International Journal of Security and Its Applications
Vol.8, No.2 (2014), pp.231-242
http://dx.doi.org/10.14257/ijsia.2014.8.2.24

A Small-time Scale Netflow-based Anomaly Traffic Detecting Method Using MapReduce

Wang Jin-Song, Zhang Long, Shi Kai and Zhang Hong-hao

School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin 300384
jswang@tjut.edu.cn

Abstract

Anomaly traffic detecting using Netflow data is one of important problems in the field of network security. In this paper, we proposed an approach using MapReduce model, which was realized by means of the entropy observation and DFN (Distinct feature number) distribution deviations of traffic features under anomalies at small time scales. The MapReduce was used to deal with huge amounts of data with the aid of computer cluster processing. Experimental results show the effectiveness of the proposed approach.

Keywords: *Netflow; MapReduce; small time scales; traffic features*

1. Introduction

In recent years, with the Internet rapid development in global and kinds of internet applications popularizing, the Internet has become an essential tool for carrying information in people's daily life [1]. The security of Network information system becomes more and more important too.

Netflow is a protocol of traffic statistics developed by Cisco [2]. The work principle of Netflow is as follows: with the use of the standard exchange model, Netflow can process the first IP data packet of the data flow and form Netflow buffer, and then the same data based on cache information transfer in the same data flow, no longer matched the strategies of access. At the same time, Netflow cache contains the statistics information of data flow afterward. In other word, flow is a unidirectional data packet which has the same source IP, destination IP, source port and destination port. According to different version, there are several forms of Netflow data collection. At present, the widely used Netflow versions are V5 and V8 [3].

Several traffic features (*e.g.*, flow size, ports and addresses) have been suggested as candidates for entropy based anomaly detection [4]. The goal of this paper is to provide a better understanding of the use of Netflow-based methods in anomaly detection and accelerate the efficiency of anomaly detection.

In this paper we use the existing equipment and some low cost hardwares to design a small-time scale Netflow-based anomaly traffic detecting method using MapReduce. Through analyzing Netflow data, it can discover attack and intrusion behavior in the network. We propose a ten-dimensional anomaly analysis index to detect anomaly traffic and found their stability at small-scale time. The MapReduce computing model helps us to accelerate detection efficiency.

We base on the Netflow form as follows:

210.*.*.12|210.*.*. 95|7|12|1434|135|6|4|40

ISSN: 1738-9976 IJSIA

The meaning of the fields are source IP, destination IP, flow in port, flow out port, source port, destination port, type of protocol, number of packets, number of bytes[5].

The remainder of this paper is organized as follows: we introduce the MapReduce computing model in Section 2. Then in Section 3 we analyze the characteristics of network anomalies, present our proposed method and explain the details of the procedure. In Section 4, the experiment set-up, results and their analysis are explained. Finally, we conclude our paper in Section 5.

2. Formatting your Paper

MapReduce initialized by Google is a programming model for expressing distributed computation on massive amount of data and an execution framework for large-scale data processing on clusters of commodity servers [6]. The underlying idea of MapReduce comes from the well-known principles in parallel and distributed processing [7]. Hadoop is an open source implementation of MapReduce [8] written in java which provides reliable, scalable and fault tolerance distributed computing. Hadoop environment set up involves a great number of parameters which are crucial to achieve best performance. It allows programmers to develop distributed applications without any distributed knowledge.

