Study on Anti-worm Diffusion Strategies Based on B+ Address Tree

Dewu Xu
College of Mathematics, Physics & Information Engineering, Zhejiang Normal University, Jinhua, Zhejiang, 321004, China
Email:xdw_zjnu@126.com

Jianfeng Lu and Wei Chen
College of Mathematics, Physics & Information Engineering, Zhejiang Normal University, Jinhua, Zhejiang, 321004, China
Email:{lujianfeng, chen_wei}@zjnu.cn

***Abstract*—To improve the efficiency of resistance of anti-worms to malicious worms and enhance their diffusibility, diffusion strategies of anti-worms based on B+ address tree are proposed in order to speed up anti-worm diffusion rate in the network and reduce anti-worm influence on network system when diffused. The diffusion strategies are simulated by scilab. Results show that anti-worms using B+ addresses tree strategies have faster diffusion speed and less traffic impact on network compared with traditional strategies.**

***Index Terms*—anti-worm, propagation mode, diffusion strategies, B+ address tree, active diffusion, detection host**

I. INTRODUCTION

In recent years, network worm threats to the computer system and network security have rapidly increased. Active detection worms represented by CodeRed, Blaster, and Slammer and E-mail worms represented by Melissa, LoveLetter, and MyDoom live longer, cover a wider area, and have caused tremendous damage to information systems[1]. Researchers have developed a variety of approaches[2-4] to prevent the network from damages caused by worms, but the defensive approach is after all a temporary solution and a manageable method of proactive protection is urgently needed. Anti-worms can proactively fix the vulnerabilities of the host before the outbreak of worms or during the early stage of the outbreak in order to control the scope of the worm outbreak. The use of anti-worms to combat malicious worms is becoming a new emergency measure.

At present, anti-worm diffusion strategies and proactive counter strategies need to be improved. The current process of anti-worms against malicious worms uses the same diffusion strategies as malicious worms. Such strategies have caused serious impact on the network during the counter process and put this technology into a heated debate in the long term; thus the pace of the research has been slowed down[5]. Based on the counter idea of anti-worms, this paper has designed practical B+ tree address diffusion strategies to reduce network traffic.

At present, worm diffusion strategies include uniform random diffusion, local priority diffusion, diffusion based on the target list, diffusion based on K-Way algorithm[6], diffusion based on search engines, passive diffusion and so on. Some strategies could easily lead to network traffic overloading and network congestion; therefore effective strategies and algorithms to control the diffusion methods and speed of anti-worms are becoming more and more essential.

A uniform random diffusion algorithm randomly generates IP addresses from address space to be detected to carry out the diffusion, which will produce a large amount of abnormal traffic.

In contrast, a local priorities diffusion algorithm generates a subnet IP address of the infected host so as to increase the ability to detect the target host and reduce the abnormal flows. Additionally, such a strategy can exclude the unallocated and retained IP addresses of the address space to be detected.

A diffusion algorithm based on the target list generates a pre-test target address list, and then tentative diffusion is done according to the destination addresses in the list[7]. The algorithm generates a target address list based on routing table information and the generated diffusion rate of worms is 3.5 times the rate of the random scan algorithm. But the drawback is that the IP address library must be carried during the diffusion, which increases the load traffic.

The K-Way diffusion algorithm was originally used in flash worm. The basic idea is to build all websites to be probed into *M* K-Way trees. All nodes within each tree are 1/K of total nodes and 1/K node sets are mutually disjoint with each other. Each node is an internal node of a tree and all internal nodes at most have K branches. Those internal nodes are leaf nodes within other trees. This greatly improves the stability of flash worm

Manuscript received October 8, 2011; revised December 29, 2011; accepted January 18, 2012.
Project number: Y201120829 , Y1110483 and 60873234
Contact author: Dewu Xu

doi:10.4304/jcp.7.9.2093-2098

diffusion, but at a price: the code becomes complex, and the scanning numbers of the target nodes become larger.

Increasingly powerful search engines have lead to the emergence of "Google Hacking" which represents this kind of attack means and method. Santy, the first smart worm that used a search engine to find the attack target, shows that the idea of a smart worm has been realized. The development of smart worms signifies that network security has converted from a scattered counter into an overall security alliance.

