Research Journal of Applied Sciences, Engineering and Technology 5(7): 2392-2398, 2013
DOI:10.19026/rjaset.5.4670
ISSN: 2040-7459; e-ISSN: 2040-7467
© 2013 Maxwell Scientific Publication Corp.
Submitted: July 26, 2012 Accepted: September 03, 2012 Published: March 11, 2013

Research Article

A Security Routing Algorithm of P2P Network Based on Multiple Encryption and Positive Detection

Lu Chuiwei, Wu Honghua and Liu Zhiyuan
Computer School, Hubei Polytechnic University, Huangshi, 435003, China

Abstract: Routing plays a fundamental role in the P2P network. Most attacks are aiming at P2P routing. We proposed a novel security routing algorithm to fight against those attacks. The algorithm adopt the means of multiple encryption and positive detection and periodically detect each node in routing path using encryption packet, which can accurately find malicious or instable nodes in routing path and exclude them from routing table. Simulation experiments also demonstrate the algorithm can effectively enhance the routing security and reliability of P2P network.

Keywords: Multiple encryption, P2P network, positive detection, routing security

INTRODUCTION

With the rapid development of information technology and the worldwide prevalence of the Internet, it has implied massive data resources, services resources, computing resources out of the Internet. It is P2P technology that provides an inexpensive and relevant simple solution to integrate these resources and build a new high-speed and large-capacity resources sharing platform. P2P network is a kind of logic network whose users can join or quit freely, which has brought numerous problems on security, efficiency, etc. The P2P routing algorithm is the core of P2P protocol. A robust, efficient and reliable routing algorithm plays a pivotal role in solving problems that P2P network is facing; hence, many research institutions are focused on it Though current attack ways toward the P2P network are numerous (Keith and Minseok, 2009), most of the attacks are against the routing (Stefan and Udo, 2009), which is because the destruction of P2P routing may result in the collapse of the whole P2P network, while the price is very low (BongSoo *et al*., 2008; Dagon and Zou, 2006). Since the P2P routing mechanism is the key factor to determine the capability of the system against the network attacks, the network stability, the information security and the communication performance. The design of routing becomes very important part among all the P2P protocols.

The common form of attacks towards P2P network mainly are worms, DDos attacks and falsification of identity attacks (Cheng and Friedman, 2006; Yu and Rexford, 2005), such as Sybil and Eclipse. Due to the huge scale of P2P network, it is not obvious damaging effect that a few malicious nodes launch attack. The study (Roger, 2005) indicates that only the proportion of malicious nodes reach 12% will result in significant damaging effect. Attackers usually infect the benign nodes in the P2P network by spreading virus, the infected nodes will then form a “zombie” network and while the number of the “zombie” is sufficient, attackers will kidnap them and suddenly launch a high-intensity attack toward the P2P network.

There are numerous researches on the aspects of P2P route attacks and defense (Atul, 2004; Zhou *et al*., 2006; Staniford *et al*., 2002; Kannan and Lakshminarayanan, 2003; Wei, 2004; Hongfei, 2003). The famous study (Anjali *et al*., 2004) pointed out that the overload of the routing information the nodes stored will cause the information cannot be update efficiently and in time. The study (Emil and Morris, 2002) Carried on a deep research and simulation on the DHT resource querying system’s security problem in P2P network. Then they proposed an effective improvement to detect and defend the attacks to P2P routing and destruction to the DHT resource querying system. The study (David and Dawn, 2006) got the statistics and the analysis of various P2P attacking and defending method, then pointed out these detecting and defending methods can only deal with certain type of attack, which can’t fundamentally eliminate the threat.

The study proposed a Security Routing Algorithm Based on Multiple Encryption and Positive Detection: SRABMEPD. It periodically detects the attribute of relaying nodes in routing path, finds malicious or instable nodes and excludes them from routing table. This measure can establish a safe and reliable routing path and help to optimize the performance of P2P network and improve its working efficiency.

