A Multidimensional Mapping Mechanism Based Secure Routing Method for DHT

Zhixin Sun[1,2], Kai Bu[1], and Ke Ding[1]

[1] Nanjing University of Posts and Telecommunications, College of Computer
Xin Mofan Road. 66, 210003 Nanjing, P.R. China
[2] State Key Laboratory for Novel Software Technology,
Nanjing University, 210093 Nanjing, China
sunzx@njupt.edu.cn, bukai521@yahoo.com

Abstract. For improving the routing security of traditional DHT, in this paper, a Multidimensional Mapping Mechanism and a secure routing method based on it are proposed against various routing attacks. The proposed mechanism, which maps the resource search and related peers to a smaller space following the same topology with current DHT protocol to simplify the routing operation and decrease the coupling degree between security mechanisms and routing geometry, lays a solid foundation for applying to diversified DHT protocols. The subsequently proposed security measures based on Multidimensional Mapping Mechanism for DHT routing still keeps independent of certain DHT protocol. It pervades throughout the whole routing process, and peers could correct the malicious routing under its security rules. The theoretical analysis and simulation experiment result show that the secure routing method proposed in this paper can improve the average success ratio of lookup through effectively inhibiting malicious behavior.

Keywords: Multidimensional Mapping Mechanism, DHT (Distributed Hash Table), Routing Attack, Secure Routing.

1 Introduction

DHT (Distributed Hash Table) [5] is the key technology of research and realization of structured P2P (Peer-to-Peer) network, which employs distributed hash algorithm of DHT protocol to solve the structured distributed storage problems. In P2P network over DHT based topology, each peer and resource have a unique identifier which generated by hash computing, the most obvious characteristic is that peer does not need to record all other peers' information, but only maintains a relatively smaller routing table including certain peers. Then lookup message is effectively routed based on this simple routing table and related storing rule of resource's distribution information and this can abolish flooding algorithm in unstructured P2Pnetwork and improve the routing efficiency and lower the routing overhead. Due to its distribution, self-organizing and high scalability, designing routing algorithm with high efficiency for DHT is becoming a hotspot in international research on structured P2P. The representative DHT protocols prototype include the MIT's

P. Mueller, J.-N. Cao, and C.-L. Wang (Eds.): Infoscale 2009, LNICST 18, pp. 192–205, 2009.

(Massachusetts Institute of Technology) Chord [8], Microsoft Research Institute's Pastry [1], as well as UC Berkeley's CAN [16] and Tapestry [2] etc.

However, an important prerequisite to running DHT mechanism effectively is that peers are fully trusted in P2P system, which means peer security is highly required. But compared with the traditional C/S network, P2P network drastically increase the probability of peers implementing malicious behavior due to its characteristics of self-governing and dynamic, which make it hard to assure the security of DHT routing algorithm. [3,6] analyzed the routing algorithms of typical DHT and summed up potential malicious behaviors of peers. Malicious peer may forward lookup message incorrectly or broadcast incorrect routing updating information which may prevent the attacked peers from using P2P service normally. Also, following the regular routing algorithm, malicious peer may deny that it does store the searched data or declare its storage state but refuse to provide them. Besides, inconsistent behavior performed by malicious peer usually makes it harder to identify and preclude the malicious one when choosing appropriate peer to forward lookup message to.

For improving the routing security of traditional DHT protocols and overcoming the disadvantages of current security mechanisms, this paper proposes a Multidimensional Mapping Mechanism and a secure routing method for DHT based on it. The Multidimensional Mapping Mechanism maps the resource search and related peers to a smaller space topologized under the same routing geometry with current DHT protocol to simplify the routing operation and decrease the coupling degree between security mechanism and topology, and this lays a solid foundation for applying to diversified DHT protocols.

The rest of this paper is organized as follows. Section 2 gives an overview and analysis of currently related research work. Section 3 presents the Multidimensional Mapping Mechanism and a series security measures for DHT routing algorithm based on it. Section 4 analyzes the theoretical performance of the security method proposed in this paper and reports on the simulation experiment. Finally, conclusion with a brief discussion is provided in Section 5.

2 Related Work

Currently, the research methods of secure DHT routing mainly focus on the following directions.

2.1 Improving the Routing Algorithm

Castro et al suggested restoring optimized routing table for fast routing and constrained table for secure routing for each peer [10]. While executing the data search, the peer forwards the lookup message according to the optimized routing table and verifies the lookup result through collaborating with some neighbors. However, the overhead introduced by communication with neighbor peers in each verification process, recursive query according to the constrained table after verification failed, and broadcast operation followed by identifying unreliable peers during queries, will effect the method's performance to some extend.

Paper [9] added an additional table storing information of some neighbors to each peer and introduced a wide path lookup mechanism. Several routing paths are

constructed according to neighbors' information for lookup message. Only when peers at the same level fail in all paths will cause the resource search failure. It makes the structured P2P system more robust to malicious routing attacks. But this method require a $O(\log^2 N)$ connections for each peer and increase the complexity of routing operation to $\Omega(\log^2 N)$.

