International Journal of Communication and Information Technology

E-ISSN: 2707-6628
P-ISSN: 2707-661X
IJCIT 2020; 1(1): 40-43
Received: 12-11-2019
Accepted: 16-12-2019

Ediga Sravani
Department of Computer Science, Sri Venkateswara University, Tirupati, Andhra Pradesh, India

Corresponding Author:
Ediga Sravani
Department of Computer Science, Sri Venkateswara University, Tirupati, Andhra Pradesh, India

Internet traffic monitoring system using big data spark streaming

Ediga Sravani

DOI: https://doi.org/10.33545/2707661X.2020.v1.i1a.6

Abstract

Previous network analysis methods that usually work on a single machine are no longer suitable for huge traffic data owing to their poor processing ability. The Big data frameworks, such as Hadoop and Spark, can handle such analysis jobs even for a large amount of network traffic. However, Hadoop and Spark are inherently designed for offline data analysis. To cope with streaming data, various stream-processing-based frameworks have been proposed, such as Storm, Flink, and Spark Streaming. In our study, we proposed an online Internet traffic monitoring system based on Spark Streaming. The system comprises three parts, namely, the collector, messaging system, and stream processor. We considered the TCP performance monitoring as a special use case of showing how network monitoring can be performed with our proposed system. We conducted typical experiments with a cluster in standalone mode, which showed that our system performs well for large Internet traffic measurement and monitoring.

Keywords: Traffic, Frameworks, Streaming, Monitoring, Experiments, Standalone

1. Introduction

To give a protected and well-performing system for the constantly changing the internet, Internet administrators need to screen and investigate the system status continuously. Be that as it may, this is troublesome these days because of the colossal versatility of systems and the tremendous measure of traffic to be dissected. As indicated by the most recent measurements by Cisco [1], the yearly worldwide IP traffic was 1.2 zettabyte (ZB) in 2016 and expected to achieve 3.3 ZB by 2021. The quick development of traffic volume has forced incredible difficulties for conventional Internet observing stages. Customarily, Internet traffic estimation and examination have been executed on a superior focal server [2]. Be that as it may, because of registering capacity impediments, focal servers can't adapt to expansive volumes of information in a brief timeframe. These days, this renders them unsatisfactory given the enormous measure of Internet traffic today. For instance, when a DDoS assault happens, a checking framework is required so as to manage the gigantic measure of Internet traffic, which is an extreme undertaking for a solitary server. Different checking techniques use bundle examining so as to lessen the measure of info information. Notwithstanding, this creates an incorrect result [3]. Besides, a solitary server makes the framework defenseless against disappointments. On the off chance that the server smashed, we would not have the capacity to recoup it rapidly without influencing the progressing task [4].

2. Related Work

Cyberspace is dynamical and helpless against assaults. Along these lines, it requires arrange suppliers to screen the status of their system progressively. Online Internet traffic checking advances have been broadly considered. In 1999, Paxson [16] proposed the Bro framework to distinguish organize gatecrashers continuously. Brother originally caught a parcel stream utilizing libpcap and after that decreased the approaching stream into a progression of larger amount occasions utilizing an occasion motor. They additionally proposed a custom scripting language called Bro contents, which can be executed by the strategy content translator to manage occasions. In spite of the fact that Bro is single strung, it tends to be set up in a high throughput group condition. Comparable investigations incorporate Snort [17] and Suricata [18], which are inalienably founded on single-machine processing.

Different related examinations have been led on online Internet traffic estimation and checking utilizing Spark. Gupta *et al.* [12] utilized Spark Streaming to dissect the system

Progressively. They introduced three system checking applications that can be communicated as spilling investigation issues; in particular, reflection assault observing, application execution examination, and port output identification. They made utilization of programmable switches, for example, OpenFlow changes, to extricate just the traffic that was of intrigue, which diminished the information that should have been handled. Be that as it may, their framework was not doable for systems utilizing non programmable switches. In this investigation, we propose an Internet traffic estimation and observing framework that chips away at both programmable and non-programmable switches. Karimi *et al.* [13] proposed a dispersed system traffic highlight extraction technique with Spark for an ongoing interruption recognition framework. They utilized a gatherer part to catch parcels from the switch and concentrate the required data from bundle headers.