Key-value pairs form the basic data structure in MapReduce. Keys and values may be primitives such as integers, floating point values, strings and raw bytes or they may be arbitrary complex structures (lists, tuples, associative array, *etc.*). Programmers typically need to define their custom data types. The map function takes the input records and generates intermediate key and value pairs. The reduce function takes an intermediate keys and a set of values to form a smaller set of values. Typically just zero or one output value is produced by the reducer. In MapReduce, the programmer defines a mapper and reducer with the following signature:

$$\text{Map } (k1, v1) \rightarrow [(k2, v2)]$$

$$\text{Reduce } (k2, [v2]) \rightarrow [(k3, v3)]$$

[…] denotes the list [9]

MapReduce framework is responsible for automatically splitting the input, distributing each chunk to workers (mappers) on multiple machines, grouping and sorting all intermediate values associated with the intermediate key, passing these values to workers (reducers) on multiple resources, this is shown in Figure 1. Monitoring the execution of mappers and reducers as to re-execute them when failures are detected is done by the master. It is common for MapReduce jobs to have thousands of individual tasks that need to be assigned to nodes in the cluster. In large jobs, the total number of tasks may exceed the number of tasks that can be run on the cluster concurrently, making it necessary for the scheduler to maintain some sorts of task queues and to track the progress of running tasks so that waiting tasks can be assigned to nodes as they become available.

HDFS [10] is the subproject of Apache foundation which is under project Hadoop and used to construct a distributed file system with cheap PC hardwares. Compared to the other distributed file systems, HDFS has the advantage of high reliability and low cost. HDFS has the following mechanisms:

- High fault tolerance: If some nodes break down, HDFS can quickly detects faults and takes measures to restore the data for fault nodes.

- Support streaming data access: Data in HDFS needs to use flow method to access, and doesn't support the random access model.
- Support massive data: HDFS supports for large-file storage; a large amount of small files will result the poor system performance.
- Simple consistency model: HDFS data supports the access mode of write once and read many; if the files are created, they can't be modified.

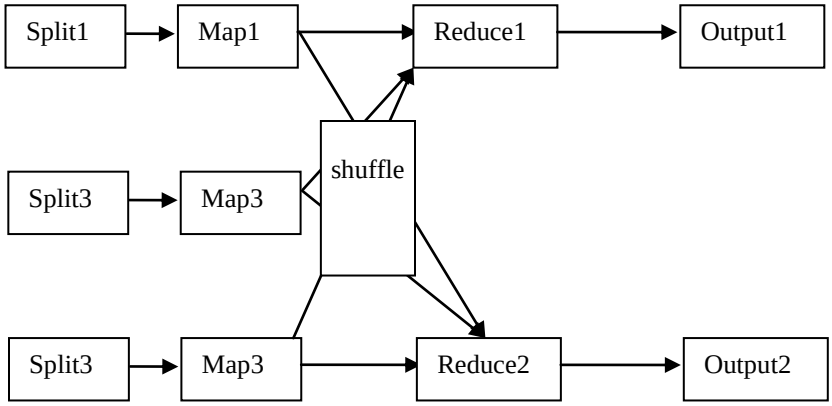

Figure 1. Simplified View of MapReduce

3. Implementation of the Anomaly Traffic Detecting Method

Generally, Netflow data reflects the real-time network performance. The network traffic features always change when anomaly traffic happens. We pick up some network traffic features from Netflow data and discover the law of the relationship of anomaly traffic and network traffic features.

3.1. The Network Traffic Features in Anomaly Traffic Detection

Entropy [11], reflecting the distribution probability of system microscopic, shows micro state diversity or uniformity in the rmodynamics. From the point of communication, the interference of randomness is inevitable. Therefore, communication system has the characteristics of statistics; information source can be seen as a set of random events. The randomness of this set is similar to the chaos degrees of micro state in thermodynamics, the information entropy will be formed when the thermodynamic probability are extended to the chance of all information source signal of system appear. The information entropy marks how much information contained, it is the description of the uncertainty of the system.

The entropy values of a sample of size n lie in the range [0, logn]. The minimum value 0 is taken when there is no variation in the data items (*e.g.*, single IP address or port) and the maximum value logn appears when all the data items are distinct or when the variation is large. In entropy-based detection techniques. The entropy of a random variable N with possible values {n1, n2, n3 …., nN} can be calculated as:

$$H(X) = -\sum_{i=1}^{N} (\frac{n_i}{S}) \log(\frac{n_i}{S}) \quad (1)$$

and $S = \sum_{i=1}^{N} n_i$.

