://nmap.org/book/man-host-discovery.html
[8] Burp Suite; https://portswigger.net/burp/
[9] Offensive Security; https://www.offensive-security.com/
[10] THC Hydra; http://sectools.org/tool/hydra/
[11] HPing3; http://www.hping.org/hping3.html
[12] Sayad Ali Ahmed, Elmustafa and Rashid A.Saeed. "A Survey of Big Data Cloud Computing Security." 3.1 (2014): 78-85.
[13] Challenges, Top Ten Big Data Security and Privacy. Cloud Security Alliance. November 2012. 15 4 2016 <https://downloads.cloudsecurityalliance.org/initiatives/bdwg/Big_Data_Top_Ten_v1.pdf>.
[14] Jin, Xiaolong , et al. "Significance and Challenges of Big Data Reserch." Big Data Research 2 (2015): 59-64.
[15] M. Chithik, Raja and M. Munir Ahamed Rabbani. "Comprehensive and Coordinated Security of Knox Gateway in Big Data ." IJARSE (2015): 61-69.
[16] narasimha Inukollu, Venkata, Sailaja Arsi and Srinivasa Rao Ravuri. "Security Issues Associated with Big Data in Cloud Computing." International Journal of Network Security & Its Applications (IJNSA) 6.3 (2014): 45-56.
[17] Saraladevi , B. , et al. "Big Data and Hadoop- A Study in Security Perspective." Procedia Computer Science 50 (2015): 596-601.
[18] Sarvbhatla, Mrudula, Mouli Reddy M Chandra and Sekhar Vorugunti Chandra . "A Secure and Light Weight Authentication Service in Hadoop using One Time Pad." 2nd International Symposium on Big Data and Cloud Computing (ISBCC'15) (2015): 81-86.
[19] Sharma , Priya P. and Chandrakant P. Navdeti. "Securing Big Data Hadoop: A Review of Security Issues, Threats and Solution." International Journal of Computer Science and Information Technologies 5.2 (2014): 2126-2131.
[20] Somu , Nivethitha, A. Gangaa and V. S. Shankar Sriram. "Authentication Service in Hadoop using One Time Pad." Indian Journal of Science and Technology 7(S4) (2014): 56-62.
[21] Tankard, Colin. "Big Data Security ." Network Security 7 (2012): 5-8.
[22] Vivekanand and B. M. Vidyavathi. "Security Challenges in Big Data: Review." International Journal of Advanced Research in Computer Science 6.6 (2015).
[23] Wang, Guohui, T.S. Eugine Ng and Anees Shaikh. "Programming Your Network at Run-time for Big Data Applications." 978-1-4503-1477-0/12/08 (2012).
[24] Zuech, Richard, Taghi M Khoshgoftaar and Randall Wald. "Intrusion Detection and Big Heterogeneous Data: A Survey." Journal of Big Data (2015): 1-18.
[25] Big Data to the Enterprise, http://www-01.ibm.com/software/data/bigdata/what-is-big-data.html
[26] Ottenheimer, Davi. "Seven Ways to Kill and Elephant". Black Hat Europe 2014. https://www.blackhat.com/eu-14/briefings.html#hadoop-security-seven-ways-to-kill-an-elephant
[27] Dean, Jeffrey and Ghemawat, Sanjay. "MapReduce: Simplified Data Processing on Large Clusters". Google, Inc. http://static.googleusercontent.com/media/research.google.com/en//archive/mapreduce-osdi04.pdf
[28] CeWL. https://digi.ninja/projects/cewl.php
[29] CuPP. https://github.com/Mebus/cupp
[30] Crunch. https://sourceforge.net/projects/crunch-wordlist/files/crunch-wordlist/
[31] John the Ripper. http://www.openwall.com/john/
[32] Apache Knox. https://knox.apache.org/
[33] Apache Ranger. http://hortonworks.com/apache/ranger/
[34] Apache Storm Security. https://github.com/apache/storm/blob/master/SECURITY.md
[35] IBM Security Guardium. http://www-03.ibm.com/software/products/en/category/data-security
[36] Nagios. https://www.nagios.org/
[37] Ganglia. http://ganglia.info/
[38] Cloudera. http://www.cloudera.com/resources/solution-brief/enterprise-data-hub-solution-brief.html
[39] Terradata; Hortonworks. Putting the Data Lake to Work A Guide to Best Practices. April 2014. CITO Research. May 2016 https://hortonworks.com/wp-content/uploads/2014/05/TeradataHortonworks_Datalake_White-Paper_20140410.pdf.
[40] Sotto, Lisa J. and Aaron P. Simpson. Data Protection and Privacy. Ed. Rosemary P. Jay. Getting the Deal Through , 2014 https://www.hunton.com/files/Publication/1f767bed-fe08-42bf-94e0-0bd03bf8b74b/Presentation/PublicationAttachment/b167028d-1065-4899-87a9-125700da0133/United_States_GTDT_Data_Protection_and_Privacy_2014.pdf.

Authors' biography with Photo

Michael G. Brown earned his undergraduate degree in Computer Science at Iona College in New Rochelle New York, USA. He is currently completing his graduate degree in Computer Science at Iona College in 2016 under the supervision of Dr. Paolina Centonze. His research is centered on identifying vulnerabilities in Big Data infrastructure.

Paolina Centonze has been a professor in the Computer Science Department of Iona College since August 2011. Her areas of research include language-based security and mobile computing. At Iona College, she has been responsible for extending the Computer Science curricula into the field of Cyber Security.

Before joining Iona College, Dr. Centonze was a researcher at IBM's Thomas J. Watson Research Center in Yorktown Heights, N.Y. She has published extensively at numerous conferences worldwide, such as ISSTA, ECOOP, ACSAC, MDM, MOBILESoft, MobileDeLi. Dr. Centonze is also the author of two book chapters in the area of cloud and mobile security, which will appear in 2016 in books published by IGI Global and John Wiley & Sons. She is also the inventor of 10 patents issued by the United States Patent and Trademark Office.

Dr. Centonze received her Ph.D. in Mathematics and MS degree in Computer Science from New York University (NYU) Tandon School of Engineering in Brooklyn, N.Y., and her BS degree in Computer Science from St. John's University in Queens, N.Y.

.