With the passive diffusion algorithm, worms are latent in the infected hosts first, and they monitor network data packets to obtain other users' activity information through the host, with loopholes actively contacting infected hosts in order to find new target hosts. Therefore, in the process of finding the target hosts, abnormal network traffic will not be generated, which makes it difficult for the detection system to find. But the drawback is that the diffusion rate is slow when the target number is small or scan frequency is low, and the diffusion rate is fast when scan frequency is high.

There are four factors that affect the propagation speed of worms: selection of target address space, whether it adopts a multi-thread search for vulnerable hosts, whether there is a list of susceptible hosts, and diversity of propagation. The main difference of each diffusion strategy lies in the selection of target address space. The key to fast and light-load propagation lies in the design of diffusion strategies. Compared with the passive anti-worm strategy proposed in [8], anti-worms that include proactive information collection modules can automatically look for hosts with loopholes in the network, penetrate the target hosts, and propagate a duplicate copy and repair tools. The propagation of anti-worms under this counteractive strategy is very fast. The difference is that after the penetration of the hosts, anti-worms carry out the tasks such as antivirus and repair, which is called the counteractive model of proactive diffusion.

II. DIFFUSION STRATEGY OF ANTI-WORM B+ ADDRESS TREES

A. Divination of Anti-worm Proactive Diffusion and B+ Address tree

The strategy of parallel transmission of information with proactive diffusion is used in this study. Control information is taken in the process of information transmission so that anti-worms are in controllable diffusion state.

B+ tree is the general form of multi-channel expansion of "binary search tree". It is used to manage and maintain large-scale data index with high effective random query, low update overhead, self-balancing and so on. B + tree further allows for all "keys" to appear only in the leaf nodes and links up all the leaf nodes in a manner of list in order to improve the query efficiency.

This study uses the above advantages of B+ tree to design diffusion strategies. The definition of B+ tree is as follows:

Definition 1: B+ tree address

1. All IP addresses are 0-order B+ address tree. When the number of hosts being detected is M, the order of B+ address tree is L;

2. Assuming that T_i is a T_0 –rooted i-order B+ address tree, add an IP address group to all the nodes of T_i, the resulting T_{i+1} is a T_0 –rooted i+1-order B+ address tree.

3. Let i = i + 1. When i ≥ L, Server and Center no longer allot IP address groups to the newly-detected penetrated hosts. During the drawing, penetrated hosts are put below the existing hosts that do not participate the detection; when i <L, go to 2;

4. Only trees obtained by 1 and a certain number of 2 and 3 are called N-order B+ address tree.

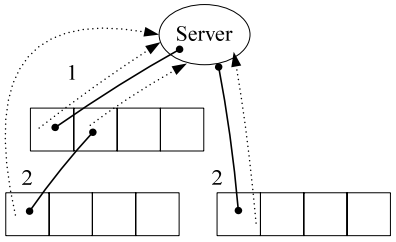
Therefore, the maximum order that participates in the detection in B+ address tree is L.

Assume that worms are not affected by worms or before the penetration of anti-worms are all loophole hosts and that loopholes are 0day loopholes. Even without the patch, technicians can take temporary protective measures that are applied in anti-worms to guard the loophole hosts until release of the patch. To study this problem, we further assume that:

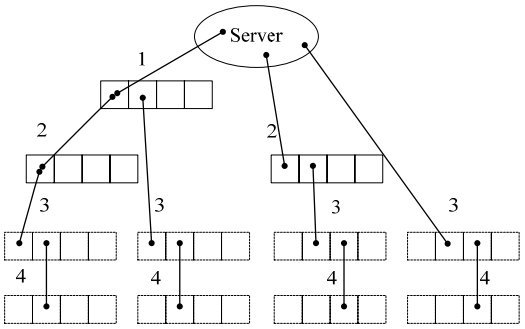
All node hosts within the network can visit each other, the propagation delay between any two points is almost the same. Broadband network, switching network and fully connected internal switching fabric can be regarded as such an environment.

This paper descripts the diffusion in the manner of a target list. Divide the whole list of IP detection addresses into N IP addresses with n IP addresses in each group. Assume that in the process of anti-worm diffusion, the maximum number of detection hosts is M (M<N), and all the penetrated hosts are allocated an IP address group before the number of detection hosts reaches M. Detection hosts that are already allocated an IP address group only detect hosts within the allocated IP address group and send feedbacks to the Center (or Server) after the detection.

