Attacks and Contermeasures in a Hadoop CLUSTER

K. Daoudhiri, J. Abouchabaka, N. Rafalia

LaRIT, MOROCCO

Abstract— The present study addresses the impact of insider attacks and security issues in a Hadoop cluster at five security levels using different scenarios, from low to high-impact attacks. We investigated potential security threats that can come from compromised nodes, malicious users, and network intruders.

Index Terms— Authentication, Big data, Encryption, Hadoop, Hdfs, High availability, Insider attacks, Intrusion prevention, Kerberos, MapReduce, Network, Security.

1 INTRODUCTION

Big data has become an important topic for a large number of research areas namely machine learning, computational intelligence, the web semantic and information security. the capability of analyzing what is known as big data has a diversity of uses across numerous fields. It has good uses in healthcare, financial trading, science and research, sports, business process optimization, security and law enforcement [12].

We live in the age of big data, where the data volumes we need to work with in a daily basis have surpassed the storage and processing capabilities of a single host. Big data brings two essential challenges: how to store and work with massive data sizes, and more important, how to understand data and turn it into a competitive advantage.

Hadoop fills a gap in the maket by effectively storing and providing computational capabilities over substantial amounts of data. It is a distributed system and it offers a method to parallelize and execute programs on a cluster of machines.

Hadoop in the other hand has security issues, it lacks a security model and only provide basic authentication for HDFS, which was not very successful, since it is extremely easy to impersonate another user. This is beacause when D. Cutting and M. Cafarella started developing Hadoop, security was not exactly the priority.

The rest of the paper is organized as follows: In Sect. 2 we review Hadoop. In Sec. 3 we present the attacks used against Hadoop cluster, the tools and thechniques used to implement these attacks in detail, as well as the countermeasure for each attack. In Sec. 4 we show the other side of the coin, which is Hadoop security solutions at different levels. In Sec. 5 we give final Remarks.

- *Kaoutar Daoudhiri is with the Departement of Computer Science, faculty of science, Ibn-Tofail University, Kenitra, Morocco. E-mail: kaoutar.daoudhiri@ uit.ac.ma.*
- *Jaafar Abouchabaka is with the Research Laboratory in Computer Science and Telecommunications (LaRIT), Department of Computer Science, faculty of science, Ibn-Tofail University, Kenitra, Morocco. E-mail: abouchabaka3@yahoo.fr.*
- *Najat Rafalia is with the Research Laboratory in Computer Science and Telecommunications (LaRIT), Department of Computer Science, faculty of science, Ibn-Tofail University, Kenitra, Morocco. E-mail: arafalia@yahoo.fr.*

2 OVERVIEW OF APACHE HADOOP

2.1 HDFS

HDFs is the storage unit of hadoop. It's a distributed file system that's designed after the **G**oogle **F**ile **S**ystem (GFS) paper. Hdfs is enhanced for high capacity and works best with large files (gigabytes and more) input/output (I/O). To support this capacity HDFS provides particularly large block sizes and data locality optimizations to reduce network I/O.

Scalability and availability are also key qualitys of HDFS, accomplished in part due to data replication and fault tolerance. HDFS replicates files for a configured number of times, is tolerant of both hardware and software failure, and re-replicates data blocks on data nodes that have collapsed. Fig.1 depicts the high-level Hadoop architechture.

2.2 MapReduce

MapReduce is a batch-based distributed computing framework designed after Google's paper on MapReduce. It grants you to parallelize work over a massive volume of raw data, such as joining relational data with weblogs to model user interactions with a website. The MapReduce model facilitates parallel processing by abstracting the complexities implicated in working with distributed systems. MapReduce enables the programmer to focus on addressing business needs, instead of getting confused in distributed system complications.

MapReduce breaks down submitted work into small parallelized map and reduce workers. The role of the programmer is to define map and reduce functions, where the map function outputs key/value tuples, wich are processed by reduce functions to produce the final output.

