Internet Worm Detection based on Traffic Behavior Monitoring with Improved C4.5

S. Divya[a], Dr. G. Padmavathi[b]

[a]Research scholar, [b]Professor and Head,
Department of Computer Science, Avinashilingam University, Coimbatore, India

Abstract- Internet worms are identified as one of the serious security threats caused by their anomalous behaviors. Worms in the network cause many cyber security threats such as distributed denial of Service, illegal network traffic, spreading spam and stealing personal user information. In this paper, the network traffic is monitored and their abnormal behavior on the Internet is detected and classified based on attribute payload using C4.5 algorithm with Pearson's Correlation Coefficient. The proposed approach detects the Internet worm activities based on their traffic behavior using a decision tree algorithm. The categorization of continuous attributes is performed by C4.5 algorithm to test the distinct values and the method reduces the processing time of the large input. The experimental results obtained identify the unknown worms with improved accuracy in detecting malicious flows. The results obtained show improved precision value; recall value and better accuracy detection of Internet worms.

Keywords: Traffic flow, Attribute vector, Information Gain, Pearson Correlation Coefficient.

1. Introduction

In the Internet world today, Worms cause billion dollars damage every year throughout the world. Internet worms are those malicious codes which propagate automatically by themselves. Moreover, they do not need any human intervention for their propagation into the vulnerable hosts. The worms spread in the network affects the computers by stealing their confidential information, deleting files, reducing the speed of network functioning, creating a Distributed Denial of Service (DDOS) and with the infected hosts, they also further damage other hosts by launching attacks [1] [5] [10] [12].

Using epidemic spreading style, Nimda and Code Red Worms caused immense damage in the Internet world during 2001. In 2003, Slammer worms within 3 minutes, scanned more than 55 million machines and damaged within 10 minutes, nearly 90% of vulnerable hosts in the network [8]. In 2004, Witty Worm affected more than 12,000 vulnerable hosts within minutes and in 2007, Storm worm damaged millions of computers [2]. Conficker worm in 2008 infected the cloud network and controlled 6.4 million vulnerable hosts globally in 230 countries [9] [11]. Information securities top research is to stop the propagation of worms on the internet.

Internet worms creating illegal traffic behaviors are one of the challenges of existing in the network. These intrusions found in networks are done by worms in the form of payload replication and malicious traffic behaviors. Packet payload analysis will not provide better network security if the contents are encrypted. To overcome the above limitation with payload monitoring, traffic should be scanned and detected. Existing System lacks in handling the missing values when they are huge and are stored only once in numeric attributes [3]. The approach proposed achieves better detection accuracy by reducing the missing values and grouping the continuous attributes. C4.5 with Pearson's Correlation Coefficient minimizes the data file input space and communication overhead.

The organization of the paper is as follows: Section 2 reviews the related works on Internet Worm detection. Section 3 describes the proposed methods for the Internet Worm detection based on attribute

Downloaded from www.edlib.asdf.res.in

payload. Section 4 describes the experimental evaluation results for detection. Finally, in section 5 paper is concluded.

2. Related Works

Internet worms infect the network through illegal traffic flow. Monitoring and detecting the malicious traffic behavior provides better and faster communication. Rather than payload Inspection, traffic flow monitoring detects the network traffic and exploits the internet worms illegal traffic. Various techniques proposed for Internet worm detection are listed below.

Chao Chen et al. [2] proposed a novel approach, that it diminished the future internet epidemics using effective technique of Divide-conquer-scanning worm. This technique is faster and stealthier than the random-scanning worm. In this paper author also described two defense mechanism, they are infected host removal and active honey nets. Deguang Kong et al. [4] Introduced a novel method for detecting the network based worm. It first generates the signatures automatically by Semantics Aware statistical algorithm. This is used to remove the non-critical bytes, which is combined with a hidden Markov model to automatically generate worm signatures.