Corresponding Author: Wu Honghua, Computer School, Hubei Polytechnic University, Huangshi, 435003, China

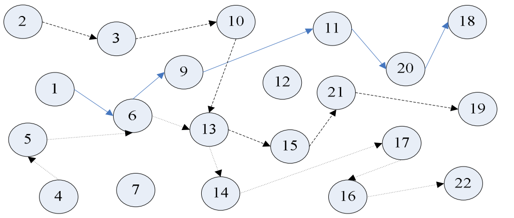

Fig. 1: Internal routing paths in P2P network

P2P NETWORK ROUTING MECHANISM

Structured P2P network system can be abstracted as a directed graph G = <P, E>, every node in P2P network will be mapped to a certain point in the graph and said with P_i, where, $0<j<n$, n is the total number of the nodes in P2P network. We define $C_{ij} = <P_i, P_j>$ as the communication relation between P_i and P_j. P_i will establish a routing path consisted by some relaying nodes to arrive at P_j. When the data are transmitting in the path, its source address, target address will be rewritten several times, which keep the malicious nodes from detecting the true information, thereby protecting the safety of data and routing. The Fig. 1 indicates a P2P topology graph which contains three routing paths.

In the Fig. 1, the nodes 1, 2 and 4 are the initiator of the communication while the nodes 18, 19 and 22 are the receivers. Generally, these nodes have to go through several relay nodes to complete the communicating task. The communicating path the arrow in the figure points to is known as routing path. The routing path will be undermined if there one or more nodes in one routing path fail or quit. If this happened, it is needed to recreate a new path, which called the routing resetting.

According to characteristic of the routing topology described above, we can abstract a P2P routing path R into the form:

$$R = < P_s, P_1, P_2, P_3, \ldots\ldots, P_M, P_d > \tag{1}$$

The P_s in formula 1 is known as the source node of communication, the P_d is target node of communication and the P_1, P_2, P_3,..., P_M are relaying nodes.

Quantitative analysis of attacks factors: From the analysis in above section, there are several main factors that relate to the effect of routing attacks. For quantitative analysis, we assume that the P2P network nodes and resources are evenly distributed, the node degree of each node is equal, the threshold value of routing reset times is λ, the total number of P2P network nodes is N, in which numbers of malicious nodes are M, the average routing path length of each communication channel is L. This section will get a theoretical analysis and get the minimum average times n in destroying a routing path. Since the malicious nodes is randomly attack any possible routing paths, only when the same communication channel C_{sd} is destroyed more than λ times, the channel is completely disable.

For simplicity, we assume that there is only one communicating process between every two nodes in P2P network, i.e., a routing path, so there are N (N - 1) /2 routing paths in N nodes. If these M malicious nodes are uniformly distributed in M routing path, they can destroy M routing path in one round attack. If these M malicious nodes are distributed in same routing path, they can only destroy one routing path in one round attack. The two cases above are two extreme cases, usually the number of the routing path that is destroyed by malicious nodes distribute in [1, M]. In addition, the longer the length of the routing path, the more the malicious nodes sneaked into the path, thus the possibility that the routing path is destroyed becomes greater. Overall, we assume the number of the destroyed routing path in one round attack is (1 + M) /2 ln L and this value is relatively compromised. It can be deduced that the possibility p that the malicious nodes damage the same routing path in round is (1 + M ln L) /N(N - 1).

Each attack that the malicious nodes launched is a random and independent event and all for one purpose, which fits the feature of independent distributed central limit theorem. We can use this theorem to quantitatively describe the attacking event. Assume X is the number of time of the same routing path that was damaged by n rounds of attacks by the malicious nodes, then from the central limit theorem there is $X \sim b$ (n, p). When the same routing path is damaged more than λ times before the end of the communication, the path is completely disable:

$$P\{X \geq \lambda\} = \sum_{k=\lambda}^{\infty} C_n^k p^k (1-p)^{n-k} \tag{2}$$

We believe that when $P\ \{X \geq \lambda\} \geq 0.99$ then the damage can be identified success, thus to anti-derivate the average minimum times n that to destroy a routing path.