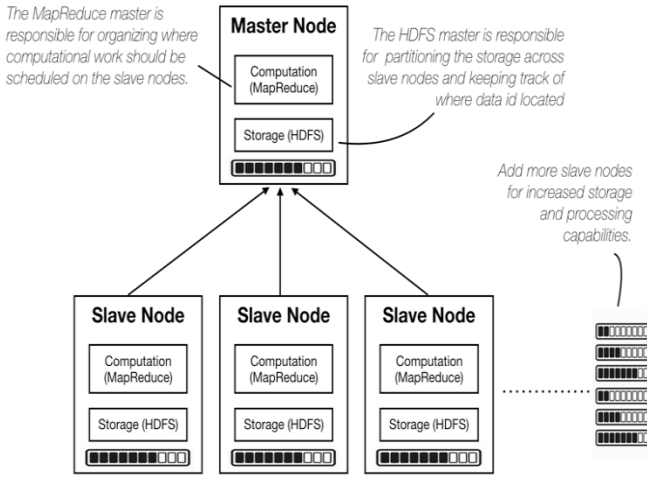

Fig. 1. Hadoop High-level architecture

2.3 Yarn

While HDFS manages the storage of the data and MapReduce handles the work of analyzing and processing data, YARN schedules all the jobs with regard to the requirements of each job and the ressources availability. Yarn contains a global RessourceManager daemon (manages the cluster ressources), and a per-applicaton ApplictionMaster daemon, witch ensures that any requested job is successfully completed [1].

Slaves on the cluster have their own NodeManager, wich communicates the status (active/down through Heartbeat messages) of the node to the RessourceManager on the master. The task or job's ApplicationMaster is placed within the slave nodes. Each slave node hosting an ApplicationMaster is managed by the ApplicationManager, this is the part of the RessourceManger daemon that assigns the job to be completed to the ApplicationMaster, witch actualy execute the job by obtaining the needed ressources.

The other half of the RessourceManager is the Scheduler, it does not only schedule a job to be completed, but also is what the Application master communicate with to get any ressources it may need. Ressources are illustrated through Container objects [1].

It is by means of YARN that MapReduce jobs are able to be accomplishes through the job scheduling and resource allocation.

3 Attacks and Contermeasures

In this section, various attacks on hadoop cluster and its countermeasures have been suggested. We used VirtualBox virtualization platform to implement a four nodes hadoop cluster, one master and three slaves, a client machine (edge node), a Cloudera client and a kali Linux attacker machine.

3.1 Port Scanning Attack

Port scanning is a prominent information gathering method that identifies wich ports and services are open and records how they respond to the queries to gather information on the target. Port scanning has an identifiable signature. A firewall or host will respond to a port scan in one of three ways: (1) Open and listening means the host indicates a service is listening and will respond, (2) Closed means the host responds by denying connections to the port, (3) No reply means that the host has filtered, blocked or dropped the request and is in stealth mode.

Method. We used the Nmap open source network exploration tool and security/port scanner [2]. To accomplish its goal, Nmap sends specially crafted packets to the target host(s) and then analyzes the responses [3].

Contermeasure. Because the information may be used as a precursor to an attack, efforts should be made to prevent port scanning attempts. Some next generation firewalls and intrusion prevention systems use adaptive behaviour, they block ports if a suspected port scan is in play.

3.2 Dictionary Attack

A dictionary attack attempts to defeat an authentication mechanism by systematically entering each word in a dictionary as a password or trying to determine the decryption key of an encrypted message or document. Hence the dictionary attack is always faster than brute-force attack.

Method. We used the Crunch [5] wordlist generator allowing to create, from certain set of characters the totality of possible combinations (the dictionary), then we passed it to the THC-Hydra [6] parallelized login cracker, wich supports numerous protocols to attack (Telnet, FTP, SSH, SMTP, mysql, postgres, etc.).

Countermeasure. Dictionary attack can be avoided by selecting a strong password.

3.3 Remote to User (R2L) Attack

A remote to user attack is an attack in wich a user sends packets to a machine over the network, in order to expose the machines vulnerabilities and exploit privileges wich a local user would have on the computer e.g. xnsnoop, sendmail, guest, Dictionary etc.

Method. This attack was conducted based on the dictionary attack by targeting the SSH protocol wich was reported open in scaning phase.

Hadoop core uses Shell (SSH) for communication with slave nodes and to launch the server processes on the slave nodes. It requires a password-less SSH connection between the master and all slaves and the secondary machines, so every time it does not have to ask for authentication as master and slave require rigorous communication.

Countermeasure. The countermeasure is based on fail2ban [7] intrusion prevention framework that protects servers from brute-force attacks, it is able to run on POSIX systems that have an interface to a packet-control system or firewall installed locally (iptables, TCP wrapper, etc.).

3.4 Computer Exploit Attack (DoS)

A computer exploit is an attack on a computer system, especially one that takes advantage of a particular vulnerability the system offers to intruders.