Figure 1 shows 2-order and 4-order B+ address trees consisting of an IP address group with 4 IP addresses. Each small box represents a host and the IP address group is represented by a rectangle composed of 4 small squares. The solid line represents loophole hosts that are already detected by detection hosts, and the dash arrow indicates the feedback from the penetrated hosts to the Server (4-order address tree in the figure omits the feedback dash arrow). The dash box indicates that penetrated hosts within the IP address group do not participate in the detection.



a) 2-order B+ address tree



b) 4-order B+ address tree with 4 detection hosts

Figure 1 Schematic diagram of 2-order and 4-order B+ address tree.

It can be seen from Fig.1, Definition 1 and the definition of ET tree that ET tree is a special case of B+ address tree. When all the IP address groups contain an IP address, the B+ address tree is an ET tree. Because this study provides that all penetrated hosts send feedbacks to the Server, the B+ address tree is more stable than the ET tree.

Related theorems and proofs are given as follows:

Theorem 1:*T* is the B+ address tree with the maximum order *N*. When the number of detection hosts reaches *M*, the order of *T* is *L*. Then the total node number of *T* is $2^L + M * (N\text{-}L)$ and the total number of IP addresses are $n*(2^N + M * (N\text{-}L))$-n.

Proof: 1. When *N*=0, then *M*=*L*=0,so $2^N+M*(N\text{-}L) =2^0+0*(0\text{-}0) =1$; therefore in this case, the theorem is proved;

2. Assuming that when *N*=i (i!=0), the theorem is tenable, the nodes number of *N*-order B + address are $2^L + M*(\text{i-}L)$;

For *N*=i+1, according to definition 1, the order participating in the scan is the same *L*, the (i+1)-order B+ address tree increases just *M* nodes which are not involved in the scan compared to the i-order B+ address tree. So, the total node number of *T* is $2^L + M*(\text{i-}L)+M=2^L + M*(\text{i+1-}L)=2^L + M * (N\text{-}L)$.

Besides the original node, namely server node, every B + address tree nodes contains an IP address, so the total number of IP addresses is $n*(2^N + M * (N\text{-}L))$-n.

Therefore, we can see that this theorem is proven by the mathematical induction.

Theorem 2:Let *T* denote the *N*-order B+ address tree and *M* denote the total number of detection hosts (1 approaches to *M* in B+ address tree manner), it takes $n*\left\lceil (\lceil \Omega/n \rceil - \sum_{j=1}^{L} a_j) \Big/ M \right\rceil + L$ amount of time to diffuse the anti-worm duplicate to the IP address group of the *N*th-order leaf in a B+ address tree from the information source. And it takes $n*\left\lceil (\lceil \Omega/n \rceil - \sum_{j=1}^{L} a_j) \Big/ M \right\rceil + L + n - 1$ amount of time to diffuse the anti-worm duplicate to all IP address list spaces.

Proof: It can be seen from theorem 1 that the system needs *L* detection time before the number of hosts involved in the detection approaches *M*, at the same time the order of B+ address tree approaches to *L* from 0, all of these need *L* times. After the detection number reached *M*, *N*-order B+ address tree remains $\lceil \Omega/n \rceil - \sum_{j=1}^{L} a_j$ amount of the IP address group, which needs $n*\left\lceil (\lceil \Omega/n \rceil - \sum_{j=1}^{L} a_j) \Big/ M \right\rceil$ amount of time to retreat, so it takes $n*\left\lceil (\lceil \Omega/n \rceil - \sum_{j=1}^{L} a_j) \Big/ M \right\rceil + L$ amount of time to diffuse the anti-worm duplicate to the IP address group of the leaf in a B+ address tree from the information source. Moreover, it takes n-1 amount of time to finish detecting and penetrating *L*th-order in B+ address tree. Therefore, the theorem is proved.

Theorem 3: If the number of hosts that participate in the detection is *M*, then it can be represented as:

1. If *L*<n, then the relationship between *L* and *M* is $2^{L+1}\text{-}L \geq M+2 > 2^L\text{-}L+1$;

2. If *L*≥n, then the relationship between *L* and *M* is $2^{L+1}\text{-}2^{L\text{-n}+1}\text{-n} \geq M > 2^L\text{-}2^{L\text{-n}}\text{ - n}$.