The formula 2 is fairly complex and difficult to calculate, so we use De Moivre-Laplace theorem to get its approximate value, the transformed formula is shown as follows:

$$\begin{aligned} P\{X \geq \lambda\} &= P\left\{ \frac{X - np}{\sqrt{np(1-p)}} > \frac{\lambda - np}{\sqrt{np(1-p)}} \right\} \\ &\approx \int_{\frac{\lambda - np}{\sqrt{np(1-p)}}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} dt \\ &= 1 - \Phi\left(\frac{\lambda - np}{\sqrt{np(1-p)}} \right) \end{aligned} \tag{3}$$

The Φ in formula 3 is a standard normal distribution function, in the study, its value should be equal or lesser than (1 - 0.99):

$$\Phi\left(\frac{\lambda - np}{\sqrt{np(1-p)}}\right) \le 0.01 \tag{4}$$

We replace *p* in formula 4 with its true value and get below formula:

$$\Phi\left(\frac{\lambda - n\frac{(1+M)\ln L}{N(N-1)}}{\sqrt{n\frac{(1+M)\ln L}{N(N-1)}(1-\frac{(1+M)\ln L}{N(N-1)})}}\right) \le 0.01 \tag{5}$$

If assume Φ (Z) = 0.01, we can find Z = -2.33 from standard normal distribution table and get below formula:

$$\frac{\lambda - n\frac{(1+M)\ln L}{N(N-1)}}{\sqrt{n\frac{(1+M)\ln L}{N(N-1)}(1-\frac{(1+M)\ln L}{N(N-1)})}} = -2.33 \tag{6}$$

In the case that *N, L, M, λ* are all known, formula 6 become a quadratic equation of variable *n*, through which we can calculate the value of *n*:

$$n = \frac{2\lambda + 5.43(1-p) + \sqrt{29.47(1-p)^2 + 21.72\lambda(1-p)}}{2p}$$
$$p = \frac{(1+M)\ln L}{N(N-1)} \tag{7}$$

Using the formula 7, we can calculate the average minimum number of attack times *n* that to destroy a routing path and also easily analysis the effect that the number of malicious nodes, the average length of routing path to process of destroying a routing path.

Associate with formulas above, we can derive the minimum value of the attack time's *n* in multiply cases. Supposing there are 10^5 nodes in P2P network, *λ* is 6, according to the theory and the formula above, when the proportion of malicious nodes in P2P network is 5, 10, 25 and 35%, respectively, we approximatively calculate the relationship between the minimum attack times *n* and the length of routing path *L*. The result is shown as follow.

From Fig. 2, we can find that the length of routing path hasn't vital affect to the routing security, but the proportion of malicious nodes has. The discovery becomes the important foundation that we design SRABMEPD algorithm.

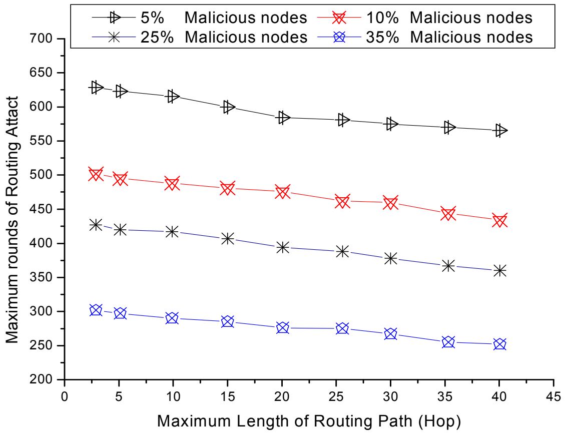

Fig. 2: Length of routing path and the proportion of malicious nodes to the affect of success rate of routing attack

THE DESIGN OF SRABMEPD ALGORITHM

Comprehensively analyzing the characteristics of P2P routing attacks, we consider that to exclude malicious nodes from P2P network is fundamental counter-measure to those attacks. So the emphasis of SRABMEPD algorithm lies in attribution detection of P2P nodes with special encryption message.

The attribution detection of P2P nodes based on communicating primitives: There are five primitives in P2P protocol: Ping, Pong, Query, Query Hit and Push. Primitive Ping owns the capacity that finds or activates P2P node and can be used to send detection information toward other nodes in routing path. Primitive Pong can fetch the response information from detected nodes. The attribution of the detected nodes can be analyzed by the information. Thus, making appropriate improvements to Primitive Ping and Pong can achieve the attribution detection of all the nodes.