An exploit is a piece of software, chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability to cause unintended behavior to occur on computer software or hardware. Such behavior frequently includes things like gaining control of a computer, allowing privilege escalation, or a denial of service attack.

Method. By performing the Remote to User attack, we took control of a slave machine in the cluster, the post exploitation of this attack would be other attacks from inside the cluster.

We used the compromised machine to exploit vulnerabilities on open services, in this case OpenSSH_7.2, wich does not limit the password length for authentication. Hence to exploit this vulnerability we will send a crafted data wich is of 90000 characters in length to the password field while attempting to log in to a remote machine (Master machine) with a username figured out in earlier attack, the user enumeration attack, that OpenSSH_7.2 is also vulnerable to [8]. The impact of this exploit results in a total shutdown of the affected resource.

By exploiting OpenSSH_7.2 vulnerabilities, we were able to DoS the master machine (runing master Daemons), thus the hole cluster was completely unavailable.

Countermeasure. The issue can be resolved by updating the package OpenSSH to recent version.

3.5 Attacks on MapReduce Computation Time

These attacks, in accord with the classic CIA triad of computer security, affect the availability of the services provided by hadoop while leaving the rest intact. The availability is disturbed through the prolongation of the process through wich informations are extracted from big data. Considering the time-sensitive nature of big data, this can have consequences for any that wish to use the information they have gathered effectively.

Method. We first have to review how MapReduce completes the job to determine how an attacker, from a compromised DataNode, can impact the time taken for a job to complete. The slaves store the data blocks and the master sends commands to slaves about how to process the data. Based on the received command, the slave Nodes accomplish the computation and send the data back to the master. Thus, to affect the computing time, the attacker can either affect the computation ressources (CPU) on the slaves, or disrupt the communication between the slaves and the master.

1. Block Communication: to block communication, an attacker can turn off the machine or make it unusable. However, as mentioned earlier, hadoop is resilient to hardware failure. One way an attacker can overcome this failure mitigation scheme is to only allow the heartbeat messages (the way that the master keep track of active nodes) to be transmitted between master and slave, whilst all other forms of communication are disabled. If heartbeat messages can be transmitted successfully, the master will not be able to tell wether a slave node has been compromised or not.
   Even though all communications will be blocked the master will still send tasks of MapReduce jobs to the compromised node to complete. None of these messages will actualy reach it, leading to the disruption of job completion and thus extend completion time.
   If the distributed file system is configured to use replication, any task sent to be completed on compromised node can still be sent to the uninfected ones. This will allow for overall job to be completed despite unavailability of some blocks on compromised node.
   To implement this attack, we examined the network traffic using the tcpdump network sniffer [9], we found that ther was constant traffic between the master and the slave on the master port 9000(depends on each hadoop configuration) and 8025(default ResourceTracker port). Interestingly, the contents of packets on both of these ports reveals that both of them are involving heartbeat messages. An example of Some of the recorded traffic is illustrated in Fig. 2.
   All input traffic coming from these two ports must be blocked, this was conducted using a firewall rule added through iptables tool.
   This attack resulted in slave node no longer particpating in MapReduce jobs while master node would continue to send its tasks to complete despite this fact.
2. Delay Communication: Another way to affect MapReduce jobs completion time is delaying the communication between master and the compromised slave.
   Because communication between the master and slave nodes is done through TCP, this means delaying individual TCP packets.
   Tis attack was conducted by delaying the network traffic sent from the compromised node, wich was implemented through the tc tool [10], witch comes by default in ubuntu and allows network traffic manipulation in variety of ways including delaying the data transmission as desired here. We postponed every paquet by adding extra 2s delay.

Countermeasure. The countermeasure for these attacks, in our case, falls back to protecting the cluster nodes from remote attacks.

3.6 Attack on NameNode Availability

This attack is also conducted from the compromised slave node. It consists of putting down the NameNode daemon to put the Hadoop cluster out of service.

Method. This attack was conducted by connecting through SSH to the master machine from the compromised slave node (Hadoop lays on passwordless SSH connection), listing the running processes, identifying the process id of the NameNode, and finaly kill it.