Jun-qun et al. [6] Analyzed to find the vulnerable host. Here the author implemented the gradual hybrid anti-worm. This approach was combination of active and passive anti-worm. The work done by the active anti-worm was detecting the vulnerable host on the network and patches them up. Listening process was handled by passive anti-worm, that it attacks the worm from the host after patching it for the process.Qian Wang et al. [7] Proposed the approach to analyze the internet worm infection family tree and it is named as worm tree. Through mathematical analysis, captures the key characteristics of the internet worm detection and applying it for bot detection.

Yu Yao et al. [13] Implemented an approach based on time delay to reduce the network worm and also decrease the economic loss rate. In this paper, one critical value is derived. If the time delay is greater than the critical value, then the worm will be eliminated from the network. Zaki et al. [14] Introduced WSRMAS; an anti-worm system. This approach effectively reduces the spreading of the infected worms in network routers and consists of a multi-agent system that can limit or even stop the worm spreading.

Table 1 lists the different techniques proposed and implemented by various authors and the parameters used for their experimentation and evaluation. The observations show the achievement of the methods and they are listed in the table 1 below.

Table 1. Review of Literature for Internet Worm Detection

Year	Author	Technique(s) Used	Parameters Used	Observations
2010	Chao Chen et al	Divide-conquer-scanning worms	Scanning Rate, Scanning Probability, and Scanning Space	Analyze the characteristics of DCS worms and potential countermeasures.
2010	Zaki and Hamouda 2010	Multi_Agent system	Number of machines, Detection time, Worm spreads interval, Anti-worm spreading interval, Anti-worm movement interval	Centralized planning capability improves the system effectiveness by decreasing the percentage of infected machines with about 40%.
2011	Deguang Kong et al	Semantics Aware Statistical algorithm	False Positive, False Negative	Accurately detect worms with concise signatures. Fast in online detection speeds, better in noise tolerance.

Downloaded from www.edlib.asdf.res.in

2011	Jun-qun et al	Gradual Hybrid Anti-Worm System	Transformation threshold rate	Detect and attack the worm using active and passive anti-worm, then patches up.
2011	Yu Yao et al	Time delay in quarantine	το	Network worm was detected and eliminated using time delay and decreases the window size.
2012	Qian Wang et al.	Probabilistic Modeling and Sequential Growth Model	Geometric Distribution with parameter 0.5	For forensic analysis, bots from bot assessment is exposed to worm tree.

From the above table 1, the different methods have been proposed to detect the Internet worms infecting the network. From the observations, it is found that they detect through monitoring payload and traffic misbehaviors. Payload detection lacks detection of worms when they are encrypted. Monitoring traffic behavior detects only after their spread. To overcome the above limitations, the proposed approach detects the Internet worms by monitoring the traffic flow collection.

3. Proposed Methodology

The proposed approach finds out the malicious traffic based on the characteristics of network flow using improved C4.5 algorithm. To classify the Internet worms, TCP and UDP flows are examined, they are split into time windows and attributed vector is extracted. Based on the attribute vectors malicious and non-malicious traffic is detected and classified. The existing method REPTree is a decision tree learner, uses information gain as splitting criterion. Its limitation is that the numeric attribute values can be sorted once only. C4.5 with Pearson's Correlation Coefficient gives an efficient classification between the malicious and non-malicious in network traffic based on their flow characteristics. Figure 1 below shows the complete process of detecting Internet worms through their traffic flows.