Process of the initialization of routing path: When a node want to communicate with another node, there must be built a routing path between them first. The initial establishment of routing path does not adapt the active detecting method, because nodes need to consume certain system resources for detection, it will cost a lot to establish a routing path if uses the method of one by one detection and it will also cost longer time, which is no good for communicating. After the establishment of routing path, the work of attribution detection of relaying nodes is merged in the routine of topological maintenance of the P2P networks. The means is efficient but low cost and will not bring significant negative impact in P2P communicating. The specific way to build a routing path is described below.

The initial nodes of the communication send the request of connecting to some nodes in the routing table, when these nodes receive the request, according to the P2P handshake protocol; they forward this request to their neighbor nodes at some forwarding probability ξ. These neighbor nodes forward the request in the same way until it reaching the correct target node. The forwarding probability ξ has an inverse proportion with the load statue of the nodes. According to the method of establishing connection above, the last initiator nodes may receive several reachable routing paths to choose from. Since the overlong routing path will bring the negative impact to the communicating efficiency and the routing security, we prior choose the routing path with the least number of hops. The Small World Theory indicates that through average 6 persons, we can find the one that we want. Thus we limit the max length of the routing path be or less than 6. If there still have several routing paths are under the condition, we then choose the routing path whose total time delay is the least, that is $Min\left(\sum_{i=1,j=2}^{i=M-1,j=M} RTT_{ij}\right)$, the RTT_{ij} is the delay between two nodes.

When the routing path is chosen, the initial node of the communicating first need to record every relaying node's IP address, port number and P2P ID and then to consult with every relaying node to get a set of asymmetric keys and lastly to save the public keys from them for the future detection. To the malicious nodes, including the zombie nodes which were controlled by the virus, if each normal P2P network activities shows refuse or even destruction, they will be kicked out from the P2P network due to the early exposing. Thus those nodes perform their malicious activities on certain probability, like sometimes good but sometimes evil. Based on the discussion above, the pseudo-code of initially creating a routing path is as follows:

```
$P_s \xrightarrow{Build\ Connection} P_d$

Repeat following routing search course:

$P_s \xrightarrow{\xi} P_1, P_3, P_6.....$

$P_1 \xrightarrow{\xi} P_4, P_5, P_2.....$

$P_4 \xrightarrow{\xi} P_7, P_9, P_{10}.....$
……………..
Until arriving at $P_d$
Obtain multiple routing path between $P_s$ and $P_d$
Put those routing path into array R $\{r_1, r_2, …., r_n\}$
If routing length of $r_i$>6 then
    Kick $r_i$ out of R // i = 1, 2, 3….
End if
$R$ = Sort by $\left(\sum_{i=1,\ j=2}^{i=M-1,\ j=M} RTT_{ij}\right)$ with low to high
For $k$ = 1 to $n$ do
    Peers in $r_k$ send their <$ID_k$, $IP_k$, $Port_k$> to $P_s$
    $P_s$ consult Asymmetric Key Pair with every peer in
$r_k$ and occupy their Public Key
If above course is triumphantly executed then
    Adopt $r_k$ as the ultimate routing path
    Exit cycle
Else
    Discard $r_k$
      $k = k + 1$
    End if
End for
```

Detection technology based on multiple nested encryptions: When the routing path is established, before the end of the communicating, the initiator will use the Ping to send specific encrypted information to the receiver. Along with the Ping command, this information is forwarded to the target node upward node by node. Every time to pass a relaying node, part of the information will be modified and signed for recording the behavior of nodes. If a problem occurs to a relaying node, the relevant problem information will be send back through the Pong command to the initiator. Then the initiator will analyze the returned information and to determine the nature and location of the problem. The result of the analysis will determine whether to put the reported node into the malicious nodes' table and to determine neither to reset the routing path or retest.

To prevent the malicious node to tamper the information along with the Ping and Pong command, we adopt the Public-key nested encryption to encrypt the transmitting information to shut down on this malicious behavior. The specific method is as follows.

