

Implementation of big data protection analytics in virtualized infrastructures

B.NILOUFER, S. JESSICA SARITHA²

¹M.Tech Student, Dept Of Computer Science And Engineering, Jntua College Of Engineering, Pulivendula,
Pulivendula-516390, Andhra Pradesh India

²Assistant Professor, Dept Of Computer Science And Engineering, Jntua College Of Engineering, Pulivendula,
Pulivendula-516390, Andhra Pradesh India

Abstract—Enormous Data and disbursed computing are two important issues inside the ongoing years, empowers figuring assets to be supplied as Information Technology administrations with excessive proficiency and viability. Virtualized framework in distributed computing has been a beautiful awareness for cyber attackers to dispatch propelled attacks. This paper proposes novel sizeable information based totally security examination manner to cope with distinguishing propelled attacks in virtualized frameworks. System logs and moreover purchaser software logs collected now and again from the traveler virtual machines (VMs) are put away within the Hadoop Distributed File System (HDFS). At that factor, extraction of attack highlights is performed through diagram based occasion connection and MapReduce parser primarily based ID of capability assault approaches and we practice okay-implies grouping calculation for making the attacks into bunches. Next, assurance of attack nearness is finished through two-enhance machine adapting, mainly strategic relapse is connected to discern assault's restrictive chances as for the tendencies, and conviction unfold is attached to ascertain the faith in presence of an attack in view of them.

Index Terms—Virtualized infrastructure, virtualization security, cloud security, malware detection, rootkit detection, security analytics, event correlation, logistic regression, belief propagation

I. INTRODUCTION

So as to investigate complex records and to understand designs it is imperative to soundly save, oversee and proportion a number of complex facts. Cloud accompanies an unequivocal protection project, i.e. The facts proprietor likely may not have any control of in which the records is put. The reason for this manage difficulty is that in the occasion that one needs to get the blessings of distributed computing, he/she must likewise use the allotment of property and moreover the making plans given via the controls. Consequently it's miles required to ensure the information amidst dishonest strategies. Since cloud consists of broad many-sided best, we believe that rather than giving a comprehensive solution for anchoring the cloud, it is perfect to make imperative enhancements in anchoring the cloud that will at ultimate grant us with a included cloud. Google has provided shape for getting ready plenty of statistics on product device. Apache's Hadoop dispersed file framework (HDFS) is advancing as a prevalent programming phase for dispensed computing consolidated along coordinated elements, for example, MapReduce. Hadoop, that's an open-source execution of Google MapReduce, along with a conveyed document framework, offers to the software program engineer the deliberation of the guide and the lower. With Hadoop it's far less annoying for associations to take a few to get back a few composure on the massive volumes of statistics being produced each day, except inside the intervening time can likewise make issues recognized with security, statistics get to, looking at, excessive accessibility and enterprise development.

In this paper, we consider some methodologies in giving security. We should a framework which could scale to address an intensive quantity of locales and moreover have the potential to process expansive and massive measures of information. In any case, satisfactory in elegance frameworks the usage of HDFS and Map Reduce isn't precisely enough/adequate because of the way that they don't deliver required protection efforts to make sure delicate facts. Besides, Hadoop shape is applied to take care of problems and oversee data advantageously via using numerous systems, as an example, consolidating the ok-implies with information mining innovation.

II. PROPOSED SYSTEM

Overall Framework:

The vital thought of our proposed technique is to distinguish progressively any malware and rookit assaults via all encompassing effective utilization of all plausible facts were given from the virtualized foundation, e.g., exceptional machine and customer application logs. Our proposed approach is a main information problem for the accompanying qualities of the machine and patron software logs collected from a virtualized basis:

- **Volume:** Depending on the quantity of tourist VMs and the volume of the device, the degree of the device and client software logs to be amassed can go from roughly 500 MB to 1 GB 60 minutes;
- **Velocity:** The system and purchaser utility logs are accrued gradually, retaining in mind the cease intention to apprehend the nearness of malware and rookit assaults, in like way the accrued facts containing its conduct must be prepared as speedy as time lets in;
- **Veracity:** Due to the "low and slight" technique that malware and rookit take sequestered from the entirety their essence within the visitor VMs, statistics examination needs to depend on event courting and stepped forward research.