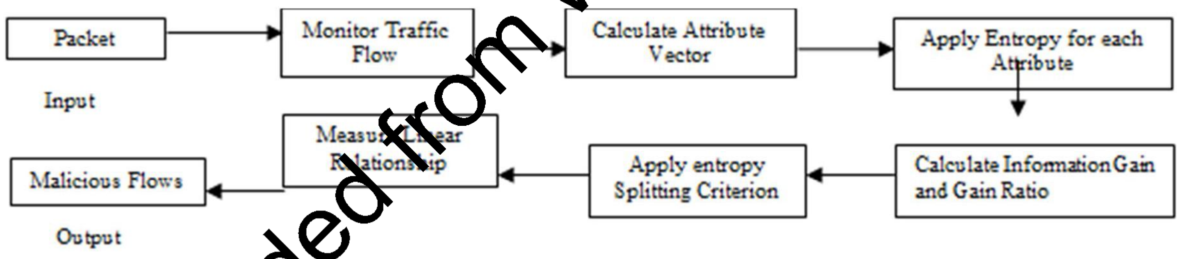
Packet
Input
Monitor Traffic Flow
Calculate Attribute Vector
Apply Entropy for each Attribute
Calculate Information Gain and Gain Ratio
Apply entropy Splitting Criterion
Measure Linear Relationship
Malicious Flows
Output

Figure 1. Proposed flow of the overall process

The steps followed for monitoring and detecting Internet worms based on the network flow characteristics is shown in the table 2 below

Table 2. Steps Proposed for Malicious Traffic Flow Detection

tep 1: Create an initial node and calculate Attribute Vector for incoming flow
Step 2: Apply entropy, Information Gain and Gain Ratio for each attribute
Step 3: Select highest Information Gain for attribute A, for best splitting criterion.
Step 4:Consider the best splitting criterion and partition the flows.
Step 4: If the linear relationship value exceeds then set as Malicious Flow occurred.

Downloaded from www.edlib.asdf.res.in

The above figure1 and table 2 shows the proposed methods procedure and the steps involved in detecting the Internet worms based on their illegal traffic flow.

A. Selection of Attribute vector

Attribute represents the numeric value that contains the collection of flows gathered at a time Window T given. It consists of source and destination IP address and ports of flow of destination and source, average time for the packets and average length of packets exchanged in the time interval, for effective detection of internet worms.

Attribute vector consists of the collection of attributes, that gathers the characteristics of individual flow for a specified time interval. The attribute vector is measured by comparing the total number of flows made by a particular single address with the total flows count made in some limited time period. The techniques proposed are decision tree's C4.5 algorithm with Pearson correlation coefficient.

B. C4.5 Algorithm

C4.5 is one of the decision tree based algorithm with a big tree and it contains certain attributes values and finalizes the decision rule using pruning method. It has features such as handling missing values, categorization of continuous attributes, pruning of decision trees and rule derivation.

The most significant attributes are selected by considering all the samples, in which root nodes are considered as the top nodes of the tree. The subsequent nodes, which are termed as branch node, receive the sample information. The decision is made when it is terminated in the leaf node. Root node to leaf node is a path defined by several notes in which rules are generated.

Some limitations of C4.5 algorithm are empty branches, insignificant branches and over fitting. Empty branches make the tree bigger and more complex. Insignificant branches reduce usage of decision tree and over fitting. Over fitting branches picks up the data with uncommon characteristics.

To overcome the limitations and detect the Internet worms, steps are considered for constructing C4.5 algorithm in the table 4 below

Table 4. Steps to Detect using C4.5 Algorithm

Step 1: If all cases are of the same class, the tree is a leaf and is returned labeled with this class. Step 2: For each attribute, calculate potential information provided by a test on the attribute. Step 3: Also calculate the gain in information that results from a test on the attribute. Step 4: Find best attribute to branch depending on the current selection criterion.

C. Counting Gain

Here entropy is implemented and is defined as to measure or calculate the disorder of the data. It is defined as $Entropy(\bar{y}) = -\sum_{j=1}^{n} \frac{|y_j|}{|\bar{y}|} \; log \frac{|y_j|}{|\bar{y}|}$ (1)

Iterating over all possible values of$|\bar{y}|$. The conditional Entropy is $Entropy(j|\bar{y}) = \frac{|y_j|}{|y|} log \frac{|y_j|}{|\bar{y}|}$ (2)

The gain is defined as $Gain(\bar{y}, j) = Entropy(\bar{y} - Entropy(j|\bar{y}))$ (3)

Downloaded from www.edlib.asdf.res.in

The goal is to maximize the gain, dividing by overall entropy due to split the argument $\bar{y}$ by value j.

