*International Journal of Research in Advent Technology, Vol.7, No.4, April 2019*
*E-ISSN: 2321-9637*
*Available online at www.ijrat.org*

# Hadoop Based Network Traffic Analysis To Count Domain Hits From Large Internet Logs

Hasmukh B. Domadiya[1], Dr. Girish C. Bhimani[2]
[1]*Assistant Professor, National Computer College, Jamnagar.*
[2]*Head, Department of Statistics, Saurashtra Unviersity, Rajkot.*

**Abstract:** The Internet has millions of web resources each of which with unique URL – Uniform Resource Locator. Nowadays every website has a large number of web resources under one domain name. The recent era of computerization and digitalization is seeing a rapid increase in the number of users as well as number of web resources accessed by these users. It is indeed a necessity to analyze user activities by finding some information from their Internet logs. Conventional methods like sniffers, firewalls are suitable for small to medium scale networks due to limited processing capabilities. Hadoop is the present solution which can be used to analyze Internet logs generated by a large or very large scale networks. This piece of work is based on analysis network traffic by collecting URLs visited by users. Domain hits are counted based on Hadoop. The implementation has been done with Cloudera Apache Hadoop Ecosystem.

**KEYWORDS:** HADOOP, URL, DOMAIN, HIT COUNTER, CLOUDERA

## 1. INTRODUCTION

The Internet has became one of the most complex structured repository of resources(web pages, documents, videos, databases) with fastest increasing growth. Every second, a large number of web resources are added making it more and more complex. Every day a large number of users start using the Internet too. The challenging task is how to handle such rapid growth of the Internet to facilitate large number of web resources and large number of users. The Internet is based on unique identification for each of the web resources with its URL – Uniform Resource Locator concept. Every website has a unique domain which can be used to access it. URLs are corresponding addresses of the web resources for a website. Various authorities like ISPs, Network administrators keep logs of user activities by maintaining records of all the web resources accessed by various users. The easiest way is to list out accessed URLs timestamp wise. As the large number of users are accessing the Internet regularly, very large logs are maintained at various devices like log servers, firewalls etc. Analysis of the usage log helps in finding security threats, trends etc. This research paper focuses on introducing a novel approach to develop a domain hit counter which is computationally capable to process extremely large Internet logs [1].

Conventional storage and processing units have limitations while processing huge amount of data – specially in GBs. As discussed earlier, the increase usage of Internet generates a large Internet logs which is very time consuming to get processed by devices with conventional algorithms for analysis. The Internet itself is distributed and so its logs too. A system which can work in a distributed manner is also a good solution for fast and efficient processing. Hadoop is a technology which works efficiently in distributed environment as well as for large files. The Map-Reduce structured program makes it easy and efficient to do analysis. Cloudera's Apache Hadoop System is a set of most widely used open source components to change the way a large amount of data is stored, processed and analyzed. This research work develops a Hadoop based solution to determine domain popularity by counting number of visits per domain. The purpose here is to count number of hits per domain, which includes hits of all web resources corresponding to that domain [1].

Section 2 explains how the Internet is named today and how the conventional monitoring systems maintain logs of user activities. Section 3 explains how hadoop is superior and useful as compared to conventional methods when used on large scale basis. Section 4 explains the implementation with Cloudera Apache Hadoop Ecosystem. Section 5 concludes the paper with future directions.

## 2. INTERNET & URL

The Internet is based on TCP/IP model where every connected device gets and IP address. Every device is identified with its unique IP address and so the web servers too. Web server which are hosting web resources like websites are assigned public IP addresses. With the increasing number of websites, it is difficult to remember or search websites and corresponding web resources based on numerical IP addresses. To overcome this issue and to increase ease for the users, an alphanumerical identification system is used called URL - Uniform Resource Locator [1].

### *2.1 DNS - Domain Name System*

As the IP addresses are difficult to remember corresponding to various websites, DNS – Domain Name System was introduced for user convince while specifying the websites and corresponding web resources. DNS maps a user friendly name of a website with its IP address so underlying network can facilitate the communication. The structure of DNS based resource access is shown in Figure 1[1].

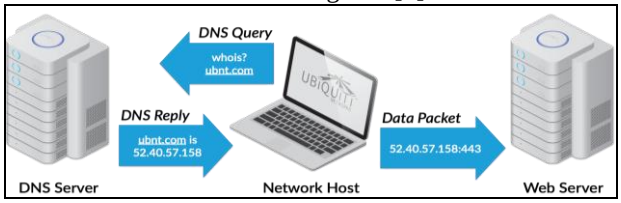


Figure 1 – DNS System

A user tries to access the Internet with an intention to visit a web resource (Most of the time it is a website) from its computer called Network Host. User enters user friendly address in the browser (Here ubnt.com). The Network Host needs to identify the corresponding IP address so that communication could be made. It sends a DNS query asking for the IP address to the DNS Server. DNS serve replies with the corresponding IP address of the server having specific web resource(Here 52.40.57.158). Subsequently, Network Host forwards data packets to the web server whose IP is provided by DNS Server. Further to the process, communication can be performed in a client server scenario[1].