2.2 Trust Model

Papers [14,17] concluded and analyzed typical trust models based on P2P environment, and pointed out that the recommendation based trust relationship from social society is a reasonable research method. The global trust model is just based on this idea; it calculates peer's global trust value through iterative of satisfactory value amongst peers.

Literature [13] distributes a unique global reputation value according to the upload logs of document, when peer i wants to know peer k's global reputation value, it firstly gets the reputation information which have traded with peer k, then generalizes peer k's reputation from its trade partner's reputation value. However, this proposed model does not provide punishment mechanism to peers which cause service failure. The reputation value of malicious peers will not decrease immediately and this will result that the model can not reflect peers reputation value in that period of time. In addition, the iterative messages in the whole network caused by each trade process will also affect the performance of the network.

2.3 Employing Certificate Authority

Myrmic model [12] introduced an on-line CA (Certificate Authority) named NA (Neighborhood Authority) based on the off-line CA. NA participates in the structured P2P network management by issuing Neighborhood Certificate to certain peers when peers join or leave. Then peer can verify the lookup result through querying neighborhood certificate copies stored on neighbors of the peer which returns the result. This verification costs $2l$ (the number of neighborhood certificate copies) communication messages. During NA failure, newly coming peers can search resource via the peers which have neighborhood certificate but can not join the network until any on-line NA available. This rule further delays the overall routing table update, and the introduction of NA itself is not a reasonable choice due to against the essential characteristics of P2P network, namely distributed or non-centralized, to some extend.

Besides, current security mechanism usually aims at specific DHT protocol, which leads to a limitation of generality and scalability when applying to multiform DHT routing geometries or new-style P2P applications. Report mechanism introduced by Trust Model may also be used by malicious peer to slander other peers by deliberately sending malicious reports. And the inconsistent routing table update amongst peers will also cause a mistake to judge a legitimate peer as malicious one.

Therefore, it is important to find a more reasonable research method with higher generality and scalability but less influence to routing performance of the traditional DHT protocol compared with current secure mechanisms. In fact, if the ideal state that each peer can acquire information of all other peers within P2Pnetwork could be

achieved, then all malicious routing attack could be detected and the routing security issues figured out. But this can hardly be realized for structured P2P network including numerous peers with frequent join and leave. In this paper, we propose a Multidimensional Mapping Mechanism to map routing operation of lookup message and related peers into a smaller space aiming at relatively strengthening information sharing amongst peers and simplifying implementation of secure routing measures.

3 Multidimensional Mapping Mechanism Based Secure DHT Routing Method

In this section, we firstly enumerate and analyze several typical routing attack behaviors of malicious peers and then introduce the Multidimensional Mapping Mechanism and secure DHT routing method based on it.

3.1 Analysis of Typical Routing Attacks

In order to meet the demand of scalability, maintainability and the resource discovery algorithm's efficiency for structured P2P, DHT protocol stipulates that each peer can only store a small number of network peers' information, which makes peers vulnerable to malicious routing attacks due to lack of enough information of other peers. Fig. 1 demonstrates several possible routing attacks malicious peer may commit. Peer *Q* sends lookup message to search resource *k* with identifier *key* of which the distribution information is stored on peer *B*. Peer *A* is peer *B*'s predecessor in the correct routing path, when receiving the lookup message *A* forwarded, if peer *B* is malicious it may ① deny that it stores the information related to *k* or discard this inquery message, and return lookup failure to *Q*; ② deny that it stores the information related to *k* and forward the lookup message to certain next hop peer *C*, then *C* or other sub sequent peers returns the failed message. But if *A* itself is a malicious peer, it might

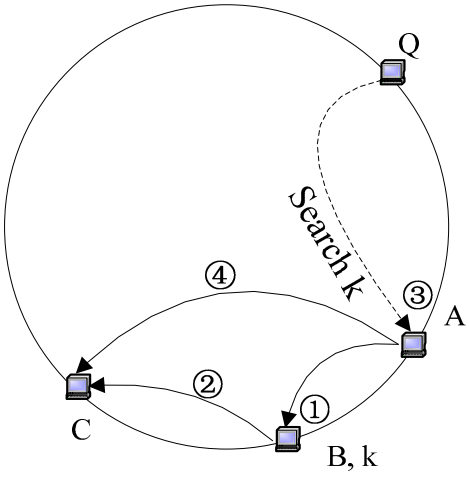

Fig. 1. DHT routing attacks examples

③ discard the lookup message and return a failure result; ④ wrongly route the lookup message to peers which are not supposed target peers, and the peers that receive messaged forwarded by A would return search failed message to Q.