Entropy is used to captain the degree of dispersal or concentration of the distributions for traffic features. The higher entropy indicates more dispersed distribution, whereas the distribution is more concentrated. At the same time we use N in the formula (1) as a new indicator as DFN (Distinct Feature Number). DFN also shows the degree of dispersal or concentration of the distributions for traffic features. We define the fixed value of consequent packets in Netflow data as a PU (packet unit). PU is the unit of the value of entropy and DFN. By analyzing the value of entropy and DFN in PU, anomaly traffic could be detected. Then we get a ten-dimensional anomaly analysis index system including the entropy and DFN of source/destination IP, source/destination port number and packet length as shown in Table 1.

Table 1. Ten-dimensional Anomaly Analysis Index

X(*)	Characteristics of flow	Entropy	DFN
X(SIP)	Source IP	H(X(SIP))	N(X(SIP))
X(DIP)	Destination IP	H(X(DIP))	N(X(DIP))
X(SPT)	Source port number	H(X(SPT))	N(X(SPT))
X(DPT)	Destination port number	H(X(DPT))	N(X(DPT))
X(PKT)	Packet length	H(X(PKT))	N(X(PKT))

3.2. Implementation using MapReduce

The MapReduce distributed data analysis framework model is good at large-scale data parallel computing. We get all the Netflow data from Netflow collector of the detected network. The Netflow data is so large that the traditional detection method always takes a sample. MapReduce help us to use all the Netflow data in detection of anomaly traffic. The initial Netflow data files are divided into several new files which are classified by network traffic characteristics, entropy and DFN of source/destination IP, source/destination port number and packet length. Upload all the processed files to HDFS. Setting the map function and the reduce function under which the files are analyzed. Through studying the analysis results we can find the relationship between the network traffic features and anomaly traffic. The specific process is as follows:

Step1:Data Collecting: Collect Netflow data in Cisco router.

Step2:Data processing: Separate the files into new PU files. The new files are classified by network traffic features, entropy and DFN of source/destination IP, source/destination port number and packet length.

Step3:Uploading files: Upload all the processed files to HDFS, using MapReduce to process all the files.

Step4:Analyzing results to find the relationships of the network traffic characteristics and anomaly traffic.

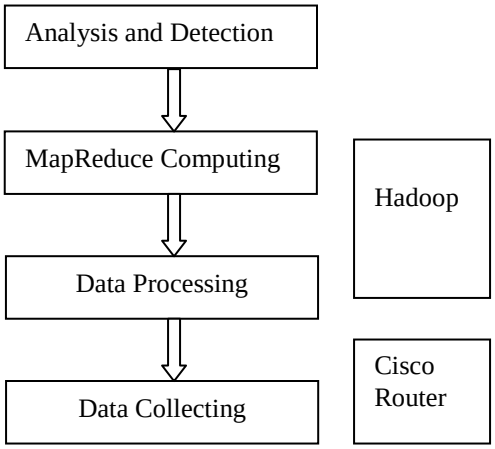

Figure 2. Implementation Procedure of the Method

4. Simulation

The experimental environment is shown in Figure 3. The data source in the simulation comes from Tianjin Urban Education Net. R1 is the border router, Cisco 7606; node A is the Netflow collector, it is a Netflow collector server with nfdump [12] which collected Netflow data and wrote into a file every one minute; Host B is the payload analysis server which is a Linux machine with snort which is a famous open source NIDS, used to monitor the network anomaly and compare to the Netflow detection results. C is the Hadoop cluster, used for storing and analyzing Netflow data. In this experiment the fixed value of consequent packets 6000, means that 1 PU contains 6000 consequent packets. One file always includes 450 PUs around.