From the above equation 3, entropy has some limitations. They are listed below

If a large number of distinct values are used by both continuous and discrete attributes, then it provides poor result.

There is no particular technique for predicting the information gain also this information gain is generated after attributes value generation. The system gives the less performance and accuracy based on the mismatch of the attribute value.

The system become a failure because of the amount of attribute is very much higher than the information gain.

More difficult to select the next attribute value, if the previous attribute value is less than it leads to unconditional selection of attributes.

When same valued attributes are used in the decision tree generation, they split up is a complex task; gives unbalanced trees.

To overcome the above limitations in C4.5, the proposed approach introduced Pearson's Correlation Coefficient and is used for entropy. This solves the unconditional selection of attributes, poor result, less performance and accuracy based on the mismatch of the attribute value, uncertainty in entropy.

Downloaded from www.edlib.asdf.res.in

D. Improved C4.5 Algorithm

To overcome this entropy limitation of C4.5 algorithm, improved C4.5 algorithm is introduced. Entropy is used to find out the linear relationship between two variables by comparing their strength and direction. These variables are determined by -1 to +1. Obtain the maximum value of +1 using perfect linear relationship by increasing relationship. Attain the gain -1 using perfect linear relationship by decreasing relationship. For not linear case it attains the value of zero (0)

Let X and Y be the two interval or ratio variables. Joint distribution of these two variables is called bivariate normal. Pearson's Correlation Coefficient is used for evaluating the entropy. Equation 4 gives the formula for Pearson's Correlation Coefficient

$$r_{xy} = \frac{\sum XY - \frac{(\sum X)(\sum Y)}{n}}{\sqrt{(SS_X)(SS_Y)}} \text{ or } \quad (4)$$

$$r_{xy} = \frac{\sum XY - \frac{(\sum X)(\sum Y)}{n}}{\sqrt{\left[\left(\sum X^2 - \frac{(\sum X)^2}{n_x}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{n_y}\right)\right]}} \quad (5)$$

Where, ΣX is the sum of all the X scores, ΣY is the sum of all the Y scores, $\sum X^2$ is square of each X score and then total of them, $\sum Y^2$ is square of each Y score and then total of them, $\sum XY$ is multiply of each X score by its associated Y score and then add of the resulting products together.

Table 5 below shows the pseudocode for the proposed method. The algorithms used are C4.5 algorithm with the Pearson correlation coefficient for better accuracy detection rate.

Table 5. Pseudocode for proposed approach

```
Input: Dataset containing tuples with a set of attributes, D
Output: Decision tree with highest information gain
Procedure
        Create an initial node N
        Apply entropy for each attribute in the dataset
        Calculate information gain and gain ratio for each attribute in a dataset
        Select highest information gain of attribute A (D, attribute list) to find the best splitting criterion
        While not end of attributes do
           If the dataset is partitioned with a single attribute A then
               Apply entropy and information gain for attribute A
               Apply gain ratio for attribute A
               Select highest information gain of attribute A to find the best splitting criterion
           Else
               Declare as leaf node
           End if
        End while
```

In the above table5, the traffic flows collected are monitored and the malicious flows are detected using C4.5 algorithm steps. Further enhancing its accuracy detection, Pearson coefficient correlation is integrated.

Figure 2 below shows the detailed flow of proposed process C4.5 with Pearson coefficient correlation for detecting and classifying Internet worms based on their network traffic flow characteristics.

