IJIREEICE

ISSN (Online) 2321-2004
ISSN (Print) 2321-5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 8, Issue 11, November 2020

DOI 10.17148/IJIREEICE.2020.81104

A Distributed Deep Learning System for Web Attack Detection On Edge Devices

Ranjana Battur[1], Jyoti Amboji[2], Amruta Deshpande[3], Sangeeta Awati[4]

Assistant Professor, Department of Computer Science, Belgavi, Karnataka, India[1,2,3]

PG Student, Department of Computer Science, Belgavi, Karnataka, India[4]

Abstract: In this project first the data sets are read from the text file, after that file is processed and stored as datasets. First the data cleaning algorithm is executed with the help of stop words, special symbols and then clean data is obtained. After that the clean data into a set of words which will be represented as a wordized matrix. After that the word stream count of each of the words is found out. IDF and TF-IDF of each of the data set rows are obtained. Finally, the classification algorithm is executed and then class label is assigned to the event.

Keywords: IDF, TF-IDF, CNN

I.INTRODUCTION

The conventional ways to perform the various tasks have reduced with the help of internet technology. Web applications also have grown in a dramatic fashion and used by various fields across the globe. With the usage of cloud based systems it has become easy for maintenance of applications. Each of the web application on cloud or on a dedicated server will be identified by using unique URL. There is more probability of the system being under attack and then hacking of the server can take place. Among the different kind of attacks web server attacks are prominent with a 75% occupation. There are two different paradigms of gains from web attack with the first one being access to personal data for the end users and second one is injecting script which can hack the data when user clicks or performs some kind of downloads. Each web application by the various companies have to implement protocols to secure the system. However, the number of such attacks can be brought down by identifying and classifying them by reading the attributes of the request and then executing the machine learning algorithm.

II.LITERATURE SURVEY

The command-line execution [1] will be done by making use of a power shell and it is the application of windows that will run the data. The .Net based service will help in the execution of restrictions reducing the number of vulnerabilities, memory execution, logging of data. The attack chain will download the value of malicious data and then will generate the report which can be downloaded in the format of PDF file. The natural language processing (NLP) will detect the classification of data with the execution of a conventional neural network (CNN).

Deep neural network can be performed with the help of detection of malwares [2]. The value of detection rate will be done based on low false positive and volumes of hardware is done. the source code is taken from the customers and then detection of internal malware is done and then non-parametric method is used along with classifier scores is also done.

The network users make use of small hand held devices [3] in large numbers, hence possess more threat of malware attacks along with privacy concerns. The hidden apps are placed within the normal apps which are responsible for web attacks.

Web sites can be categorized into data holding and non-data holding. For data holding web site the web attacks can execute malicious scripts which can get the entities information for unauthorized and irrelevant users. The classification of the attacks is done using the process of SVM[4]. The dimensions of the tree system are increased in order to perform the classification.

IJIREEICE
ISSN (Online) 2321-2004
ISSN (Print) 2321-5526
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
Vol. 8, Issue 11, November 2020
DOI 10.17148/IJIREEICE.2020.81104

The process of finding the system points along with time taken for executing the functions is done with the help of monitoring the system along with detection of data with the help of system log [5]. The neural network along with memory computation will help in the process of assigning the class notion by making use of trained input and then predicting the right class notion. The Deep Log analysis is done by making use of specific data keys and then generating the work flow with the connected points in the log.

The execution of model for the user behavior [6] is done with the help of modeling the detection of behavior. Hidden Markova model will find the normal behavior and then perform the detection of threat using study of user behavior and also the deviations from the various behaviors are done with the help of novel method used for modeling of data [7].

III.EXISTING SYSTEM

In the existing system the request is tackled by making use of encoding techniques and word generation for each of the pages to avoid the Cross Site Scripting and other attacks. Each and every company has to write its own custom implementation for each of the attacks to handle. There is no plugin kind of a system which can take, analyze and pre-categorize the request.

IV.PROPOSED SYSTEM

In the proposed system, word vector along with neural network is used with the hidden layers along with TF-IDF methodology. The word vector block and TF-IDF are combined together in order to determine whether a given request is a web attack or it is not a web attack. There are different category of clusters which are considered in order to perform the category checks for the web attacks and the category includes the SQL, html. JavaScript and punctuations.