These headers are written in CSV documents and isolated when window. At that point, Spark occasionally peruses information from the CSV records inside a little time window to make it almost constant information. Notwithstanding, the occasional composition and perusing of documents corrupts the execution of Spark as an online Internet traffic observing framework. Our framework utilizes Spark Streaming to legitimately adapt to the stream so as to accomplish a higher speed.

3. Architecture

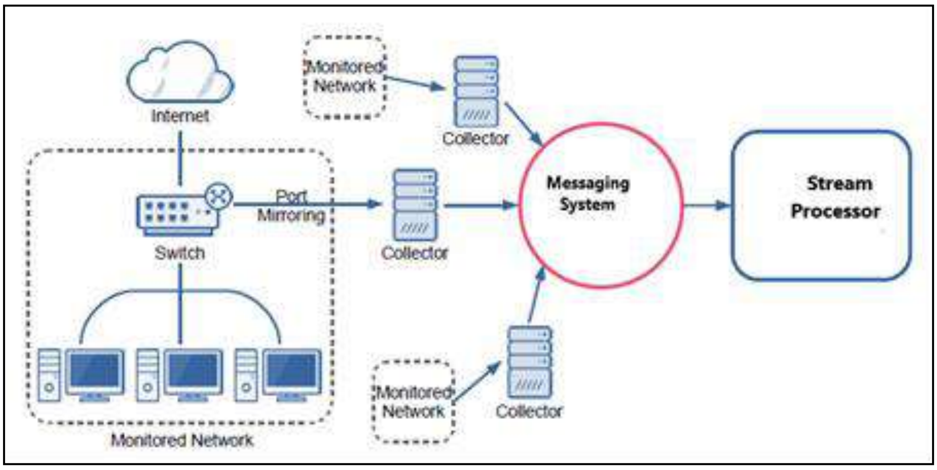


Fig 1: System Architecture

The proposed online monitoring system comprises three components, namely the collector, messaging system, and stream processor. A collector is a device that is used to capture packets from the network. It captures all the inbound and outbound packets from a switch using port mirroring. To capture packets from multiple switches, multiple collectors may be present in the system. The stream processor is the core component of our system and processes the input data transmitted from the collectors. First, the collector preprocesses the captured packets and only sends necessary protocol header data to the stream processor to reduce the amount of input data. Moreover, we use a messaging system as a bridge to help the data transmission from the collectors to the stream processor.