Assuming a routing path is R = <P_0, P_1, P_2, P_3,…, P_m>, P_0 is the initiator. The detection begins, the P_0 randomly generates a positive integer X and nested encrypt it with all the <*IP address*, *Public key*>, then to add the two Ping command to forward to the next hop P_1 node. The encrypting format of detecting information packet is as follows:

$$K_1\,(X, A_2, K_2, K_2\,(A_3, K_3, K_3\,(..),.., A_{m-1}, K_{m-1}, K_{m-1}\,(A_m, K_m))..)$$

This is an "onion" type of nested encryption data structure, in which the K_i is the number i relaying node's public key, the A_i is the number i relay node's IP address and $1<i<m$. The follow relay nodes have to use their own private key to peel the "onion" data layers by layers if they want to use it.

In order to protect the authenticity of the random number X, the P_0 need to sign the X by its private key S_0 and forward the result $S_0\,(X)$ to P_1. The node P_1 received that information, it decrypts the data packet by its own private key S_1, however, it can only get and use the data X, A_2, K_2, because the last half of the information was encrypted by the P_2 node's public key K_2. Thus the P_1 is not able to decrypt this information, so that even if the P_1 happens to be malicious node, it still cannot tamper the detecting information of the follow nodes.

According to the address A_2 of the node P_2, the node P_1 encrypts (X + 1) by K_2 and then forward it to P_2, at the same time, P_1 has to use its own private key S_1 to sign the (X + 1) and send the generated data $S_1(X + 1)$ together to P_2. The signature has important use on detecting the location of malicious nodes. The node P_1 has to deliver the decrypted information K_2 (A_3, K_3, K_3 (A_4, K_4, K_4 (…), …, A_{m-1}, K_{m-1}, K_{m-1} (A_m, K_m))…) to P_2. Then P_2 decrypts this segment of information by its private key S_2 and delivers K_3 (A_4, K_4, K_4 (…), …, $A_{m\text{-}1}$, $K_{m\text{-}1}$, K_{m-1} (A_m, K_m))…) to P_3. At the same time, P_2 has to encrypt (X + 2) by public key K_3 and sign (X + 2, S_1(X + 1)) by private key S_2, then forward those results to P_3. To repeat the operation given above until arrive at the target node P_{m}. If the routing path is expedite, then the X will turn into X + m when the detecting information arrives. Thus the integer m actually represents the number of routing pops. In addition, the signature information will also turn into the format below:

$$S_{m-1}(X+m-1, S_{m-2}(X+m-2, ... S_1(X+1, S_0(X))...))$$

After receiving the signature information ensuring its correctness, the node P_m will use its private key S_m to sign the information for the last time, the format is shown as follows:

$$S_m(X+m, S_{m-1}(X+m-1, ... S_1(X+1, S_0(X))...))$$

Last, the node P_m will attach the signature information described above to the Pong command and return it to the initial node P_0 through the same routing path. The P_0 uses the corresponding public key to decrypt the signature information and to get its data (X + m). If the value of m is the same with the value of its saved length of routing path, we consider there is no malicious node on the routing path.

If a certain node P_{i+1} did not return the feedback information in three times detection, the node will be regard as a malicious or disabled node.

If the initiator P_0 receives wrong feedback detection information, there may be two cases, one is that the feedback information damaged during the transmission process due to some random and irresistible faults. However, the case is also a very small probability event and has little affection to our routing detection system. Another case is that the malicious nodes in the routing path tampered the data. For example, when a malicious node produce a wrong value: (X + i)′, then transmits it to next hop node, which will cause the following nodes in routing path all fail in calculating the i. Since the parameter i represent the routing hop and the routing order, the P_0 need to decrypt all the i from all the S_i (X + i) which were send by every relaying node. In normal situation, the sequence of the value of i is a string of continuous integer beginning at 1. If the value of i is discontinuous, the first break point is malicious node. What we need to do is just to avoid the node when resetting the routing.