Countermeasure. Hadoop has introduced the High Availabilty concept starting 2.x versions, to overcome the

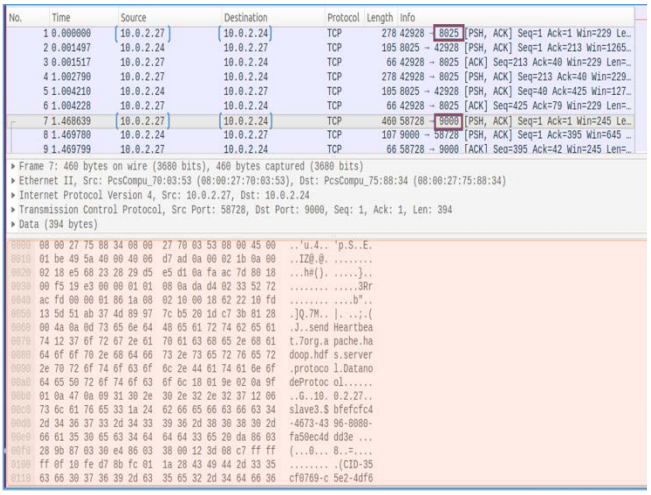

Fig. 2. Screenshot of tcpdump network traffic capture

single point of failure problem in 1.x versions. In the standard configuration of HDFS, the NameNode is clearly a single point of failure, because the moment the NameNode become unavailable, the hole cluster is unavailable.

The HA architecture resolved the disponibility issue of NameNode, by allowing the simultaneous execution on two NameNodes, in an active/passive configuration. The StandBy NameNode plays the role of Backup NameNode. When the active NameNode is down, the standby NameNode takes over the responsibility of the cluster, through automatic failover, conducted by the Zookeeper Failover Controller. By implemention High Availability Using the Quorum journal nodes, the downtime was reduced to 2 seconds.

Whereas hadoop 2.x only supports two NameNodes, in hadoop 3.x there is additional fault tolerance as it offers multiple NameNodes.

3.7 Man-In-The-Middle Attack

MITM is a kind of eavesdropping attack. An attacker comes between two hosts, and all the communication between them goes only through the attacker. So he impersonates both the parties to one another, and he may copy, alter or delete a portion of the traffic data.

MITM may be used to simply monitor the data and may not be reused also.

Method. This attack was conducted from Kali Linux machine, by listening to traffic between the client and the cluster, while a data write to HDFS was submitted, usind the Tcpdump tool. The data was captured in plain text.

Countermeasure. Hadoop offers network encryption, witch is another important aspect of security, to protect the communication in the network.

It costs some performance, but the performance impact should not prevent enabling the network encryption.

3.8 Hadoop Bypass Authentication Attack

By definition [10], authentication is the process of confirming truth or identity of an object. In the technical terms, it is a program or process which confirms user's identity, to ensure that a user really is who he claims to be.

Bypass attack: usually the root cause of an authentication bypass is either the failure of software system to impose access policies, or weakly designed authentication system architecture.

Method. To show the vulnerability of Hadoop clusters without security enabled, we set the following setup in Tab. 1.

Each user has his own home folder in HDFS protected by access list (0700), and a secret file wich must not be accessible to anyone but the owner.

Method. Hadoop have WebHDFS that provides a simple, standard way to execute Hadoop filesystem operations by an external client that does not necessarily run on hadoop cluster itself. The requirement for WebHDFS is that the client needs to have a direct connection to NameNode and DataNodes via the predefined ports. With this, it is possible to perform authentication and authorization attacks, when security is not on, through a REST API. It is also possible to have access to hadoop cluster by means of the Hadoop client.

1. REST API based attack: When security is off, the authenticated user is the username specified in the user.name query parameter when used a REST API to connect to NameNode. If the user.name is not set, the server may either set the authentication user to a default web user, or return an error response. Considering an attack on a non-secured Hadoop cluster, it is possible to perform the following curl request to get access to user's private information in Hadoop cluster:
   curl -i –L "http://<namenode@>:<port>/webhdfs/v1/?op=<operation>&user.name=<victim username>"
2. Client based attack: For this attack we use a cloudera virtual machine, where we set up an environment variable with the victim user name.
   sudo adduser <victim user name>
   su <victim user name>
   or
   Export HADOOP_USER_NAME=<user name>
   then
   hdfs dfs –fs hdfs://<namenode@>:<port> -cat <hdfs file path>

Countermeasure. Hadoop has the ability to require authentication, in the form of Kerberos principals. Kerberos is an authentication protocol wich uses tickets to allow nodes to identify themselves.

4 Hadoop Security Levels

By default, there is no security in Hadoop cluster, wich we will call Level 0, relaxed security. Here Hadoop assumes a level of trust, there may be authorization rules assigned to objects, but these rules can be easily subverted: Hadoop Bypass Authentication attack.