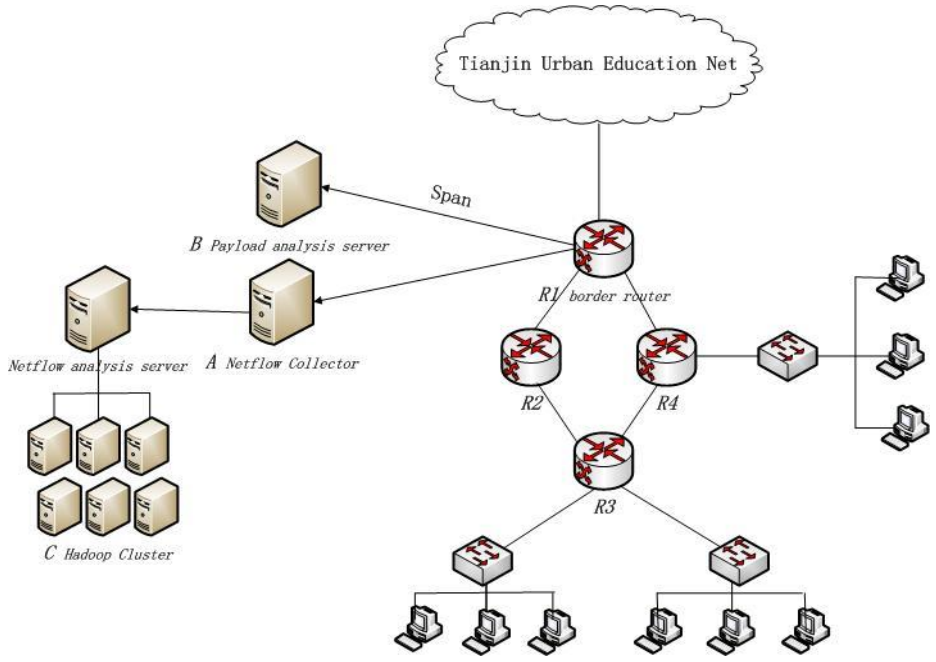

Figure 3. Experimental Environment

Case Ⅰ, The network is stable and reliable. Figure 4-Figure 8 show the entropy values of destination/source IP, destination/source port and packet length of 460 PUs, the time window is 20:00-20:01 in 21th May 2012.

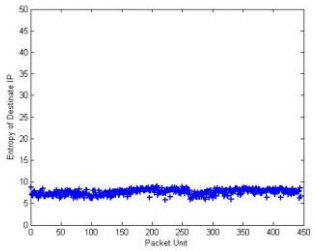

Figure 4. The Entropy of Destination IP in Normal Network

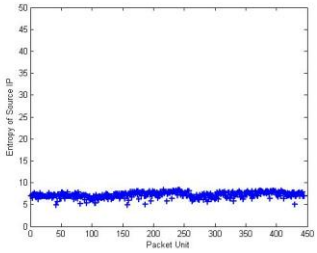

Figure 5. The Entropy of Source IP in Normal Network

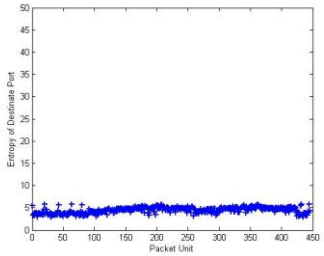

Figure 6. The Entropy of Destination Port Number in Normal Network

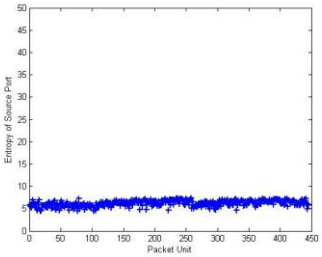

Figure 7. The Entropy of Source Port Number in Normal Network

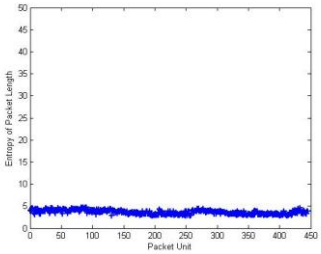

Figure 8. The Entropy of Packet Length in Normal Network

Figure 9-Figure13 show the DFN values of destination/source IP, destination/source port and packet length of 460 PUs, the time window is 20:00-20:01 in 21th May 2012.