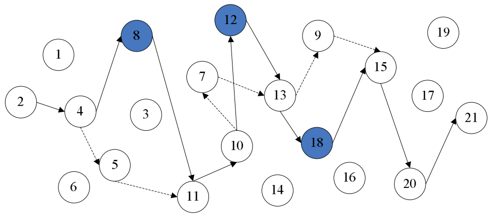

Fig. 3: Reset routing path to avoid malicious nodes

The below figure describe the process that the SRABMEPD algorithm detect malicious node and reset routing path in P2P network.

In Fig. 3, the node P_2 is the communicating initiator, node P_{21} is the receiver, the initial routing path is R = <P_2, P_4, P_8, P_{11}, P_{10}, P_{12}, P_{13}, P_{18}, P_{15}, P_{20}, P_{21}>, the length of path is 10. After using SRABMEPD algorithm to detect all the relaying nodes in the routing path, the system finds that the member P_8, P_{12} and P_{18} in the routing path are bad members, then the initiator resets the routing path and creates a new path R' = <P_2, P_4, P_5, P_{11}, P_{10}, P_7, P_{13}, P_9, P_{15}, P_{20}, P_{21}>. It reveals that the new routing path has avoided the bad members, which make the attacking behavior by malicious nodes become more difficult thereby enhancing the security of routing.

SIMULATION EXPERIMENTS AND ANALYSIS

The simulation experiments were performed in the PC that the CPU is P4 2.6 GHz and the memory is 2 G and the OS is Fedora Linux 8.0 and the simulating software is P2P sim 3.5.

To compare the effect of developed SRABMEPD algorithm, the experiments adopted Chord and Koorde algorithm as the reference. The Chord is the most common used ring network structure while the Koorde is the famous P2P network protocol based on the graph theory. These two as the reference can well reflect the effect after improving. These three algorithms all simulated 10^4 nodes and the experiment were done for three times. The purpose of the first time of the experiment is to study the effect that the SRABMEPD algorithm to detect the malicious nodes in P2P networks. The experiment set up the proportion of

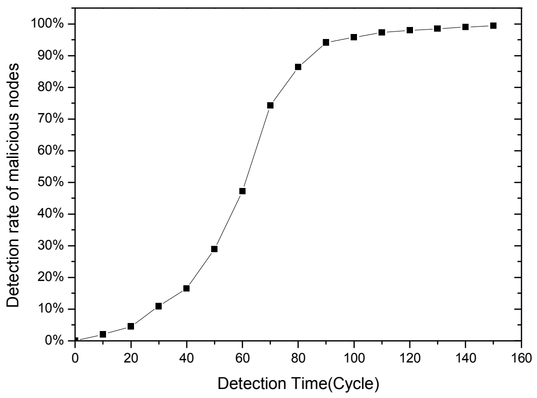

Fig. 4: The malicious nodes' detection rate of SRABMEPD

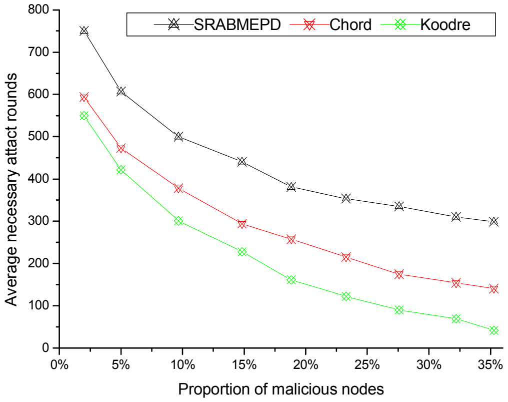

Fig. 5: The affect of malicious-node proportion to routing-attack rounds

malicious nodes as 30% and distributed evenly. The purpose of the second experiment, while the proportion of malicious nodes in P2P system gradually increases, is to study the change of the needed average attacking number of time to successfully destroy a routing path. And the purpose of the third experiment is to study how the length of routing path effects the number of average attacking time, while one of the experimental conditions are the proportion of malicious nodes in the P2P network is fixed at 20%.

It can be seen from the Fig. 4 that the detecting speed of SRABMEPD algorithm towards malicious nodes is relatively satisfied, the successful detecting rate after the P2P system running for 20 min significantly boosted. After 80 min past, the system has detected over 90% of malicious nodes and after 110 min, almost all the malicious nodes have been detected. Thus, the effect of SRABMEPD detection is significantly nice, which substantially increased the routing security in P2P networks.