4. System design

4.1. Class Diagram

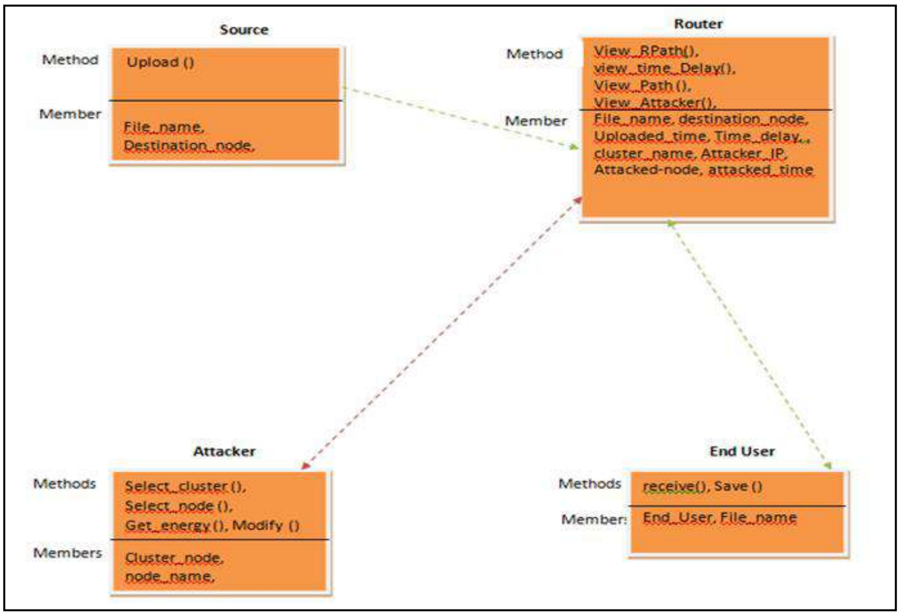


Fig 2: Class Diagram

4.2. Use Case Diagram

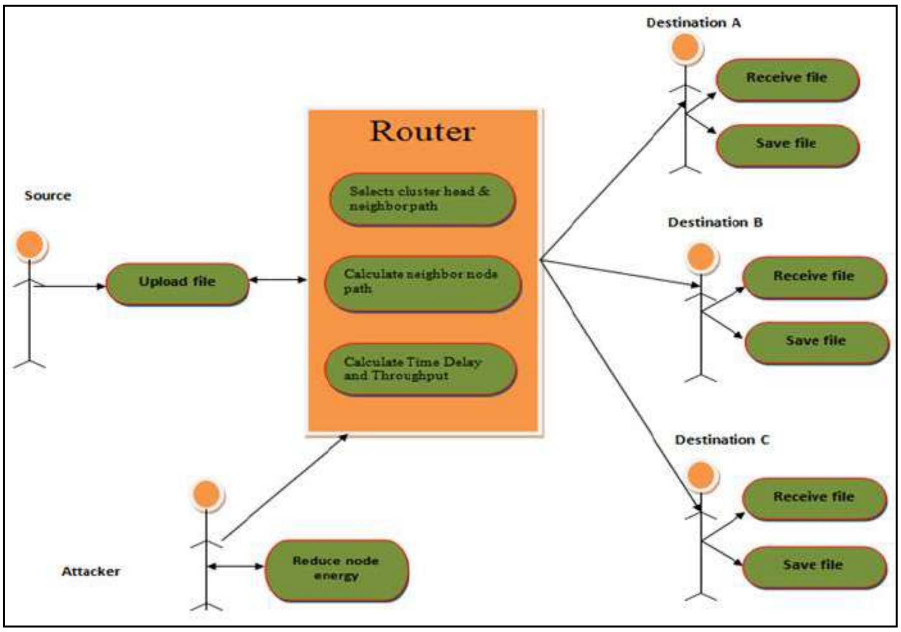
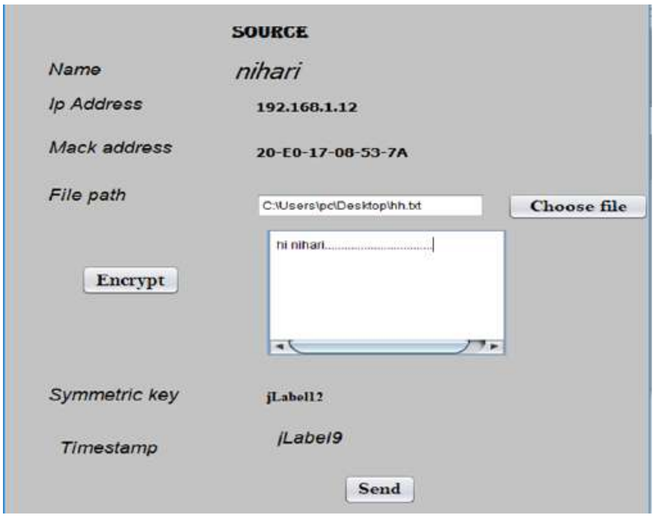


Fig 3: Class Diagram

5. Results and discussions



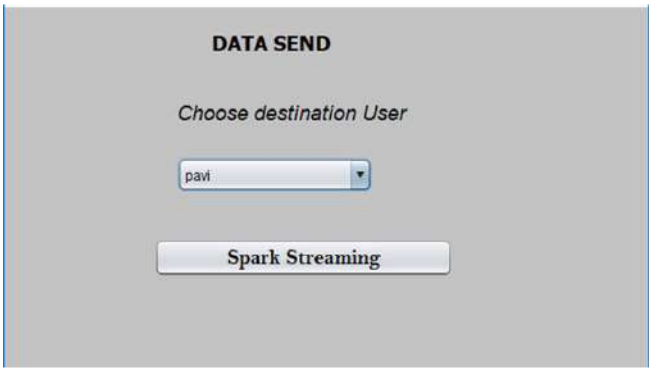
SOURCE

Name	nihari
Ip Address	192.168.1.12
Mack address	20-E0-17-08-53-7A
File path	C:\Users\pc\Desktop\hh.txt

hi nihari.......................................