### *2.2 URL – Uniform Resource Locator*

A domain name is a small piece of information which identifies a website at a higher level. Generally a website is made up of a large number of web pages, documents etc. each of these resources needs to be identifying uniquely so addresses can be specified in website with navigation facility. URL – Uniform Resource Locator is a system to identify each of the web resources uniquely on the Internet. Figure 2 shows the structure of URL[1].

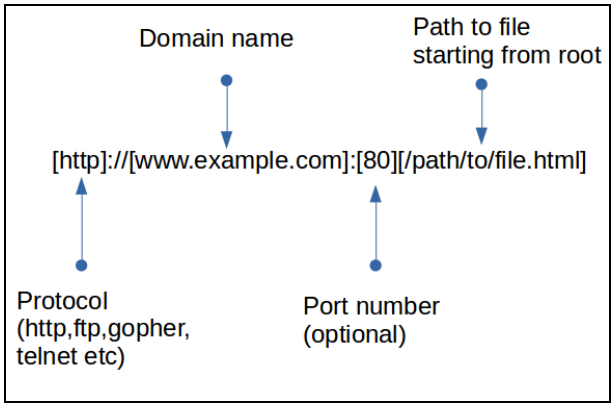


Figure 2 – URL Format

A URL starts with the protocol through which a web resource could be accessed. Domain name identifies the webserver as explained in section 2.1 and which mainly represents the website at a higher level. It is logically a user friendly representation of IP address. Port address is optional and it is used when a server hosts multiple websites at a time. Path to the file defines the exact web resource inside a specific website. Here virtual path is specified which will be translated into actual physical path by the webserver based on the user request. Followed by the path, query string may be used to send some other relevant information to the web server which can be used for dynamic content retrieval or setting some parameters. So we can conclude that a domain name is just a part of a URL [1].

### *2.3 Internet Log Structure*

The exact structure of Internet log depends on the device and its policies set by network administrator but following are the most common fields which are mostly logged by every organization by their log servers or firewalls.

TimeStamp
UserName
Source IP and Destination IP
URL
Data Size

### *2.4 Domain Hit Counter*

The domain hit counter is used to count number of web resources corresponding to a domain which are accessed in a specific period of time. The purpose here is to count domain wise resource access requested by various users. Here which resource was accessed is not important as we need to count based on domain not on URLs [1].

## 3. HADOOP

### *3.1 Why Hadoop?*

The limited processing and storage capabilities of centralized processing systems like log servers, analyzers, firewalls made analysis of extremely large volume of Internet logs very difficult. Hadoop is a set of open source softwares which can be used to process extremely large volume of data very efficiently. The base of Hadoop is doing processing in parallel and distributed way across a set of computers connected to a network. The network here is made up of commodity hardware based processing which refers to using large number of existing computers in parallel to achieve high computation. In recent era of big data analytics, Hadoop has proven to be one of the most successful framework[2].

### *3.2 Hadoop Components*

Hadoop's processing part is Map Reduce programming model while Hadoop's storage part is HDFS – Hadoop Distributed File System. The Map Reduce programming model processes the data in parallel and distributed manner on a Hadoop cluster. Figure 3 shows Hadoop architecture[3].

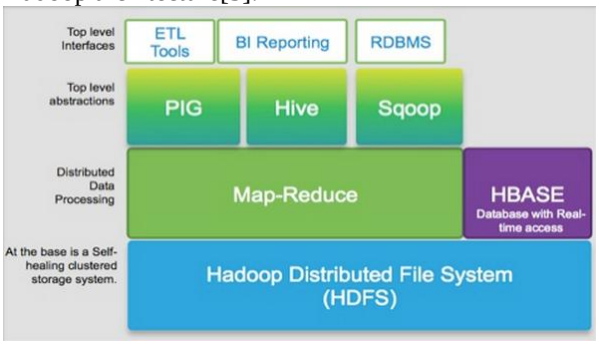


Figure 3 – Hadoop Architecture

In Figure 3 we can see that we need a distributed scalable file system which is HDFS. At next level, to achieve distributed and parallel data processing, Map-Reduce programming model is used. For database connectivity related applications, we can use HBASE. At top level to provide abstraction several graphical analyzers and query designers like Pig, Hive, Sqoop are available. The purpose is to provide easy and graphical interface to run Hadoop jobs. The top most interfaces are the system through which we will use data. ETL tools are the tools to extract, transform and load data