The outline standards, which are critical in the improvement of our BDSA manner to cope with securing virtualized foundations, can be expounded as takes after.

- **Design Principle # 1** - Unsupervised arrangement: The assault vicinity frameworks have to have the potential to order ability attack nearness in view of the facts gathered from the virtualized basis after some time.
- **Design Principle # 2** - Holistic expectation: The attack vicinity framework have to have the ability to apprehend capability attacks by way of corresponding activities at the statistics accumulated from one of a kind assets in the virtualized foundation.
- **Design Principle # 3** - Real-time: The attack popularity framework have to have the capacity to discover attack nearness as right away as plausible so with recognize to the correct countermeasures to be taken directly.
- **Design Principle # 4** - Efficiency: The attack discovery framework need to have the potential to apprehend attack nearness at a high computational talent, i.E., with as meager execution overhead as may want to fairly be expected.
- **Design Principle # 5** - Deployability: The attack location framework should be promptly deployable underway circumstance with negligible alternate required to regular technology situations.

Figure 1 represents the overall implemented system of our proposed huge facts primarily based safety investigation (BDSA) approach, with the distinctive elements featured in blue. Our BDSA approach accommodates of primary levels, in particular

- Extraction of attack includes thru chart based event courting and MapReduce parser primarily based distinguishing proof of ability attack methods, and
- Determination of assault nearness by using two-strengthen device adapting, especially strategic relapse and conviction engendering.

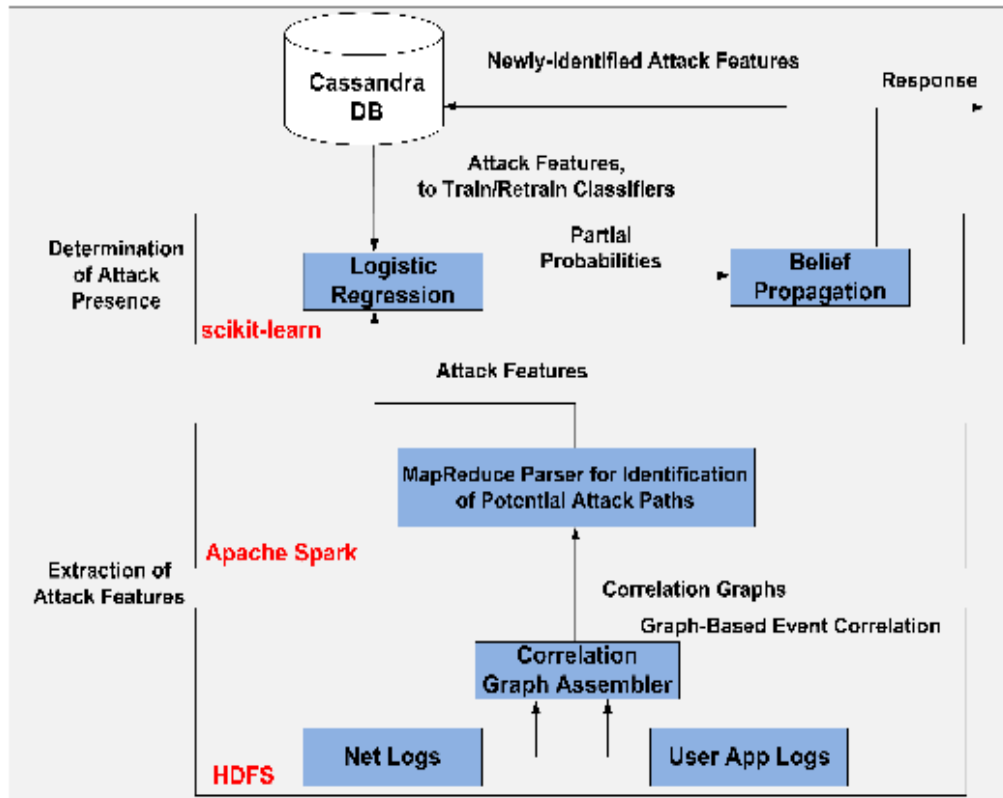


Fig. 1 Conceptual framework of the proposed big data based security analytics (BDSA) approach