Methodology

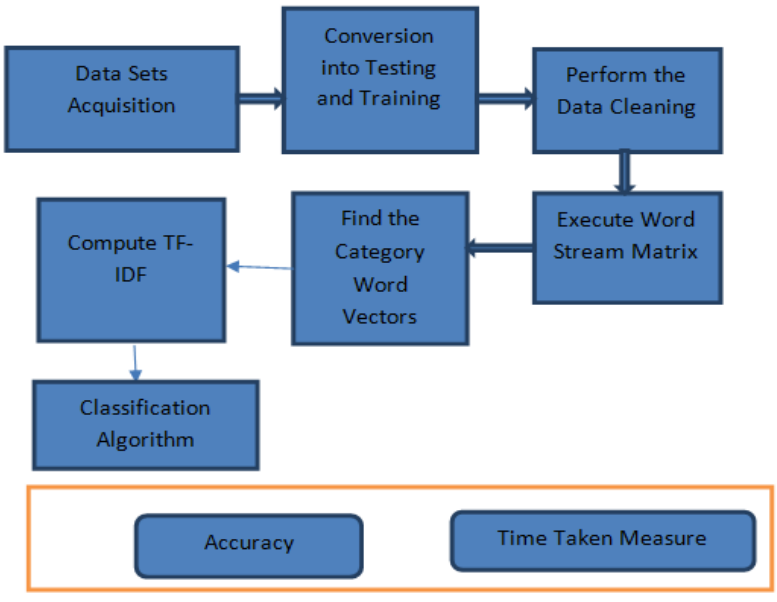

Fig1: Methodology

Data Sets Acquisition

The data in the format of CSV file is converted into a form of table by creating the table and then doing the import using Toad for MySQL database.

IJIREEICE
ISSN (Online) 2321-2004
ISSN (Print) 2321-5526
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
Vol. 8, Issue 11, November 2020
DOI 10.17148/IJIREEICE.2020.81104

Conversion for Data Sets into Training and Testing

Find the data sets into two one set will be used for training and other set will be used for prediction from the access logs

Perform the Data Cleaning

This module is responsible for removing of special characters (/ + ? & ; = , ‘ “ ( ) <> * ! $ # | ^ { } \ -- % ~ @ . ` [ ] :)and other stopwords from the HTTP headers and note not from the body

Execute Word Stream Matrix

The data cleaning statements are converted into a word format and each word is represented as an independent row in the format of word id, word and then data id. Each of request will be associated with a set of words

Find the Category Word Vectors

Each of the log section is compared against the set of attack detection words and then the feature based word stream count is computed for each of the categories. The various category related words are shown as below

Category Name Words

SQL select update delete insert create alter drop order by group by truncate replace commit rollback savepoint transaction set distinct all desc null limit top

HTML Different kind of html tags are taken into consideration

Java Script Different key words related to the java script like variable definition, function declarations, loop words and others

Compute Word Stream and Inverse Word Stream Count

The Word Stream and Inverse Word Stream will compute the Word Stream Count which is number of times the word is present in tokenization format and the Inverse Word Stream which will take into consideration number of data in which word is present and then divides it by word stream count finally applies logarithm measure.

Classification Algorithm

The classification algorithm will divide the data into a format of anomaly or non-anomaly with anomaly future divided into SQL label, HTML label and java script label.

Accuracy

This method is used to measure the number of correctly classified instances to the total number of instances

Time Taken

It is defined as the difference between the time at which classification is triggered to the time at which classification is completed.

V.DETAILED DESIGN

This chapter describes the various algorithms used in the project.

Unwanted Word Creation

IJIREEICE
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISSN (Online) 2321-2004
ISSN (Print) 2321-5526
Vol. 8, Issue 11, November 2020
DOI 10.17148/IJIREEICE.2020.81104

Unwanted Word Creation Module is responsible for creating the unwanted word which is not from the list of standard unwanted words. If the unwanted word exist then validation error is shown otherwise unwanted word is created.