from one database to another database. BI – Business intelligence reporting tools helps in providing analytics into end user convenient format. Various other databases like RDBMS can also be used [4][5][6].
Map method is used for the purpose of filtration and sort operation of initial data. The reduce method is used for the purpose of summarize the target operation. The basic concept is to split data, apply rules and combine intermediate results to find final result. HDFS file system manages storage and retrieval of data in distributed and scalable way. It uses TCP/IP sockets for data communication where the RPC calls are used by clients machine to communicate each other. The main feature of HDFS is efficient storage of large files (in GBs, TBs) across multiple computers. The purpose is to let computers take benefits of locality of references while computing. Figure 4 shows a simplified form for a Hadoop cluster with HDFS architecture. Here HDFS manages the distribution of data across all the used data nodes in a distributed file system manner. It can be seen that the Name Node controls the process by distributing and collecting data among a set of data nodes[4][5][6]..

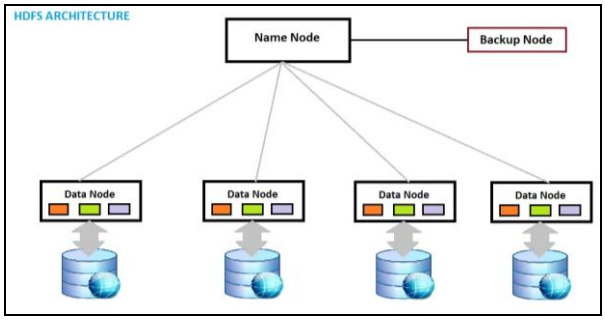

Figure 3 – HDFS Architecture

## 4. CLOUDERA APACHE HADOOP ECHOSYSTEM

### *4.1 Cloudera - CDH*

Cloudera provides Apache Hadoop based support for various softwares and services. It also organizes various trainings for customers. Cloudera has developed a hybrid open source solution which is based on Apache Hadoop. It is named as CDH - Cloudera distribution including apache Hadoop). It has various other softwares like Hive, Avro, HBase forming a combine Apache Hadoop platform. Apache Hadoop is based on mainly three components which are HDFS, Map Reduce and Yarn. Following are the features of these components. Figure 4 shows a cloudera based cluster design [8].

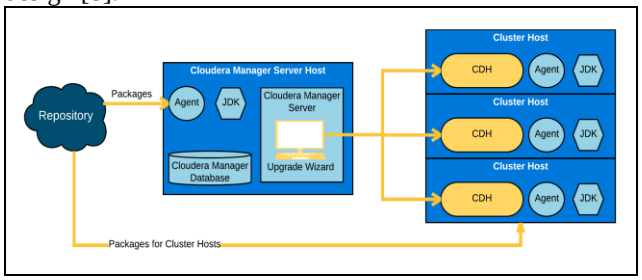

Figure 4 – Cloudera Cluster Design

### *4.2 Apache Hadoop Features*

Hadoop is scalable which allows large volume of data to store and process. At the same time, in future more data can be added by adding more computing units easily. It also allows flexible storage without any restrictions on type of data structure. It could be either structured/unstructured or semi structured. Hadoop is reliable as redundant copies make sure that no data is loss even if some of the systems are down. It has built in fault tolerance [8].
Map Reduce programming model supports many languages such as Java, Python, C++, Hive, Ping etc. It is also flexible as it supports processing of all types of data irrespective of its structure. The reliable nature and scalability with Hadoop and HDFS makes it easy to match the requirement of extensive processing [8].
YARN is basically deals with the Hadoop related resource management. It provides scalable resource management by allowing new workload on existing platform. Dynamic multi tenancy nature supports multiple engines on same cluster. It also supports, batch, real time, interactive processing. Optimal workload management takes care of effective utilization of resources. It supports priority based access support for cluster utilization [8].

## 5. SIMULATION

### *5.1 Test Data*

Any data analysis is inefficient if done with unreal data. We have used a database corresponding to an educational institute (name is not mentioned due to confidentiality reason). The data maintained by its firewall was taken. As Hadoop is able to process data which is very large in volume, we have used entire log of firewall which is of 336379 visited URLs in last one month. If we look into the number of distinct visited domains for these 336379 URLs then the count is 4949 domains. So logically we can think that users might have visited 336379 web resources corresponding to 4949 websites. Such a huge data is indeed very time consuming to get processed in traditional way. So we have used Hadoop based processing to count hits. The summary of data is given below.
Log Type: Log of Internet URLs
Institute: Educational institute.
Device: Firewall
Duration: 1 Month
Number of URLs: 336379
Number of Domains: 4949

### *5.2 Implementation*

In 1st case of the implementation, we have written a separate filter to extract hostname field values out of logs which are saved in a separate file. Later on this intermediate file is fed to the map reduce program for the purpose of analysis. This approach was found to be time consuming and manual work was required too. In 2nd case of the implementation, we have directly fed log file to the map reduce program. The map reduce program itself performs filtering to extract hostname columns for analysis purpose. This approach was more automated as well as less time consuming [7].