Before the net discovery of attacks, there's simply a framework advent, in which disconnected getting ready of the calculated relapse classifiers is executed, that is, the placed away highlights are stacked from the Cassandra database to prepare the strategic relapse classifiers. In particular, understood vindictive and in addition amiable port numbers are stacked to put together a calculated relapse classifier to decide whether or not the upcoming/lively institutions are characteristic of an assault nearness. In like manner, understood malware and real packages together with their associated ports are stacked to put together a calculated relapse classifier to decide whether or not the conduct of an software going for walks in the visitor VM is function of an attack nearness. These prepared calculated relapse classifiers are prepared for on line use, upon the extraction of recent attack highlights, to determine whether the potential attack approaches are demonstrative of attack nearness. In the Extraction of Attack Features stage, first, it completes Graph-Based Event Correlation. Intermittently amassed from the traveler VMs, machine and client application logs are placed away inside the HDFS. By amassing the facts contained in these logs, the Correlation Graph Assembler (CGA) shapes connection charts. At that factor, it completes the Identification of Potential Attack Paths. A MapReduce display is applied to parse the connection diagrams and distinguish the capability attack ways i.E., the maximum every from time to time going on chart methods as some distance as the traveller VMs' IP addresses. This relies upon at the notion that a traded off visitor VM has a tendency to create extra hobby streams because it tries to build up correspondence with an aggressor. In the Determination of Attack Presence level, two-develop machine learning is applied, specially strategic relapse and conviction engendering are utilized. While the previous is applied to determine attack's contingent chances as for singular characteristics, the final is utilized to compute the conviction of an attack nearness given these restrictive features. From the capability assault methods, the determined highlights are handled and long gone into their strategic relapse classifiers to figure assault's restrictive probabilities concerning singular qualities. The restrictive chances as for singular houses are surpassed into conviction engendering to compute the conviction of assault nearness. When attack nearness is found out, the overseer is terrified of the assault. Besides, the Cassandra database is refreshed with the currently identified assault highlights as opposed to the elegance discovered out (i.E., attack or type), that are then used to retrain the calculated relapse classifiers.

Determination of attack presence:

The potential assault methods identified of the relationship diagram as hailed up via the MapReduce parser may be promptly recovered into the exceptional assault highlights. We allude to the stripping process as assault consist of Sorter out of attack approaches. For the warranty of assault nearness two-strengthen gadget mastering is utilized, to be unique calculated relapse and conviction proliferation. Strategic relapse gives a fast techniques for coming across whether a given take a look at facts duties to one of the two pre-characterised training, and also supporting the brisk making ready of a classifier given a preparation set, $(X \sim Y)$, which means that a development of highlights as opposed to classes. This makes it affordable for computing assault's contingent possibilities as for (wrt) singular traits. Moreover, at some thing point an assault nearness has been located, the calculated relapse classifiers can be immediately retrained steadily making use of the these days prominent assault highlights for future attack recognition. Conviction proliferation considers the restrictive probabilities to parent the conviction of assault nearness in the virtualized circumstance. This takes into attention a comprehensive way to address assault discovery, ensuring that the figured conviction exactly mirrors the probability commitments from the man or woman traits. The assurance of assault nearness incorporates of two ranges, i.E.,

- Training and retraining of strategic relapse classifiers
- Attack grouping using conviction engendering Conditional probabilities as for the traits are computed in view of the highlights noticed from the logs utilizing the prepared strategic relapse classifiers.

Utilizing any of the obtained restrictive possibilities as for singular characteristics on my own isn't always sufficient to get an entire standpoint of the attack likelihood. Consequently, perceptions of all characteristics have to be exploited to discover assault nearness. Conviction proliferation is applied to compute the conviction of an assault with the aid of taking into account assault's restrictive possibilities as for each one of the characteristics.

Algorithm 1 offers the pseudo code of the conviction spread calculation that's utilized as part of our BDSA technique. At the instatement degree, the earlier chances as for singular homes are doled out traits in light of the underlying perceptions received disconnected. Amid the life of the execution of the BDSA technique, the chances contained inside the element hubs are refreshed in mild of the refreshed strategic classifiers esteems.