Proof: Suppose the first IP address of IP address group in the *L*th-order is detected and participates in detecting, the number of hosts participating in detection is *M*. It can be seen from theorem 2 that this moment B+ address tree is *L*-order, then the number of hosts involved in detection in this B+ address tree is:

1. If *L*<n, then it can be seen from theorem 2 that (*L*-n+1<n-n+1=1)-order in B+ address tree has been detected. So, the total number of hosts participating in detection is presented as: $a_L+2*a_{L-1}+3*a_{L-2}+...+La_1$.

Owing to the following expression:

$$\begin{cases} a_L = \sum_{j=0}^{L-1} a_j = A_{L-1}, L \geq 1, a_0 = 1 \\ A_L = \sum_{j=0}^{L} a_j = a_L + \sum_{j=0}^{L-1} a_j \\ = 2a_L = 2A_{L-1} = 2^L, L \geq 1 \\ a_L = 2^{L-1}, L \geq 1 \end{cases}$$

all nodes are IP address group except the original node, namely server node, so the number of IP address group that *L*-order B+ address tree contains are:

$$\sum_{j=0}^{L} a_j - 1 = A_L - 1 = 2^L - 1$$

So, the following expression is correct: $a_L+2*a_{L-1}+3*a_{L-2}+...+La_1=2^{L-1}+2*2^{L-2}+3*2^{L-3}+...+L*2^0 =2^{L+1}\text{-}2\text{-}L \geq M$. Then, the relationship between *L* and *M* is $2^{L+1}\text{-}L \geq M+2$. For *M* takes the maximum value of the above formula, $M+2>2^L\text{-}L+1$. This theorem is proved.

2. If *L*≥n, then the order of nodes in which all allocated IP address groups have been detected is (*L*-n+1). So, there are (*L*-(*L*-n+1)=n-1)-order nodes which have not been detected. From the above derivation, the number of IP address, detected in IP address group still waiting to be detected are : $a_L+2*a_{L-1}+3*a_{L-2}+...+(\text{n-1})a_{L\text{-n}+2}= 2^{L-1}+2*2^{L-2}+3*2^{L-3}+...+(\text{n-1})*2^{L-n+1}=2^L*[2\text{-}(\text{n+1})*2^{-n+1}]$, the number of IP addresses that have been detected are $n(\sum_{j=1}^{L\text{-n}+1} a_j) = n*(2^{L-n+1}-1)$. So, the total number is

$$2^{L} * [2 - (n+1) * 2^{-n+1}] + n * (2^{L-n+1} - 1)$$
$$= 2^{L+1} - (n+1) * 2^{L-n+1} + n * 2^{L-n+1} - n$$
$$= 2^{L+1} - 2^{L-n+1} - n \geq M$$

Similarly, as M takes the maximum value of the above formula, then $M>2^{L}-2^{L-n}-n$. This theorem is proved.

Theorem 4: Suppose the number of IP addresses comprised by the whole IP address list is Ω,$\Omega=n*(2^{L} + M*(N-L))-n$, then through the comparison among the diffusion strategies based on B+ address tree, the Flash Tree diffused in K-Way manner in [6], and the exponential tree diffused in ET manner in [9], the diffusion speed has the following relationship:

1. If $K \leq M$,the propagation rate of B+ address tree is the same as the exponential tree in ET manner, and faster than the Flash Tree in K-way manner.

2. If $M<K<N$, the propagation rate of B+ address tree diffuses in a uniform manner.

The proof of the first section about this theorem can be found in [9], so it is omitted here.

Now, we will prove the second section: As the maximum number of machine that the B + address tree involved in detection and network penetration is M, so each new loophole host is not assigned an IP address after reaching the maximum number, the number of machines that execute diffusion tasks is only M in the whole address tree. So every time, there are only M machines that convert from loopholes host into penetrated ones.

It can be seen from theorem 1 that the number of IP address group of B+ address tree that comprises M detection hosts is $2^{L} + M*(N-L)-1$,and the total number of IPs in the IP address list is $n*(2^{L} + M*(N-L))-n$. But in fact, the number of IP addresses can be any value and the number of IP addresses changes dynamically during the process of detection and penetration. Therefore, we introduce the concept of B+ address deformed tree to make the above situation diffuse according to B+ address tree.