***5.3 Result***

Our model has extracted 4949 domains covering 336379 URLs. The highest hit count is 25253 and lowest hit count is 1. It is difficult to show case all the hit counts for 4949 domains so the top most 20 domains are shown in Figure 5. Map reduce required around 1.5 minutes to complete the entire job which is very less as compared to the conventional programming approach. The output is generated with tab separated sequence file which can be easily converted into CSV or can be used as a database input file.

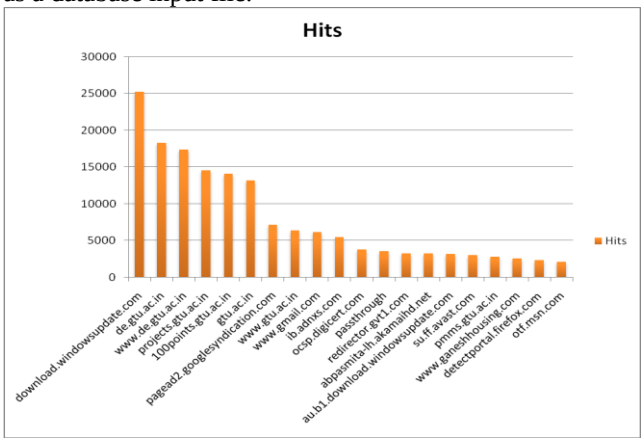


Figure 5 – Top 20 Domains Hit Count Wise

## 6. CONCLUSION

This work addressed two properties of the Internet which are responsible for its continuous increasing growth. 1st is the rapid increase of number of web resources and 2nd is the rapid increase of number of users and their Internet usage. The global addressing mechanism DNS and URL are discussed along with the basic concept of domain hit counter. It is obvious that for many reasons like security, accounting, analysis purpose, usage needs to be identified. Why centralized and conventionally programmed analyzers like log servers, analyzers, firewalls are not suitable is discussed along with the reasons. How Hadoop could be used to solve the problem with its parallel, distributed and decentralized processing concept is discussed. The introduction of Hadoop architecture, HDFS architecture and Cloudera Apache Hadoop Ecosystem is given. A Hadoop based solution to determine domain hit counter (irrespective of the respective complete URL hits) is proposed and tested for a large log maintained by firewall.

## 7. FUTURE WORK

As the Internet is growing day by day, it is indeed a requirement and necessity for any organization to analyze its inbound and outbound traffic for various reasons like security, accounting, and analysis. As the traditional centralized devices like firewalls, log analyzers, log servers have limited processing, storage capabilities; data analytics become difficult and limited in complexity. To achieve large log processing with complex analytical algorithms, advance methods such as Hadoop based solutions can be proposed. In addition to the solution of domain hit counter proposed in this piece of research, more work could be done to analyze the visited URLs. URLs can be analyzed to filter protocols, document types, date/time analysis, size etc. URLs could be analyzed to measure security threats to the organization. To find out who are the users have irregular or irresponsible Internet access pattern. To find out who are the users who are most trustworthy and who are not. a multimode cluster based approach could be used to combine the logs across multiple organization for the analysis purpose. Moreover the higher level tools such as Pig, Hive, Sqoop could be used for advance analytics with visualization support.

## REFERENCES

[1] Managing Internet and Intranet Technologies in Organizations: Challenges and Opportunities, Subhasish Dasgupta - July 2000

[2] Hadoop, http://hadoop.apache.org/.

[3] Shvachko, Konstantin, et al. "The hadoop distributed file system." Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on. IEEE, 2010.

[4] J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Cluster, OSDI, 2004.

[5]Lee, Yeonhee, and Youngseok Lee. "Toward scalable internet traffic measurement and analysis with hadoop." ACM SIGCOMM Computer Communication Review 43.1 (2013): 5-13.

[6]Lee, Youngseok, Wonchul Kang, and Hyeongu Son. "An internet traffic analysis method with mapreduce." Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP. IEEE, 2010.

[7] Big Data Analytics with R and Hadoop, Vignesh Prajapati-Packt Publishing Ltd - 2013

[8] Cloudera: Third Edition, Gerard Blokdyk, Create Space Independent Publishing Platform, 2017