Algorithm 1 Belief propagation for BDSA

Input: P_{port_change} , $P_{unknown_exec}$, $P_{in_connect}$, and $P_{out_connect}$

- 1: **Initialize:** Create the Bayesian network of attack features using factor graphs as shown in Figure 4(b)
- 2: Set the factor graphs F_{exe} , F_{in} , F_{out} , and F_{port} with the placeholder CPTs as shown in Tables 3a ~ 3e.
- 3: **while True do**
- 4: Update the factor graphs F_{exe} , F_{in} , F_{out} , and F_{port} with the respective conditional probabilities P_{Attack} and P_{Benign} .
- 5: Calculate $\mu_{exe \rightarrow Attack}$, $\mu_{in \rightarrow Attack}$, $\mu_{out \rightarrow Attack}$, and $\mu_{port \rightarrow Attack}$
- 6: For *unknown_exec* and *in_connect*, calculate F_{Attack} using Eq. 14.
- 7: Calculate BEL_{Attack} using Eq. 15.
- 8: **if** $BEL_{Attack} < lower_belief$ **then**
- 9: Alarm "attack presence".
- 10: Update the tables in *Cassandra* DB with newly-identified attack features.
- 11: **End do**
- 12: **End**

Overall algorithm of BDSA:

Our BDSA approach can be designated in pseudo codes as seemed in Algorithm 2. The fashionable information streams of our BDSA technique can be shown as in Figure five. The execution of the proposed BDSA technique starts by using stacking all notable noxious and additionally kindhearted port numbers from the dispersed Cassandra database. Both of those port sorts are then used to prepare a classifier making use of strategic relapse. This lets in the proposed way to cope with determine on-the-fly the likelihood of an obscure port being malignant, before passing it to the conviction unfold structure for conclusive collection. A prepared strategic classifier is applied to determine whether or not any of the traits are malignant or considerate, before passing their person possibilities to the conviction spread process for precise probability conglomeration. Conviction unfold procedure takes attack's contingent possibilities regarding singular ascribes to check the conviction of assault nearness, thinking about every restrictive likelihood esteems to guarantee that the esteem received isn't impacted simply by way of any restrictive probability alone.

Algorithm 2 Security analytics in BDSA

```
1: Initialize: Obtain benign and malicious parameters of
   the attack features from Cassandra DB.
2: Train classifiers for monitored features using Logistic
   Regression.
3: while True do
4:   Collect network and user application logs from guest
   VMs.
5:   Filter network log entries using the guest VMs' IP
   addresses.
6:   Form correlated_log.
7:   Use correlated_log to form a correlation graph G.
8:   Input G into MapReduce parser to identify potential
   attack paths  $\{attack\_paths\}$ , which is a sub-set of all
   graph paths as shown in Figure 3.
9:   for each attack_path in  $\{attack\_paths\}$  do
10:     $i \leftarrow 0$ .
11:    for each monitored feature  $t_{feature}$  in attack_path
    do
12:      Calculate  $P_{port\_change}$ ,  $P_{unknown\_exact}$ ,  $P_{in\_connect}$ ,
    and  $P_{out\_connect}$ 
13:      Pass  $P_{port\_change}$ ,  $P_{unknown\_exact}$ ,  $P_{in\_connect}$ , and
     $P_{out\_connect}$  into Step. 4 of Algorithm 1.
14:    End do
15:  End do
16: End
```

K-means clustering algorithm:

K-implies is a approach for grouping perceptions into a selected quantity of disjoint bunches. The "K" alludes to the quantity of organizations decided. Different separation measures exist to determine out which notion is to be connected to which group. The calculation goes for proscribing the measure among the centroe of the group and the given belief via iteratively adding a perception to any bunch and stop whilst the most decreased separation measure is performed.

Outline of Algorithm:

1. The example area is at the beginning divided into K bunches and the perceptions are arbitrarily allocated to the businesses.
2. For every example:
 - Calculate the separation from the belief to the centroide of the organization.
 - IF the instance is nearest to its own institution THEN abandon it ELSE chooses every other bunch.
3. Rehash tiers 1 and a pair of until the point whilst no perceptions are moved starting with one institution then onto the following while degree 3 ends the bunches are constant and each example is doled out a group which ends in the maximum decreased potential separation to the centriode of the institution.