Symmetric key	jLabel12
Timestamp	jLabel9

Fig 4: Source Page



DATA SEND

Choose destination User

pavi

Fig 5: Data Send

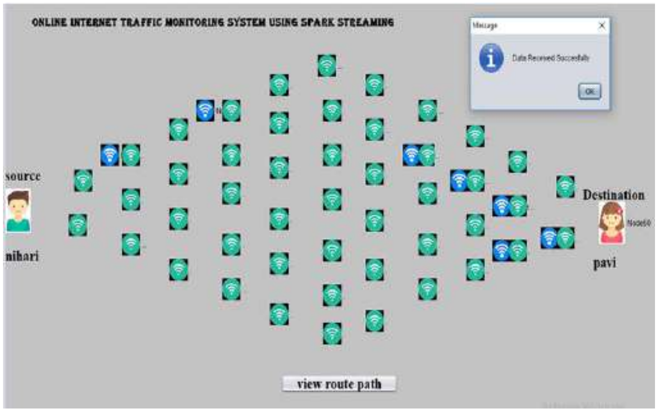
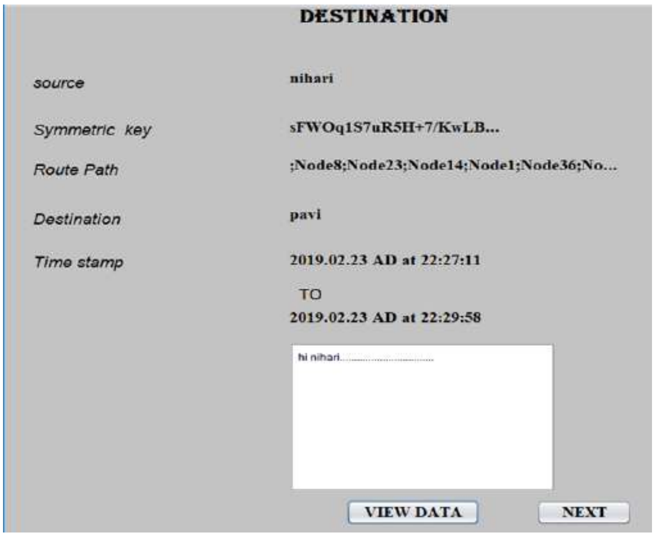

ONLINE INTERNET TRAFFIC MONITORING SYSTEM USING SPARK STREAMING
Message
Data Received Successfully
OK
source
nihari
Destination
Node60
pavi
view route path

Fig 6: Router Page



DESTINATION

source	nihari
Symmetric key	sFWOq1S7uR5H+7/KwLB...
Route Path	;Node8;Node23;Node14;Node1;Node36;No...
Destination	pavi
Time stamp	2019.02.23 AD at 22:27:11 TO 2019.02.23 AD at 22:29:58

hi nihari.......................................

Fig 7: Destination Page

6. Conclusion & future scope

With the growth of Internet traffic, traditional network analysis methods that work on single machines are no longer suitable. Existing approaches take advantage of big data frameworks to improve processing efficiency. However, these approaches mainly focus on offline data analysis. In this study, we proposed an online Internet traffic monitoring system that utilizes Spark Streaming. We demonstrated that Internet measurement and monitoring can be treated as a stream analysis problem and can be handled via a streaming processing platform. Extensive experimental results show that our system achieved good performance and robustness. In future, we will implement collectors to capture packets from switches through port mirroring so that our system can analyze all the traffics passing through monitored networks. Finally, we will test its performance in practice and compare it with some traditional single server systems in terms of scalability and reliability.

7. References

1. Cisco Visual Networking Index, Forecast and methodology, 2016-2021, White Paper, San Jose, CA, USA: Cisco, 2016.
2. Lee Y, Kang W, Son H. An Internet traffic analysis method with MapReduce, in Proc. 2010 IEEE/IFIP Network Operations and Management Symposium Workshops (NOMS Wksps), Osaka, Japan, 2010, 357-361.
3. Brauckhoff D, Tellenbach B, Wagner A, May M, Lakhina A. Impact of packet sampling on anomaly detection metrics, in Proc. 6th ACM SIGCOMM Conf. Int. Measurement, Rio de Janeriro, Brazil, 2006, 159-164.
4. Qiao YY, Lei ZM, Yuan L, Guo MJ. Offline traffic analysis system based on Hadoop, J China Univ. Posts Telecommun. 2013; 20(5)97-103.
5. Hadoop, http://hadoop.apache.org/, 2017
6. Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics, J Parallel Distrib. Comput. 2014; 74(7):2561-2573.
7. Apache Spark, http://spark.apache.org/, 2017.
8. Zaharia M, Chowdhury M, Franklin MJ, Shenker S, Stoica I. Spark: Cluster computing with working sets, in Proc. 2nd USENIX Conf. Hot Topics in Cloud Computing, Boston, MA, USA, 2010, 10.