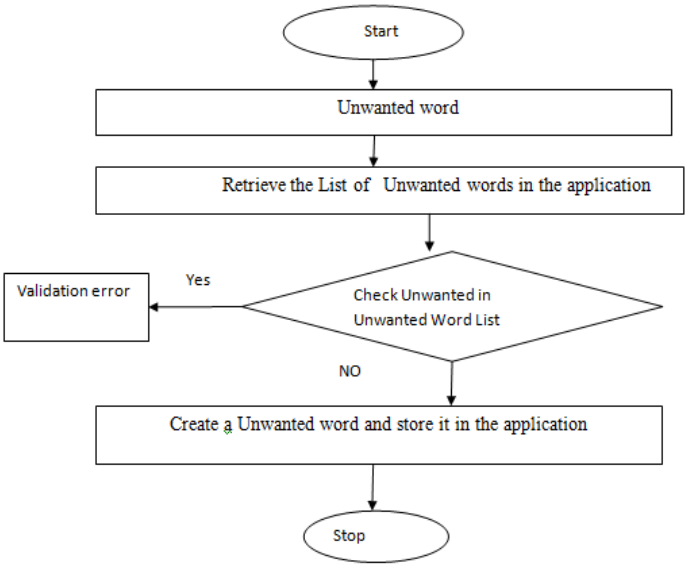

Fig: Un-Wanted Creation Module

Cleaned Data Sets

This task is responsible by making use of clean data. Each of the attributes are cleaned by removing the unwanted words and then other special symbols are removed.

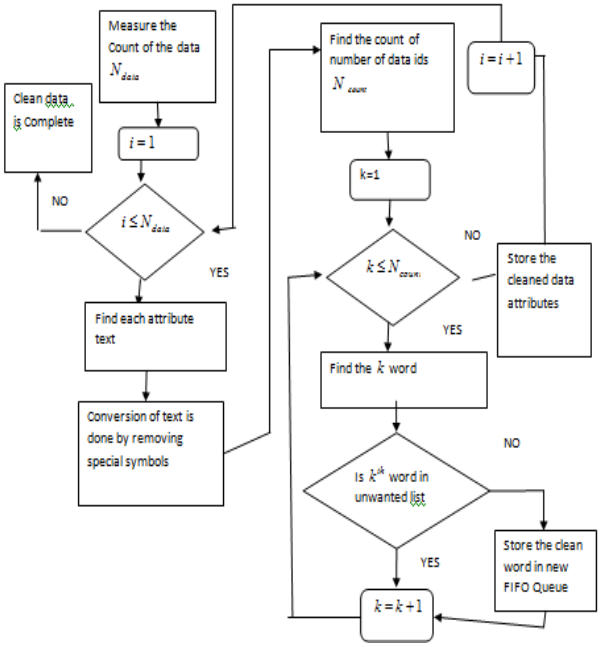

Fig: Clean Data Ids

Fig shows the clean data ids. As shown in the fig measure the count of number of data rows. After that from the 1st word till the number of words the process is repeated. For each attribute text the special symbols are removed. Find the count of number of data ids. From each word till the number of words of FIFO each word is taken and then checks whether

IJIREEICE
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISSN (Online) 2321-2004
ISSN (Print) 2321-5526
Vol. 8, Issue 11, November 2020
DOI 10.17148/IJIREEICE.2020.81104

word is present in unwanted list. If present, then skip the word otherwise the word is stored in the new FIFO queue and ten incremented. The process is repeated until the process is fully completed.

Word Stream Processing

The word stream processing is responsible for conversion of cleaned text into a set of word stream values.

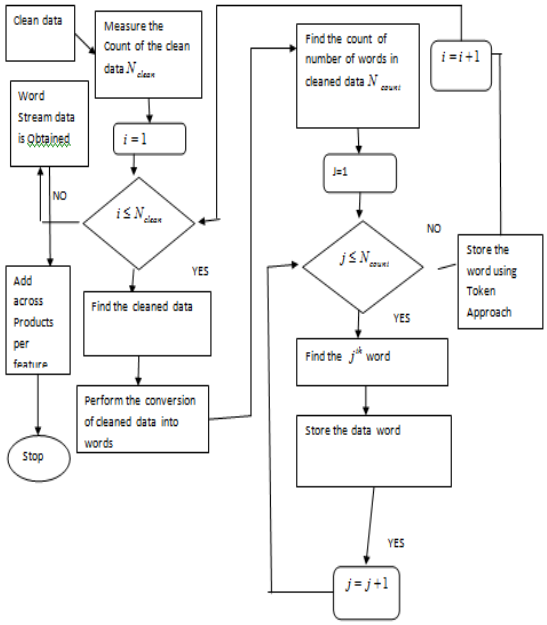

Fig: Word Stream Processing

Fig shows the word processing. As shown in the fig first cleaned text is obtained. After that the conversion is done into a FIFO queue using a delimiter. After that the count of number of words are obtained. From the 1st word till the last word each word is obtained and then storage is done of each word of the data with a specific representation of id.