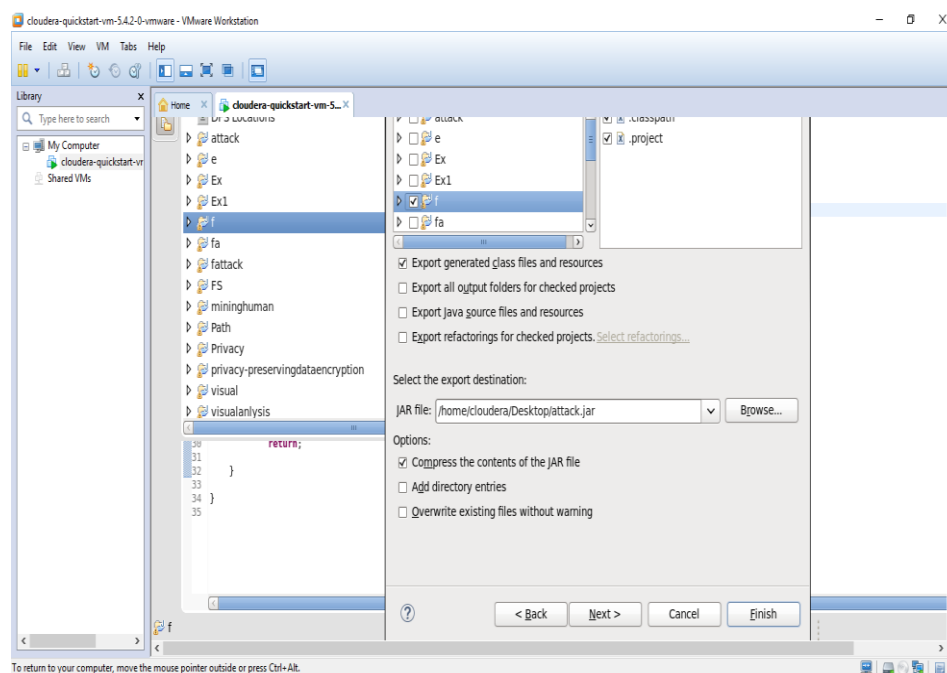
Distance measures:-

Basic separation measures contain the Euclidean separation, the Euclidean squared separation and the Manhattan or City get rid of. The Euclidean degree compares to the briefest geometric separation between focuses.

$$d = \sqrt{\sum_{i=1}^N (x_i - y_i)^2}$$

III. RESULTS AND DISCUSSION

In the below screen we will create a jar file.

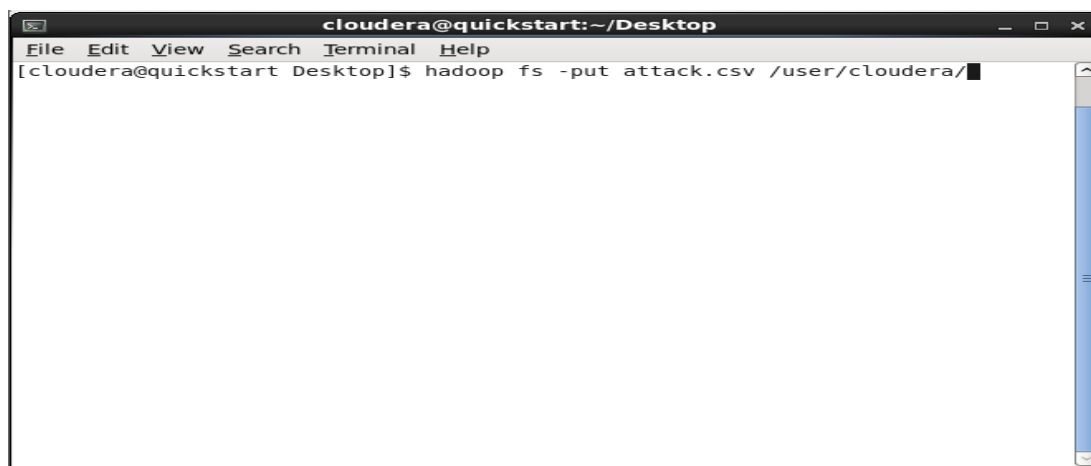


Initially we have different path on the screen now we will change this to desktop path in the below screen