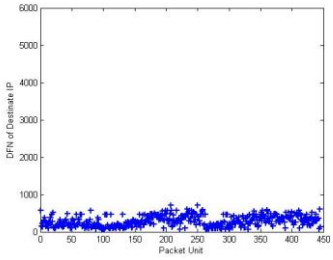

Figure 9. The DFN of Destination IP in Normal Network

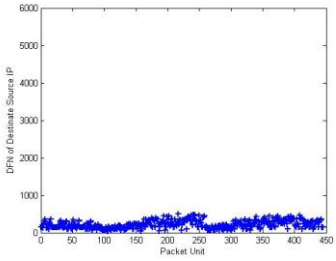

Figure 10. The DFN of Source IP in Normal Network

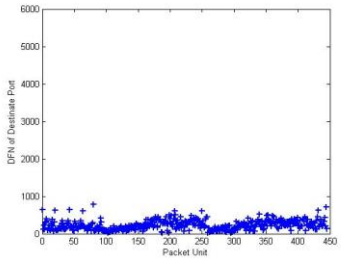

Figure 11. The DFN of Destination Port Number in Normal Network

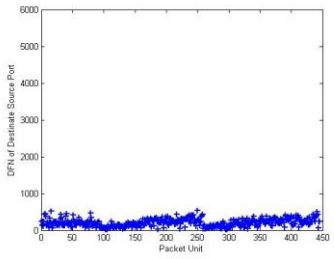

Figure 12. the DFN of Source Port Number in Normal Network

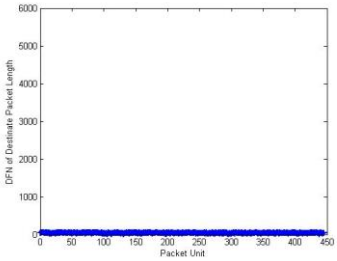

Figure 13. The DFN of Packet Length in Normal Network

From Figure 4-Figure 13, it is clear that in the stable and reliable network, the values of entropy and the DFN of the five traffic features are steady and they are in a small range.

Case Ⅱ, the network is under DDoS attack, and this is the situation where anomaly traffic happened. The Figure 14 shows the entropy value of destination IP, and the Figure 15 show the DFN value of destination IP.

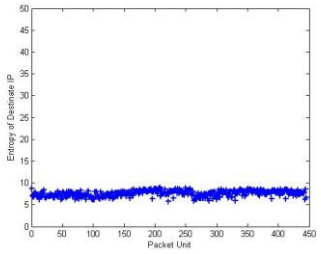

Figure 14. The Entropy of Destination IP under DDoS Attack

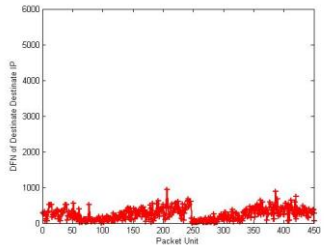

Figure 15. The DFN of Destination IP under DDoS Attack

From the Figure 14-Figure 15 show that when DDoS attack happens, the entropy value of destination IP is steady and they are in a small range but the DFN value of destination IP becomes ruleless and beyond the threshold value. DFN value could help us find anomaly traffic that the entropy value couldn't clearly shows.

Case Ⅲ, Figure 16-Figure 20 show the DFN values of destination/source IP, destination/source port and packet length of 460 PUs, the time window is 20:05-20:06 in 25th May 2012 when we scan the network.