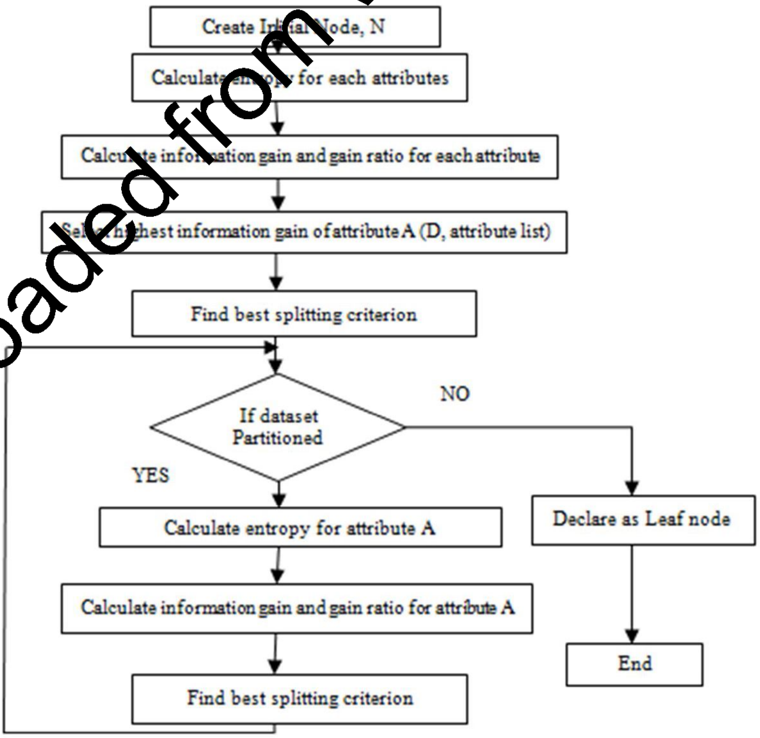
Create Initial Node, N
Calculate entropy for each attributes
Calculate information gain and gain ratio for each attribute
Select highest information gain of attribute A (D, attribute list)
Find best splitting criterion
If dataset Partitioned
NO
YES
Calculate entropy for attribute A
Declare as Leaf node
Calculate information gain and gain ratio for attribute A
End
Find best splitting criterion

Figure 2: Flowchart for Proposed Approach

Downloaded from www.edlib.asdf.res.in

The proposed approach at the network level monitors the illegal traffic behavior based on their attributes and classifies the detected Internet worms. Figure 2 above shows the proposed approach uses the decision tree algorithm C4.5 with Pearson's Correlation Coefficient for detecting and classifying the existence of Internet Worms in the network.

4. Experimental Results

The proposed systems are evaluated by using various parameters such as precision value, recall value and accuracy.

Precision value - Precision refers to the retrieved document. This is calculated by the total number of relevant documents divided by the total number of resultant documents.

$$Precision\ Value = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall value- Recall value is referred to as the relevant documents that are related to the search request.

$$Recall\ Value = \frac{True\ Positive}{False\ Positive + False\ Negative}$$

Accuracy-Accuracy provides the required related documents/measures used for classification.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + \ False\ Positive + True\ Negative + False\ Negative}$$

The proposed system is implemented using Java. Benchmark dataset is collected from the internet through the web. The dataset contains the total of 5, 03,219 data. The dataset contains both malicious and non-malicious traffic data. This proposed work compared with the existing system of reduced error pruning tree provides better accuracy in the detection of malicious traffic flows.

Table 6. Comparison of existing and proposed approach performance

Parameters	Existing REPTree	Proposed C4.5 with Pearson Correlation Coefficient	% of Improvement
Precision Value (%)	67%	81%	21%
Recall Value (%)	79%	88%	11%
Accuracy (%)	65%	76%	17%

The table 6 provides the comparison of parameters between existing reduced error pruning method and proposed improved C4.5 algorithm. The given parameters are Precision Value in (%), and Recall Value in (%) and Accuracy in (%).