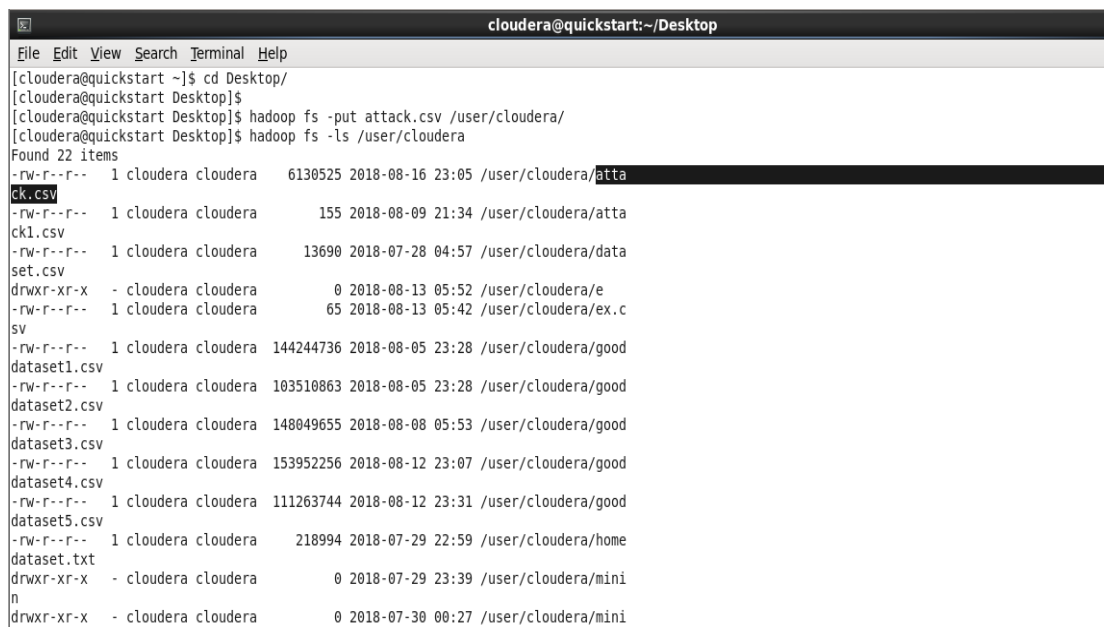
```
cloudera@quickstart:~  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ cd Desktop/
```

In this below screen we will take dataset from desktop and store it in the HDFS storage Location



```
cloudera@quickstart:~/Desktop  
File Edit View Search Terminal Help  
[cloudera@quickstart Desktop]$ hadoop fs -put attack.csv /user/cloudera/
```

In this screen we will check whether the dataset is present in HDFS storage or not.



```
cloudera@quickstart:~/Desktop  
File Edit View Search Terminal Help  
[cloudera@quickstart ~]$ cd Desktop/  
[cloudera@quickstart Desktop]$  
[cloudera@quickstart Desktop]$ hadoop fs -put attack.csv /user/cloudera/  
[cloudera@quickstart Desktop]$ hadoop fs -ls /user/cloudera/  
Found 22 items  
-rw-r--r-- 1 cloudera cloudera 6130525 2018-08-16 23:05 /user/cloudera/atta  
ck.csv  
-rw-r--r-- 1 cloudera cloudera 155 2018-08-09 21:34 /user/cloudera/atta  
ck1.csv  
-rw-r--r-- 1 cloudera cloudera 13690 2018-07-28 04:57 /user/cloudera/data  
set.csv  
drwxr-xr-x - cloudera cloudera 0 2018-08-13 05:52 /user/cloudera/e  
-rw-r--r-- 1 cloudera cloudera 65 2018-08-13 05:42 /user/cloudera/ex.c  
sv  
-rw-r--r-- 1 cloudera cloudera 144244736 2018-08-05 23:28 /user/cloudera/good  
dataset1.csv  
-rw-r--r-- 1 cloudera cloudera 103510863 2018-08-05 23:28 /user/cloudera/good  
dataset2.csv  
-rw-r--r-- 1 cloudera cloudera 148049655 2018-08-08 05:53 /user/cloudera/good  
dataset3.csv  
-rw-r--r-- 1 cloudera cloudera 153952256 2018-08-12 23:07 /user/cloudera/good  
dataset4.csv  
-rw-r--r-- 1 cloudera cloudera 111263744 2018-08-12 23:31 /user/cloudera/good  
dataset5.csv  
-rw-r--r-- 1 cloudera cloudera 218994 2018-07-29 22:59 /user/cloudera/home  
dataset.txt  
drwxr-xr-x - cloudera cloudera 0 2018-07-29 23:39 /user/cloudera/mini  
n  
drwxr-xr-x - cloudera cloudera 0 2018-07-30 00:27 /user/cloudera/mini
```