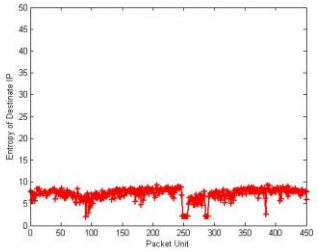

Figure 16. The Entropy of Destination IP in Abnormal Network

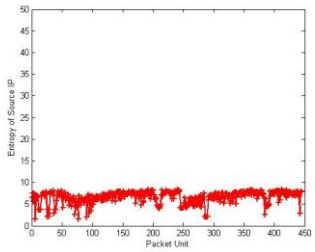

Figure 17. The Entropy of Source IP in Abnormal Network

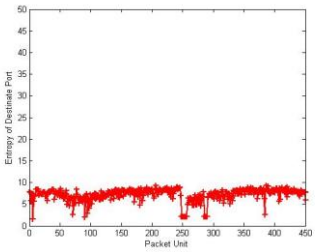

Figure 18. The Entropy of Destination Port Number in Abnormal Network

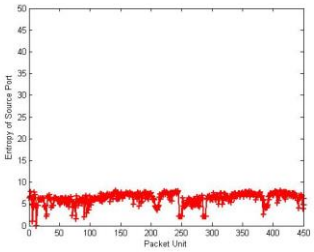

Figure 19. The Entropy of Source Port Number in Abnormal Network

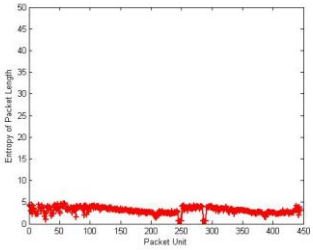

Figure 20. The Entropy of Packet Length in Abnormal Network

Figure 21-Figure 25 show the DFN values of destination/source IP, destination/source port and packet length of 460 PUs, the time window is 20:05-20:06 in 25th May 2012.

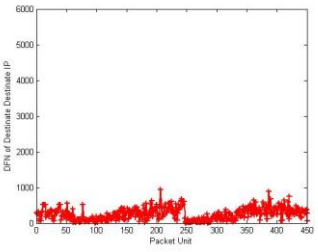

Figure 21. The DFN of Destination IP in Abnormal Network

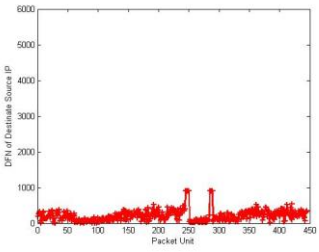

Figure 22. The DFN of Source IP in Abnormal Network

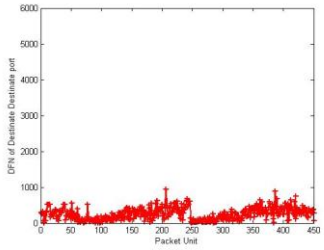

Figure 23. The DFN of Destination Port Number in Abnormal Network

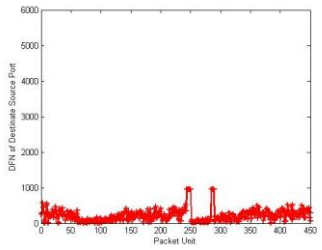

Figure 24. The DFN of Source Port Number in Abnormal Network

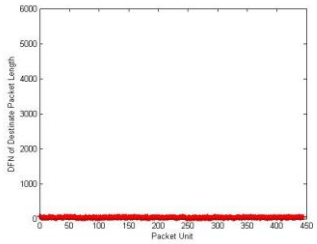

Figure 25. The DFN of Packet Length in Abnormal Network

Figure 16-Figure 25 depict that the results of entropy and DFN when anomaly traffic happed in network. They are show that the entropy and DFN become ruleless, and the values are beyond the threshold value. From long time experiment, we found that the entropy and DFN of the traffic futures are floating in a range of values. When the value is out of this range，the anomaly traffic happening in the network. The Table 2 shows the range values.