Downloaded from www.edlib.asdf.res.in

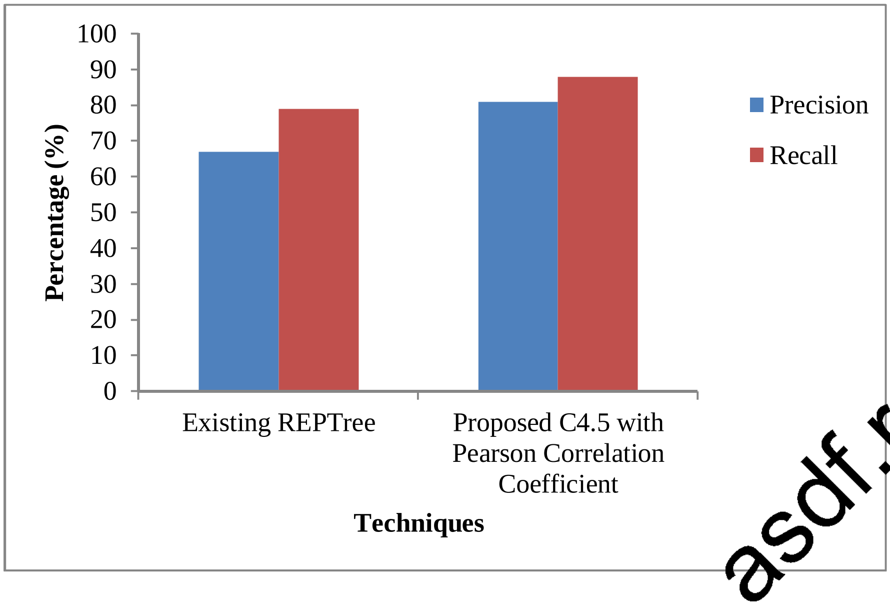

Figure 3. Comparison of recall and Precision

Figure 3 above represents the precision and recall value obtained by the existing and proposed approach. The results illustrate that C4.5 with Pearson's Correlation Coefficient performs better accuracy than the existing reduced error pruning algorithm.

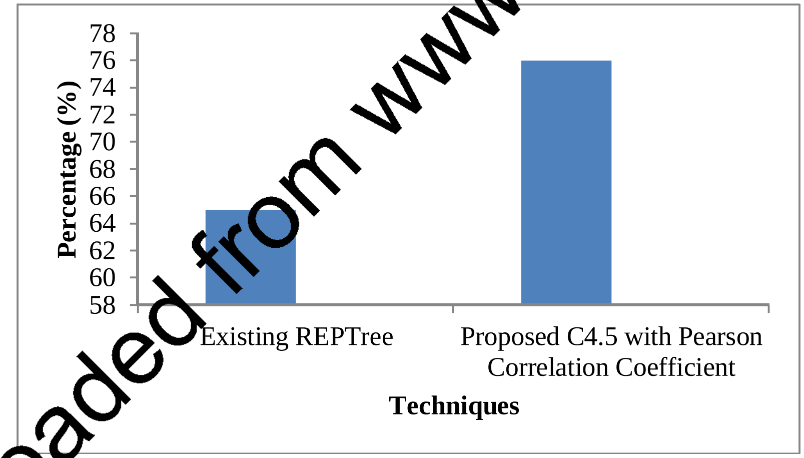

Figure 4. Comparison of Accuracy

From the above figure 4, overall accuracy obtained by existing reduced error pruning is 65% and proposed C4.5 algorithm is 76%. From the figure 4, it is clearly noticed that the proposed technique of C4.5 algorithm gives better accuracy level and improved performance than existing approach.

5. Conclusion

Internet worms are serious and challenging threats in the network and communication security. In this paper, malicious and non-malicious payload traffic is detected based on attributes. Improved C4.5 with a correlation coefficient improves the accuracy of detection overcoming the limitations of existing approaches. Based on the traffic flow characteristics, the attribute vector calculates the continuous flow of attributes. Moreover, the proposed approach provides the decision tree with highest information gain. The

Downloaded from www.edlib.asdf.res.in

proposed method provides better detection accuracy with high precision and recall value than the existing method.