Word Stream Count Processing

The word stream processing is responsible for conversion of cleaned text into a set of word stream values and then also stores the numeric value for representation of count.

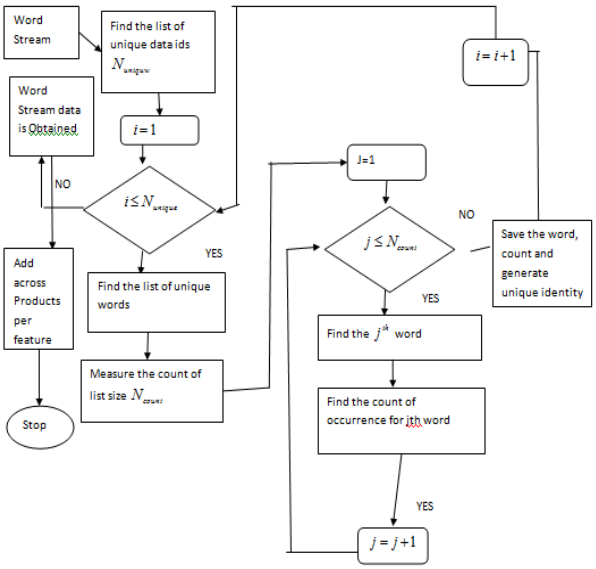

Fig: Word Stream Count Processing

IJIREEICE
International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering
ISSN (Online) 2321-2004
ISSN (Print) 2321-5526
Vol. 8, Issue 11, November 2020
DOI 10.17148/IJIREEICE.2020.81104

Fig shows the word stream processing. As shown in the fig first the list of unique data ids is obtained. After that count of unique ids is found. From the 1st word till the number of unique ids till the Number of unique ids are found. From the 1st data id list of unique words is found. The count of the unique words is found. Starting from 1st word till the number of unique words, the count of word repetition is done on the specific data id. The storage of the word stream count is done for each word.

Inverse Document Word and Weighted Count Processing

The inverse document word stream processing along with weighted measure of the word is done.

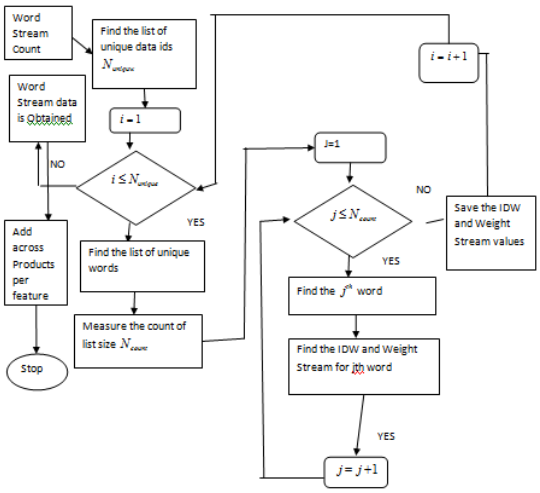

Fig: Inverse Word Stream and Weight Stream Processing

Fig shows the inverse word stream and word stream processing. As shown in the fig11 first the list of unique data ids is obtained from word stream count matrix. After that count of unique ids is found. From the 1st word till the number of unique ids till the Number of unique ids are found. From the 1st data id list of unique words is found. The count of the unique words is found. Starting from 1st word till the number of unique words, the number of different data ids in which word is present is found, after that Inverse Document Weight (IDW), then Weighted Measure is found and then process is repeated for all the Inverse Document Weight.

The IDW is computed using the following

$$IDW = \log\left(\frac{Ndataids}{C_{dataids}}\right)$$

Where,

$N = number\ of\ \ data\ ids\ \ in\ which\ \ word\ \ is\ \ present$

$f = count\ \ of\ \ word$

The Weighted Stream Word is computed using the following

$$WS = Count * IDW$$

Classification of Web Attack and Non Web Attack

1. Different words related to various kinds of web attacks are obtained

IJIREEICE

ISSN (Online) 2321-2004
ISSN (Print) 2321-5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 8, Issue 11, November 2020

DOI 10.17148/IJIREEICE.2020.81104

2. Find the unique data ids by making use of IDW and WS matrix
3. Start from the 1st data ids till the number of data ids
4. Obtain the count of various web attack category words
5. Find the maximum value of web attack words
6. The class label corresponding the maximum value is assigned a label
7.