The definition of B+ address deformed tree is as follows:

Definition 2: B+ address deformed tree

Let the total number of IP addresses to be detected be Ω and all the IP address groups contain n number of IPs. For example, $\Omega=n*(2^{L} +M*(N-L))-n$,. Then according to definition 1, generate all the IP address groups into corresponding N-order B+ address trees; otherwise, let $n*(2^{L} +M*(N-L))-n<\Omega<n*(2^{L}+M*(N-L+1))-n$ and let $\alpha=\Omega-(n*(2^{L} +M*(N-L))-n)$, add the left α number of IP address groups separately to α number of different sub-nodes of N-order B+ address tree. This formation is called N-order B+ address deformed tree.

The basic idea of a B+ address deformed tree is to let $2^{L} + M*(N-L)-1$ number of IP address groups of Ω number of IP address groups constitute a complete N-order B+ address tree and connect the left nodes to the B+ address tree.

It can be seen from definition 1 and 2 that no matter what the address trees are, they all meet theorems 1-4; what is needed is to adjust the B+ address deformed tree as follows:

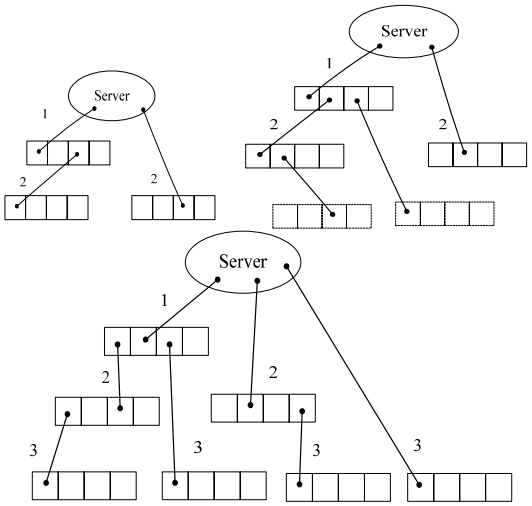


Figure 2. Schematic diagram of 2-order B+ address tree(on the above left), 2-order B+ address deformed tree(on the above right), and 3-order B+ address tree(on the below).

Fig.2 is the schematic diagram of a 2-order B+ address tree (on the above left), a 2-order B+ address deformed tree (on the above right), and a 3-order B+ address tree (on the bottom). The feedback dotted line is omitted. The Server point in the figure is the initial node and does not have an IP address group. The dash box represents the nodes that are added when the original B+ address tree is modified into a B+ address deformed tree.

B. Dynamic B+ Address Tree Generation Algorithm

Because the number of IP addresses in reality can be any value, the number of groups (the number of nodes) after the division of IP address list can also be any value. During the generation of a B+ address tree, IP address groups are allocated in a dynamic way. The steps of a dynamic B+ address tree generation algorithm are as follows:

1. Divide all the IP addresses that are to be tested into groups with n number of IP addresses in each group. Treat the left IP addresses that are less than n as one group; the total number of groups is $\lceil \Omega / n \rceil$.

2. When the Server detects, retrieve an IP address group according to the sequence of the IP address group list, then select one of the IP addresses (the loophole host is marked as A) to begin the detection.

3. After the success of the detection, penetrate that loophole host A. A duplicate copy of anti-worms in A feeds back the information to the Server and constitutes a connection. The Server takes the following steps to allocate the IP addresses: Server adds A into the penetrated host list and detection host list, then allocates the IP address group where A locates to A and lets A detect the rest of the loophole hosts with the group.

4. As a detection host, A detects the remaining loophole hosts in this group in turn. If the detection and penetration are successful, then the penetrated host (denoted as B) sends a request to the Server. If by now the total number of the detection hosts has not yet reached M, the Server will add B into the penetrated host

list and detection host list and then allocate an undetected IP address group to B; if the total number has reached *M*, the Server will not send and it will add B into the penetrated host list; if B does not send a request to the Server, the Server will add B into the corresponding host list according to the detection feedback of A.

5. Server and detection host repeat the above steps 2-4 and detect and penetrate all IP address groups.

Because the Server selects the process of Center and the process that Server periodically sends the confirmation information to the detection hosts does not have any influence on the constitution of B+ address tree, the above process doesn't need to be mentioned.

The dynamic B+ address tree generation algorithm has the following good properties:

1. In the process of dynamic generation of diffusion tree, nodes that enter the diffusion model at any time will not change their relationships that they had before those nodes entered the model and the balance of the whole tree will not be changed;

2. Nodes that enter the model earlier are sure to get the information earlier than or at the same time as those that enter the model later;

3. Regarding the given network, the structure of the corresponding B+ address tree is fixed; therefore the track of information diffusion can be learned in advance.