In this screen we will run the Jar file:

```
cloudera@quickstart:~/Desktop
File Edit View Search Terminal Help
[cloudera@quickstart Desktop]$ hadoop jar attack.jar f.df /user/cloudera/attack.csv /user/cloudera/attack
```

```
cloudera@quickstart:~/Desktop
File Edit View Search Terminal Help
18/08/16 23:08:12 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
18/08/16 23:08:13 WARN mapreduce.JobSubmitter: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your
th ToolRunner to remedy this.
18/08/16 23:08:14 INFO input.FileInputFormat: Total input paths to process : 1
18/08/16 23:08:14 INFO mapreduce.JobSubmitter: number of splits:1
18/08/16 23:08:14 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1534480349920_0004
18/08/16 23:08:14 INFO impl.YarnClientImpl: Submitted application application_1534480349920_0004
18/08/16 23:08:15 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1534480349920_0004/
18/08/16 23:08:15 INFO mapreduce.Job: Running job: job_1534480349920_0004
18/08/16 23:08:26 INFO mapreduce.Job: Job job_1534480349920_0004 running in uber mode : false
18/08/16 23:08:26 INFO mapreduce.Job: map 0% reduce 0%
18/08/16 23:08:35 INFO mapreduce.Job: map 100% reduce 0%
18/08/16 23:08:44 INFO mapreduce.Job: map 100% reduce 100%
18/08/16 23:08:45 INFO mapreduce.Job: Job job_1534480349920_0004 completed successfully
18/08/16 23:08:46 INFO mapreduce.Job: Counters: 49
  File System Counters
    FILE: Number of bytes read=551
    FILE: Number of bytes written=221323
    FILE: Number of read operations=0
    FILE: Number of large read operations=0
    FILE: Number of write operations=0
    HDFS: Number of bytes read=6130646
    HDFS: Number of bytes written=529
    HDFS: Number of read operations=6
    HDFS: Number of large read operations=0
    HDFS: Number of write operations=2
  Job Counters
    Launched map tasks=1
    Launched reduce tasks=1
```


In the below screen we view the List of Output Files:

```
cloudera@quickstart:~/Desktop
File Edit View Search Terminal Help
[cloudera@quickstart Desktop]$ hadoop fs -ls /user/cloudera/attack
Found 2 items
-rw-r--r-- 1 cloudera cloudera          0 2018-08-16 23:08 /user/cloudera/attack/_SUCCESS
-rw-r--r-- 1 cloudera cloudera      529 2018-08-16 23:08 /user/cloudera/attack/part-r-00000
[cloudera@quickstart Desktop]$
```

In this screen we will see the final output and these are in the form of clusters

```
cloudera@quickstart:~/Desktop
File Edit View Search Terminal Help
[cloudera@quickstart Desktop]$ hadoop fs -cat /user/cloudera/attack/part-r-00000
^vant 125.256.4.9, ,city.demon.co.uk,5938,Avant,1000
^vant 192.168.100.11, ,wwwproxy.sanders.com,5626,Avant,1000
^vant 173.194.45.47, ,piweba3y.prodigy.com,8341,Avant,1000
Internet Explorer 192.168.100.11,82,193.76.222.17,1938,Internet Explorer,1000
Internet Explorer 125.256.4.9,82,204.117.201.58,4262,Internet Explorer,1000
^axthon 173.194.45.47,8282,204.117.201.58,2941,Maxthon,1000
netscape Browser 173.194.45.47, ,ftp.rogers.com,9599,Netscape Browser,1000
JC Browser 56.251.76.112,82,204.117.201.58,4754,UC Browser,1000
[cloudera@quickstart Desktop]$
```