VI.RESULTS

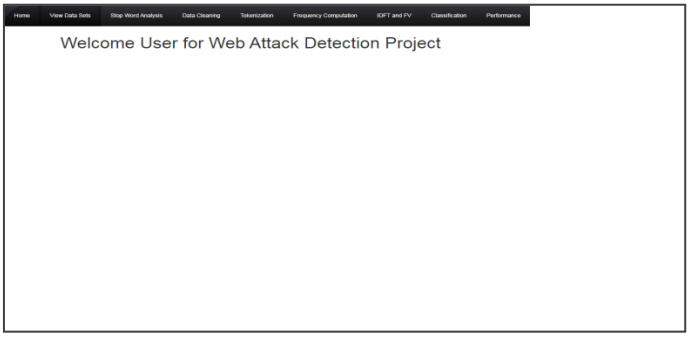

Fig: Welcome Page for Web Attack

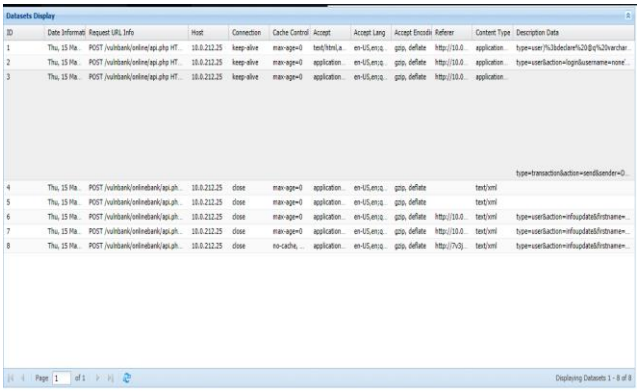

Fig: View dataset

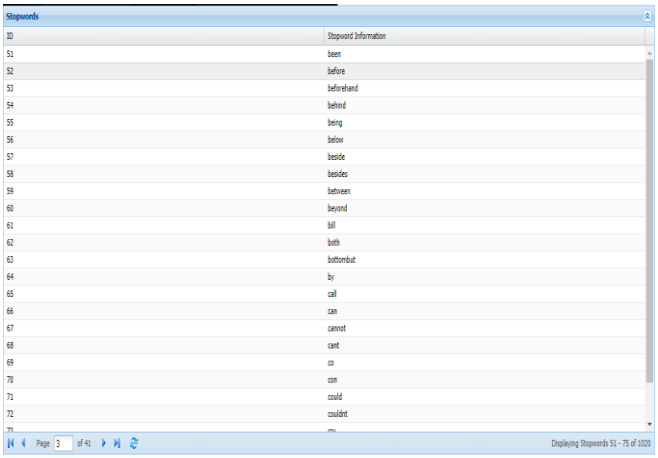

Fig: View Unwanted Words

IJIREEICE

ISSN (Online) 2321-2004
ISSN (Print) 2321-5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 8, Issue 11, November 2020