Table 2. The Range of Values of the Entropy and DFN of the Traffic Futures

Characteristics of flow	threshold value	Characteristics of flow	threshold value
H(X(SIP))	5.2-5.7	N(X(SIP))	100-700
H(X(DIP))	4.7-5.1	N(X(DIP))	100-800
H(X(SPT))	4.9-5.3	N(X(SPT))	50-500
H(X(DPT))	4.8-5.4	N(X(DPT))	50-500
H(X(PKT))	4.7-5.2		

According to the experimental results, we can arrive at the following reviews:

(1)The entropy and DFN of the network traffic characteristics are quite stable in normal network at small scale time.
(2)When anomaly traffic happened, the entropy and DFN of the network traffic characteristics has exceeded the threshold value.
(3)The DFN of the packet length has no significant change when anomaly traffic happened.

5. Conclusion

In this paper we present a Netflow-based anomaly traffic detecting method realized with the aid of MapReduce. A ten-dimensional anomaly analysis index system including the entropy and DFN of source/destination IP, source/destination port number and packet length is also proposed. With the use of MapReduce computing, the proposed approach improves the efficiency of anomaly traffic detection. The entropy and the DFN of traffic features are steady

in small-scale time. Experimental results show that the presented method is suitable to find anomaly traffic in network timely.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (61272450 and 61202381). The authors would like to thank Tianjin Key Lab of Intelligent Computing and Novel Software Technology and Key Laboratory of Computer Vision and System, Ministry of Education, for their support to the work.

References

[1] F. Bashir Shaikh and S. Haider, "Security Threats in Cloud Computing", 6th International conference on International Technology and Secured Transactions, **(2011)** December 11-14.
[2] Introduction to Cisco ISO netflow, Technical Overview, **(2007)**.
[3] Cisco System Ine. Introduction to Cisco NetFlow-A Technical Overview.
[4] Z. Jia, J. Yuehui and Y. Xiaowei, "Netflow based Anomaly Traffic Analyzer", Microcomputer application, vol. 28, no. 7, **(2007)** July.
[5] F. RasPall and A. Kock, "Implementation of a General-Purpose Network Measurement System", Budapest, Hungaray, **(2004)**.
[6] G. Yang, "The Application of MapReduce in the Cloud Computing", International Symposium on Intelligence Information Processing and Trusted Computing, **(2011)**, pp. 154-156.
[7] J. Dean, "Experiences with MapRedue, an abstraetion for large-scale computation", Proe.15th International Conferene on Parallel Arehitectures and Compilation Techniques, **(2006)**.
[8] S. Hammoud, M. Li, Y. Liu, N. K. Alham and Z. Liu, "MRSim: A discrete event based MapReduce simulator", Seventh International IEEE Conference on Fuzzy Systems and Knowledge Discovery (FSKD), **(2010)**.
[9] D. Peng and F. Dabek, "Large-scale Incremental Processing Using Distributed Transactions and Notifications", Operating Systems Design and Implementation, **(2010)** October.
[10] T. White, "Hadoop the Definitive Guide", **(2011)**.
[11] H. Xian-hua, Z. Yun, L. Ai-bing, H. Hong-rang, Y. Xiang-rong and Z. Jian, "Study of Entropy Flow Characteristics during the Evolution of Typhoon Morakot", 2011 International Conference on Electronics and Optoelectronics, **(2011)**.
[12] R. Hofstede, A. Sperotto, T. Fioreze and A. Pras, "The network data handling war: MySQL vs. NfDump", Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), LNCS, Networked Services and Applications - Engineering, Control and Management, vol. 6164, **(2010)**, pp. 167-176.

Authors

Wang Jin-song, is currently a teacher with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin 300384, China (e-mail: jswang@tjut.edu.cn).

Zhong Long, is currently a student with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin 300384, China, (e-mail:chh08290111@126.com).

Shi Kai is currently a teacher with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin 300384, China. (e-mail:shikai0229@tju.edu.cn).

Zhang Hong-hao is currently a teacher with the School of Tianjin University of Technology, Tianjin, 300384, China. His research interests include network security, trusted networks and next generation network(e-mail:zhanghonghao@tju.edu.cn).