References

1. Aliyu Mohammed, Sulaiman Mohd Nor and Muhammad Nadzir Marsono, “Analysis of Interent Malware Propagation Models and Mitigation Strategies”, International Journal of Computer Networks and Wireless Communications, Vol.2, No.1, 2012, pp. 16-20.
2. Chao Chen, Zesheng Chen and Yubin Li, “Characterizing and defending against divide-conquer-scanning worms”, ELSEVIER, Computer Networks, Vol. 54, 2010, pp. 3210-3222.
3. David Zhao, Issa Traore, Bassam Sayed, Wei Lu, Sherif Saad, Ali Ghorbani and Dan Garant, “Botnet detection based on traffic behavior analysis and flow intervals”, ELSEVIER, Computers & Security, Vol .39, 2013, pp. 2-16.
4. Deguang Kong, Yoon-Chan Jhi, Tao Gong, Sencun Zhu, Peng Liu and Hongsheng Xi, “SAS: semantics aware signature generation for polymorphic worm detection”, Springer, International Journal of Information Security, Vol .10, 2011, pp. 269-283
5. Fangwei Wang, Yunkai Zhang, Changguang, Jianfeng Ma and SangJAe Moon, “Stability analysis of SEIQV epidemic model for rapid spreading worms”, ELSEVIER, Computers & Security, Vol.29, 2010, pp. 410-418.
6. Li Jun-Qun, QIN Zheng, OU Lu, O. Salman, A. X. Liu and YANG Jin-min, “Modeling and analysis of gradual hybrid anti-worm”, Springer, Journal of Central South University of Technology, Vo. 18, 2011, pp. 2050-2055.
7. Qian Wang, Zesheng Chen and Chao Chen, “On the Characteristics of the Worm Infection Family Tree”, IEEE Transactions on Information Forensics and Security, Vol. 7, No.5, 2012, pp. 1614-1627.
8. Qian Wang, Zesheng Chen and Chao Chen, “Darknet-Based Inference of Internet Worm Temporal Characteristics”, IEEE Transactions on Information Forensics and Security, Vol. 6, No. 4, 2011, pp. 1382-1393.
9. Rizwan REhman, G.C. Hazarika, Gunadeep Chetia, “Malware Threats and Mitigations Strategies: A Survey”, Journal of Theoretical and Applied Information Technology, Vol.29, No.2, 2011, pp. 69-73.
10. Tonmoy Saikia, Ferdous A BArbhuiya and Sukumar Nandi, “A Behavior Based Framework for Worm Detection”, ELSEVIER, Proceedings of 2nd International Conference on Communication, Computing & Security, Vol.6, 2012, pp. 1011-1018.
11. Xufei Zheng, Tao Li and Yonghui Fang, “Strategy of fast and light-load cloud-based proactive benign worm countermeasure technology to contain worm propagation”, Springer, The Journal of SuperComputing, Vol.62, 2012, pp. 1451-1479.
12. Yong TANG, Jiaqing LUO, Bin XIAO and Guiyi WEI, “Concept, characteristics and Defending Mechanism of Worms”, IEICE Transactions on Information and Systems, Vol .E92-D, No .5, 2009, pp.799-809.
13. Yu Yao, Xiao-Wu Xie, Hao Guo, Fu-Xiang Gao and Xiao-jun Tong, “Hopf bifurcation in an Internet worm propagation model with time delay in Quarantine”, ELSEVIER, Mathematical and Computer Modelling, Vol. 57, No. 12, pp.2635-2646.
14. Zaki.M, Hamouda.A.A, “Design of a multi_agent system for worm spreading-reduction”, Springer, Journal of Intelligent Information Systems, Vol. 35, 2010, pp. 123-155.

Downloaded from www.edlib.asdf.res.in