DOI 10.17148/IJIREEICE.2020.81104

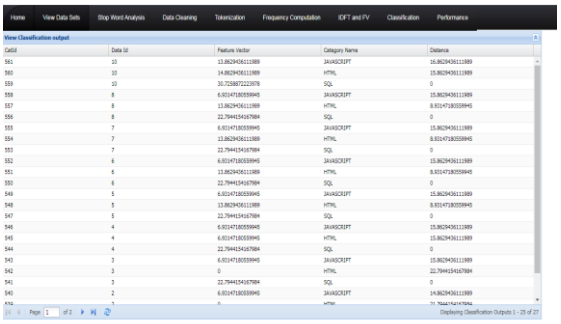

Fig: View Classification

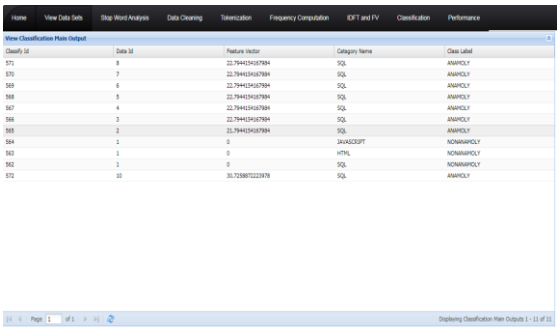

Fig: Classification Main Output

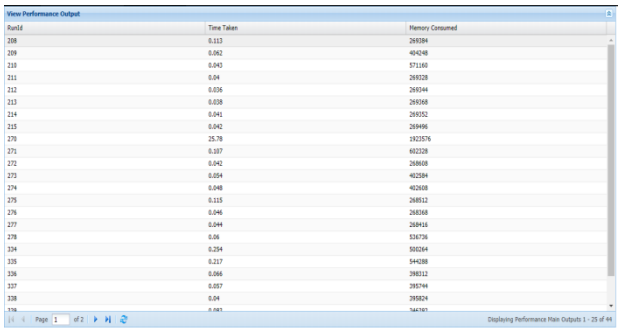

Fig: Classification Performance

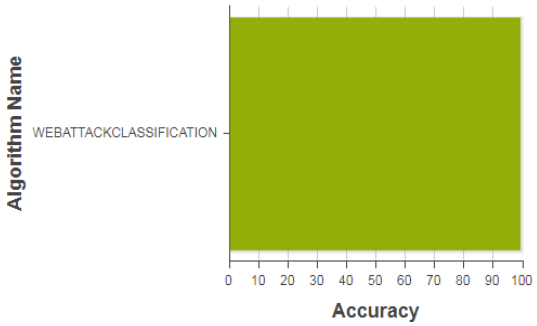

Fig: Accuracy comparison

IJIREEICE

ISSN (Online) 2321-2004
ISSN (Print) 2321-5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 8, Issue 11, November 2020

DOI 10.17148/IJIREEICE.2020.81104

VII.FINAL COMMENTS AND FUTURE WORK

Final comments

The project will take the data sets which contains web-attack and non-web-attack. The data sets undergo clean data by removing the unwanted words and special symbols. After that the clean description is converted into a sequence of words, after that redundancy is removed by doing a word stream count computation, after that Weighted Stream and Number of datasets in which word is present. After that the classification of data is done by using a classification algorithm.

Future work

1. This project can be future improved by having subjective category of web attacks

REFERENCES

[1] D. Hendler, S. Kels, and A. Rubin, “Detecting malicious PowerShell commands using deep neural networks”, in Proceedings of Asia Conference on Computer and Communications Security, ACM, 2018. DOI: 10.1145/3196494.3196511.
[2] J. Saxe, and K. Berlin, "Deep neural network based malware detection using two dimensional binary program features," in International Conference on Malicious and Unwanted Software (MALWARE), IEEE, 2015.
[3] Z. Yuan et al. (2014). Droid-sec: deep learning in android malware detection. ACM SIGCOMM Computer Communication Review. 44(4).
[4] M. Y. Kim, and D. H. Lee. (2014). Data-mining based SQL injection attack detection using internal query trees. Expert Systems with Applications. 41(11), pp: 5416-5430.
[5] M. Du et al, "Deeplog: Anomaly detection and diagnosis from system logs through deep learning," in Proceedings of ACM SIGSAC Conference on Computer and Communications Security, ACM, 2017.
[6] T. Rashid, I. Agrafiotis, and J. RC Nurse, "A new take on detecting insider threats: exploring the use of hidden markov models," in Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats, ACM, 2016.
[7] Y. Han et al, "Watch Me, but Don't Touch Me! Contactless Control Flow Monitoring via Electromagnetic Emanations," in Proceedings of ACM SIGSAC Conference on Computer and Communications Security. ACM, 2017.
[8] L. Zhu, X. Tang, M. Shen, X. Du, M. Guizani, “Privacy-preserving DDoS attack detection using cross-domain traffic in software defined networks,” IEEE Journal on Selected Areas in Communications, Vol.36, No.3, pp. 628-643.
[9] K. Zhang, S. Leng, X. Peng, L. Pan, S. Maharjan and Y. Zhang, "Artificial Intelligence Inspired Transmission Scheduling in Cognitive Vehicular Communications and Networks", IEEE Internet of Things Journal, Vol. 6, No. 2, pp. 1987-1997, Apr. 2019.
[10] M. Shen, B. Ma, L. Zhu, R. Mijumbi, X. Du, J. Hu, “Cloud-based approximate constrained shortest distance queries over encrypted graphs with privacy protection,” IEEE Transactions on Information Forensics and Security, Vol.13, No.4, pp. 940-953.