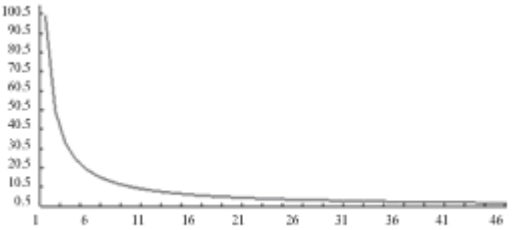
C. Analysis of Stability

It can be seen from the above algorithm that the B+ address tree needs only to divide IP addresses into groups before the detection, the construction of the whole tree is naturally completed in the process of detection, the construction does not need to be done in advance, and ET trees and K-way based flash trees have no such advantage. There are some problems with the stability of ET trees in [9] and K-way based flash trees in [6], for example, there will be many uncertainties for the diffusion trees that are built within the topology of an Internet virtual network; if some node in the spanning tree becomes a bad node because of the network or other reasons, child nodes below it will lose the chance of being infected.

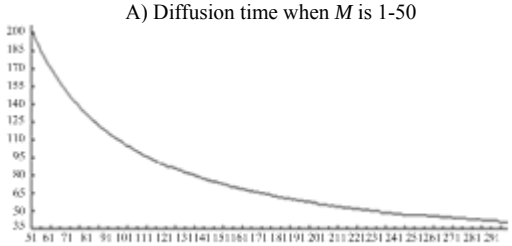
From the above algorithm, when the detected suspicious host is a bad point, this host is not penetrated, thus no duplicate copy of anti-worms will send a query to the Server and the follow-up action will not be performed. Therefore, there will be no problem for ET trees and flash trees. When the detection host that has been allocated an IP address group becomes a bad point during the detection for some reason, because the Server periodically sends confirmation information to the detection host, the only thing that will be delayed is the detection time of the IP address group that should be allocated.

III. SIMULATION OF DIFFUSION

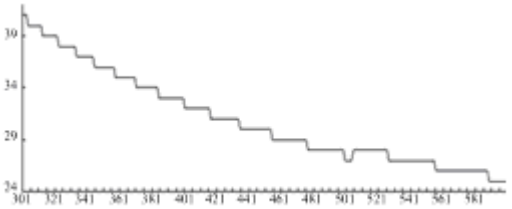
This section uses scilab to test the diffusion strategies of the B+ address tree. The number of detection hosts is denoted by *M*. It can be known from theorem 4 that the amount of time required for the duplicate copy of anti-worm to spread through the whole B+ address tree is $i+n*\left\lceil\left(\left\lceil\Omega/n\right\rceil-\sum_{j=1}^{L}a_j\right)/M\right\rceil+n-1$, Fig. 1 and Table 1 are the relationship between *M* and the diffusion time of duplicate copy of anti-worms spreading through the whole B+ address tree under condition Ω=10000,n=10:



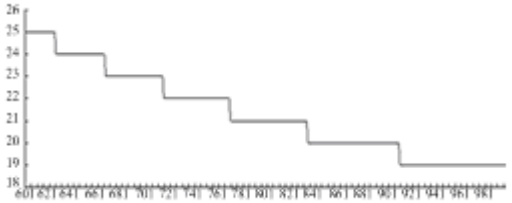
A) Diffusion time when *M* is 1-50



B) Diffusion time when *M* is 51-300



C) Diffusion time when *M* is 301-600



D) Diffusion time when *M* is 601-1000

Figure 3 Relationship between *M* and diffusion time of anti-worms

TABLE I.
RELATIONSHIP BETWEEN B+ ADDRESS TREE AND *M*

Order of B+ address tree	1	2	3
Number of detection hosts	1-3	4-6	7-13
Order of B+ address tree	4	5	6
Number of detection hosts	14-28	29-59	60-122
Order of B+ address tree	7	8	9
Number of detection hosts	123-249	250-504	505-1000

According to Fig. 3 and Table 1, we obtain the following conclusions:

1. The increment of the number of detection hosts does not necessarily accelerate the propagation speed of anti-worms or decrease the detection time;

2. The number of detection hosts can be chosen according to the situation of malicious worms.

(1) If malicious worms just begin to propagate, B map

can be chosen so that there will be little impact on the network traffic, and the propagation of malicious worms can be inhibited without influencing the working of the network. In the process of anti-worms detecting and penetrating, a user system that has been patched can be used as patch host.