IV. CONCLUSION

In this paper, we have superior a unique large records based totally safety research (BDSA) way to cope with making sure virtualized frameworks in allotted computing towards slicing edge attacks. Our BDSA technique constitutes a 3 level structure for distinguishing propelled attacks gradually. In the first region, the visitor VMs' system logs and additionally patron utility logs are intermittently collected from the traveler VMs and placed away within the HDFS. At that factor, attack highlights are separated via relationship diagram and MapReduce parser. At last, -develop system studying is used to discover assault nearness and bunching is completed through utilizing the k-implies grouping calculation. Strategic relapse is hooked up to determine attack's contingent possibilities regarding singular developments. Besides, conviction proliferation is connected to compute the general conviction of attack nearness. From the second one degree to the 0.33, the extraction of attack highlights is additionally reinforced closer to the warranty of attack nearness by the two-boost device mastering.

REFERENCES

- [1] A. Szalay and J. Gray, "2020 Computing: Science in an exponential world," *Nature*, vol. 440, pp. 413–414, Mar. 2006.
- [2] E. U. Directive, "95/46/EC of the European Parliament and of the Council of 24 October 1995 on the Protection of Individuals with Regard to the Processing of Personal Data and on the Free Movement of such Data," *Official Journal of the EC*, vol. 23, 1995.
- [3] U. States., "Health insurance portability and accountability act of 1996 [micro form]: conference report (to accompany h.r. 3103)." <http://nla.gov.au/nla.catv4117366>, 1996.
- [4] "Hypervisors, virtualization, and the cloud: Learn about hypervisors, system virtualization, and how it works in a cloud environment." Retrieved June 2015.
- [5] M. Portnoy, *Virtualization Essentials*. 1st ed., 2012. Alameda, CA, USA: SYBEX Inc.,
- [6] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," tech. rep., July 2009.
- [7] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina, L. Badger, and D. Leaf, *NIST Cloud Computing Reference Architecture: Recommendations of the National Institute of Standards and Technology (Special Publication 500-292)*. USA: CreateSpace Independent Publishing Platform, 2012.
- [8] R. Dua, A. Raja, and D. Kakadia, "Virtualization vs containerization to support paas," in *Cloud Engineering (IC2E), 2014 IEEE International Conference on*, pp. 610–614, March 2014.
- [9] S. Ghemawat, H. Gobioff and S.-T. Leung, "The Google File System", *SOSP*, 2003.
- [10] NIST Special Publication 500–291 version 2, *NIST Cloud Computing Standards Roadmap*, July 2013, Available at <http://www.nist.gov/itl/cloud/publications.cfm>.
- [11] C. Lynch, "Big data: How do your data grow?," *Nature*, vol. 455, pp. 28–29, Sept. 2008
- [12] B. Russell, "Realizing Linux Containers (LXC)." <http://www.slideshare.net/BodenRussell/linuxcontainers-next-gen-virtualization-for-cloud-atl-summit-ar4-3-copy>. Retrieved October 2015.
- [13] United Nations, "The Universal Declaration of Human Rights." <http://www.un.org/en/documents/udhr/index.shtml>, 1948. Retrieved August 2015.
- [14] A. Westin, *Privacy and Freedom*. New York Atheneum, 1967.
- [15] U. States., "Gramm-leach-Bliley act." <http://www.gpo.gov/fdsys/pkg/PLAW106publ102/pdf/PLAW-106publ102.pdf>, November 1999.
- [16] U. S. F. Law, "Right to financial <https://epic.org/privacy/rfpa/>, 1978. privacy act of 1978."
- [17] D. Bigo, G. Boulet, C. Bowden, S. Carrera, J. Jeandesboz, and A. Scherrer, "Fighting cyber crime and protecting privacy in the cloud." European Parliament, Policy Department C: Citizens' Rights and Constitutional Affairs, October 2012.
- [18] S. Stalla-Bourdillon, "Liability exemptions wanted! internet intermediaries' liability under uk law," *Journal of International Commercial Law and Technology*, vol. 7, no. 4, 2012.
- [19] N. Mimura Gonzalez, M. Torrez Rojas, M. Maciel da Silva, F. Redigolo, T. Melo de Brito Carvalho, C. Miers, M. Naslund, and A. Ahmed, "A framework for authentication and authorization credentials in cloud computing," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, pp. 509–516, July 2013.