It can be seen from the Fig. 5, with the proportion of malicious nodes in the P2P network increases, the average attacking number of time on successfully destroying a routing path declines in all the three algorithms, especially to the Koorde, coming after the Chord. The decrease rate is relatively slow to the SRABMEPD and when the proportion of malicious nodes have reached 23%, the rate of decline leveled off, which showed the excellent defense to the attack. To consider on the other side, with the proportion of malicious nodes declined, malicious nodes in the SRABMEPD had to attack much more times, whose number of attacking time was much larger than the other two algorithm, which has increased the attack difficulty, so that to enhance the security of P2P system.

CONCLUSION

This study, in connection with the P2P routing attacking issue, proposed SRABMEPD algorithm which is able to detect the malicious members and instable members in the routing path within a very short period and to avoid this member by resetting routing path, thereby enhancing the security performance of the P2P network. In addition, in the premise of maintaining the anti-attacking ability, this algorithm can limit the maximum length of the routing path, which will reduce the communicating delay and optimize the communicating performance.

ACKNOWLEDGMENT

It is a pleasure to acknowledge the contribution of Honghua Wu and Zhiyuan Lu, both of whom participated in a number of fruitful discussions. The study is sponsored by Key Project of Education Committee of Hubei Province, No. D20114401.

REFERENCES

Anjali, G., L. Barbara and R. Rodrigo, 2004. Efficient routing for peer-to-peer overlays. Proceedings of NSDI, pp: 23-30.

Atul, S., 2004. Defending against eclipse attacks on overlay networks. Proceeding of ACM SIGOPS'04, pp: 214-221.

BongSoo, R., K. Hoon and H. SungJe, 2008. The exclusion of malicious routing peers in structured P2P systems. Proceeding of AP2PC, LNCS4461, pp: 1254-1265.

Cheng, A. and E. Friedman, 2006. Sybil proof reputation mechanisms. Proceeding of ACM SIGCOMM Workshop on Economics of p2p Systems, pp: 129-136.

Dagon, D. and C. Zou, 2006. Modeling botnet propagation using time zones. Proceeding of Network and Distributed System Security Symposium, pp: 36-43.

David, B. and S. Dawn, 2006. Towards attack-agnostic defenses. Proceedings of the 1st USENIX Workshop on Hot Topics in Security, pp: 98-105.

Emil, S. and R. Morris, 2002. Security considerations for peer-to-peer distributed hash tables. Proceedings of Peer-to-Peer Systems, pp: 78-85.

Hongfei, S., 2003. An analysis of forwarding mechanism in crowds. Proceeding of ICC'03, pp: 142-151.

Kannan, J. and K. Lakshminarayanan, 2003. Implications of Peer-to-Peer Networks on Worm Attacks and Defenses. CS294-4 Project, Berkeley California.

Keith, N. and K. Minseok, 2009. Secure routing in peer-to-peer distributed hash tables. Proceeding of the 2009 ACM Symposium on Applied Computing, pp: 56-62.

Roger, W., 2005. Attacks on Peer-to-Peer Networks. Press of Swiss Federal Institute of Technology, Zurich.

Staniford, S., V. Paxson and N. Weaver, 2002. How to own the internet in your spare time. Proceeding of the 11th VSENZX Security Symposium, pp: 12-20.

Stefan, K. and P. Udo, 2009. Secure routing approach for unstructured P2P. Proceeding of ESIST, pp: 75-81.

Wei, Y., 2004. Analyze the worm-based attack in large scale P2P networks. Proceeding of 8th International Symposium on High Assurance Systems Engineering, pp: 42-50.

Yu, H. and J. Rexford, 2005. A distributed reputation approach to cooperative internet routing protection. Proceeding of Workshop on Secure Network Protocols, pp: 45-52.

Zhou, L., L. Zhang and F. McSherry, 2006. A first look at P2P worms: Threats and defenses. Proceedings of IPTPS, pp: 314-320.