Then comes the Bastion Security: Level 1, a Bastion or EdgeNode limits access to the cluster. Instead of enabling connectivity from any client, a bastion is created that users log into, and this bastion has access to the cluster.

Level 2 is Access and Authentication Control: this level introduces Kerberos, who ensures that both users and services are authenticated. Kerberos is the authentication mechanism for hadoop deployments.

Level 3 is Network encryption: we demonstrated the importance of communication encryption in previous section when a MITM attack was conducted.

The level 4 is what we call the last line of defence: even though we protected our cluster from external unauthorized access (Kerberos), encrypted network communication between the cluster and clients, we still have vulnerabilities. If a malicious user gets access to the cluster's server (e.g. compromised node), he could read the data despite of ACL, he can find blocks that store the data and review the physical files at the OS level. To prevent this, we use HDFS encryption. A summary of conducted attacks in Tab. 2.

File Owner	File Name	Content	HDFS Location	Permissions
alice	alice_secret.txt	"alice secret file"	/user/alice	0700
bob	bob_secret.txt	"bob secret file"	/user/bob	0700

Tab. 1. Set up of target Hadoop cluster

Attack Type	Service	Mechanism	Effect of the attack
Nmap	Many	Abuse of feature	Identifies Active Ports
R2L/Dictionary	SSh	Abuse of feature	Gains User Access
DoS	SSh	Buffer over flow	Denies Service on One or More Ports
MITM	Many	Abuse of feature	Gain of Information
User Impersonation	NameNode	Abuse of Feature	Gain of Information
Attack on MapReduce	Many	Block/Delay	Slowing Down Hadoop Cluster
Put Down NameNode	NameNode	Kill	Shutting Down Hadoop Cluster

Tab. 2. Summary of conducted attacks

7 Conclusion

This work helps to consider various attacks on Hadoop clusters and its countermeasures before deploying Hadoop in production environments, wich are usualy connected and more exposed to external attacks.

Attacks can also come from the inside, when employees of the organization bestowed with more power and knowledge about the environment initiates such attacks. The system administrators and network managers steal the authentication data or exchange keys.

Intrusion detection system helps to mitigate such attacks. Access control mechanism, monitoring and logging must be strictly maintained.

References

[1] "Apache Hadoop nextgen MapReduce (YARN)", The Apache Software Foundation, https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html. 2018.

[2] "Nmap reference guide", nma.org, https://nmap.org/book/man.html 2018.

[3] "Nmap", WikipediA, https://en.wikipedia.org/wiki/Nmap, 2018.

[4] J. Nam, K.R. Choo, J. Paik, D. Won, "An Offline Dictionary Attack Against a Three-party Key Exchange Protocol", *IEEE Communication Lett.*, Vol.13, pp.205-207, Mar. 2009.

[5] "Crunch package description", Kali Linux penetration testing tools, https://tools.kali.org/password-attacks/crunch, 2014.

[6] "Hydra package description". Kali Linux penetration testing tools, https://tools.kali.org/password-attacks/hydra, 2014.

[7] "Fail2Ban", fail2ban.org, https://www.fail2ban.org/wiki/index.php/Main_Page, 2016.

[8] "OpenSSH Crypt CPU Consumption", Secpod, http://www.secpod.com/blog/openssh-crypt-cpu-consumption/, 2016.

[9] "TcpDump", sectools, http://sectools.org/tool/tcpdump/, 2015.

[10] "Cyber Attacks Explained: Authentication Attak", Valency Networks Blog, http://www.valencynetworks.com/blogs/cyber-attacks-explained-authentication-attacks/, 2018.

[11] "Authentication Attacks", IBM, https://www.ibm.com/support/knowledgecenter/en/SSB2MG_4.6.0/com.ibm.ips.doc/concepts/wap_authentication.htm, 2018.

[12] Alex Holmes, *Hadoop in practice*. Manning Publications Co., 2012.

[13] Shui Yu, Song Guo, "Big Data Concepts Theories and Applications", Springer, 2016.

[14] Bhushan Lakhe, "Practical Hadoop Security", Apress, 2014.

[15] Alvaro Rocha, Anna Maria Correia, Hojjat Adeli, Luis Paulo Reis, Sandra Costanzo, "Recent Advances in Information Systems and Technologies", Springer, Vol.2, 2017.

[16] Ben Spivey, Joey Echeverria, "Hadoop Security: Protecting Your Big Data Platform", O'REILY, 2015.