(2) If the diffusion of malicious worms is serious, C map can be chosen so that the average number of detections of each detection host is about 20. Since the IP address is continuous, every segment only needs one host to eradicate its malicious worms, and it cannot influence the other segments.

(3) It is recommended that under normal conditions B should not be chosen, because, as Table 1 shows, in the late period of the propagation of anti-worms there are few malicious worms left. But there are still more detection hosts; thus the network data generated by anti-worms themselves during the detection is the main factor that influences the network. Certainly, it is hoped that a highly secure network can quickly eradicate malicious worms, and then D map can be selected.

Increasing the number of detection hosts will on the whole shorten the diffusion time of anti-worms. But in some situations, such increase does not necessarily shorten the diffusion time. Therefore for the sake of meeting a certain diffusion time, the number of detection hosts should be appropriately selected so as to reduce the impact on the network by anti-worms themselves while diffusing.

IV. CONCLUSIONS

Diffusion strategies based on B+ address tree are proposed in this study, which reduces the impact on the network system by anti-worms during the diffusion and enhances the diffusibility of anti-worms as well. It can be seen from the simulation that diffusion algorithms based on a B+ address tree are quite stable in terms of the network performance and the B+ address tree itself is also very stable.

ACKNOWLEDGEMENT

This research is financially supported by the National Nature Science Foundation of China under Grant No. 60873234, the Education Department Foundation of Zhejiang Province under Grant No. Y201120829 and the Nature Science Foundation of Zhejiang Province under Grant No. Y1110483. Thanks to the reviewers for the valuable comments helping to improve the quality of the manuscript.

REFERENCES

[1] WANG Xiu-ying, SHAO Zhi-qing, LIU Bai-xiang. Worm Detection Algorithm Under P2P Circumstances [J]. Computer Engineering. 2009(3): 173-175.

[2] J W Lockwood, J Moscola, M Kulig, et al. Internet worm and virus protection in dynamically reconfigurable hardware[C]. ACM CCS Workshop on Rapid Malcode (WORM 2003), Washington, 2003.

[3] N Weaver, V Paxson, S Staniford, et al. Large scale malicious code: A research agenda[OL]. http://www.cs.berkeley.edu/~ nweaver/, 2003.

[4] N Provos, A virtual honeypot framework[R]. Center of Information Technology Integration, University of Michigan, Tech Rep: citi-tr-03-1, 2003.

[5] Wang Bailing, Fang Binxing. A New Friendly Worm Propagation Strategy Based on Diffusing Balance Tree [J]. Journal of Computer Research and Development. 2006(9): 1593-1602.

[6] S. Staniford, D. Moore, V Paxson, et al. The Top Speed of Flash Worms[C]. In: Proc. ACM CCS Workshop on Rapid Malcode, Washington DC, USA, 2004:33-42.

[7] Staniford S., Paxson V., Weaver N. How to own the Internet in your spare time[C]. Proc of the 11th Usenix Security Symp. San Francisco: Usenix, 2002:149-167.

[8] Deng Ying-yi. Research on p2p worms and defense technology[D], ChengDu: University of Electronic Science and Technology of China, 2007.

[9] Wang Bai-ling. Friendly Worm Based Active Countermeasure Technology to Contain Network Worm [D]. Harbin Institute of Technology. 2006.

Dewu Xu received his M.S. degree from the School of Information Science and Technology at East China Normal University in 2005. He is now a lecturer in the College of Mathematics, Physics and Information Engineering at Zhejiang Normal University. His research interests include Cryptology and Distributed System Security.

Jianfeng Lu received his B.S. degree from the College of Computer Science and Technology at Wuhan University of Science and Technology in 2005, and his PhD degree from the College of Computer Science and Technology at Huazhong University of Science and Technology in 2010. He is a lecturer in the College of Mathematics, Physics and Information Engineering at Zhejiang Normal University. His research interests include Distributed System Security and Access Control.

Wei Chen received his PhD degree from the Beijing University of Post & Telecommunication in 2006. He is now an associate professor and tutor for graduate in the College of Mathematics, Physics and Information Engineering at Zhejiang Normal University. His research interests include Cryptology and Intrusion Detection.