Through analyzing the possible routing attacks mentioned above, we find that one of the fundamental reasons is that each node can only acquire a few peers' information which makes peers not transparency to each other. If peers Q, C, A can obtain the information of keys that are stored on peer B, then B can not successfully commit malicious behaviors like ①, ②; if peers Q, C can obtain the information of keys stored on A, the routing attacks like ③ or ④ can not be implemented. Next we will introduce the Multidimensional Mapping Mechanism and how it work against DHT routing attacks through strengthening information sharing amongst peers.

3.2 Overview of Multidimensional Mapping Mechanism

Rule 1. Multidimensional Mapping Mechanism: When requirement of space conversion met, peers related to the remainder routing operation with identifiers distribute in $p_{1i} \in \{p_{1a}, p_{1(a+1)}, \ldots, p_{1b}\}$ in current space S_1 (peer identifier $p_{1i} \in \{p_{10}, p_{11}, \ldots, p_{1(n-1)}\}$, where n represents the maximum number of peers S_1 can hold) will be mapped into a smaller space S_2 with identifiers $p_{2i} \in \{p_{20}, p_{21}, \ldots, p_{2(m-1)}\}$. The maximum number of peers of S_2 can be expressed as $m = 2\exp|\log_2(b\text{-}a+1)|$.

Rule 1 depicts the principle of Multidimensional Mapping Mechanism. As shown in Fig. 2, peer $Q1$ in current space S_1 sends a lookup message which is forwarded to peer $P1$. $P1$ starts space conversion operation as required condition met and maps itself and other peer related to remainder routing operation into a smaller space S_2 re-topologizing under the same geometry with S_1. When $P3$ receiving the forwarded lookup message, space conversion requirement is met again and routing operation is going on after being mapped into the new space S_3. Fig. 2 explicitly demonstrates that

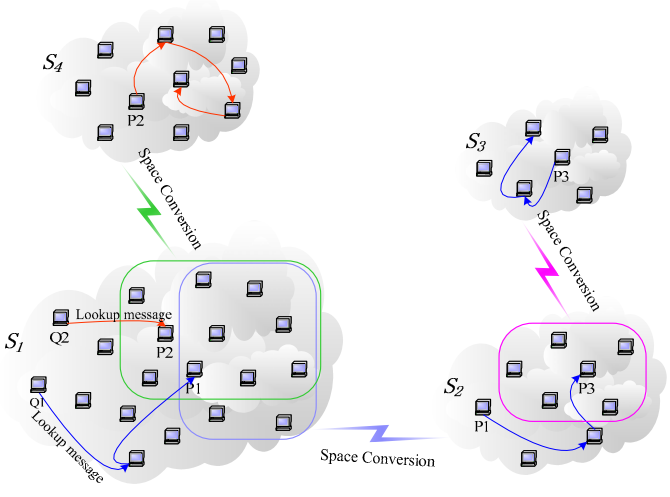

Fig. 2. Demonstration of Multidimensional Mapping Mechanism

each success implementation of space conversion can simplify the security assurance through continuing the remainder routing operation in a smaller space. S_4 represents the space converted by peer $P2$ when space conversion condition is met during the routing process of another lookup message started by peer $Q2$.

As to the requirement that space conversion needed, mapping method, and maintenance of routing information in new space, we will clarify them detailedly along the following sections.

3.3 Multidimensional Space Division

In structured P2P network, each peer (or resource) is assigned an unique identifier which is an integer generated by hash calculation. All identifiers distribute within the same value range [11]. For most DHT protocols, distribution information of resource whose identifier is *key* is stored on peer of which the identifier is equal to or larger than but nearest to *key*. Successful resource search comes after that lookup message is correctly routed to related peer and distribution information of searched *key* is returned [4]. In this paper, we realize Multidimensional Mapping Mechanism through identifier division (rule 2) and mapping multidimensional space according to divided identifier groups (rule 3).

Rule 2. Identifiers Division: Let n be the identifier length, divide it into m groups, and each group's length p_i is constrained by expression $\sum_{i=1}^{m} p_i = n$. ($1 \le i \le m$).

Rule 3. Multidimensional Space Mapping: The ith dimension space corresponds to identifier groups $[p_i, p_{i+1}, p_{i+2},...,p_m]$, and the number of peers it includes is $s_i \le 2^{n-\sum_{j=1}^{i-1} p_j}$ ($1 \le j \le$i-1, $1 \le i \le m$), or $s_i \le 2^{\sum_{j=i}^{m} p_j}$ ($1 \le i \le j \le m$).

Rule 2, rule 3 can be explained in company with Fig. 3. In order to map peers to a smaller field, thus to simplify the DHT security mechanism, we divide the identifiers into certain groups. Let $p_i$bits denote the length of identifier group I ($i \in [1,m]$, where m is the total number of divided groups). Then aiming to the target of reducing the space that routing operation works in, we assign certain number of identifier groups to one space. The ith dimension space related identifier length is $n - \sum_{j=1}^{i-1} p_j$ ($1 \le j \le i-1, 1 \le i \le m$) or $\sum_{j=i}^{m} p_j$ ($1 \le i \le j \le m$).

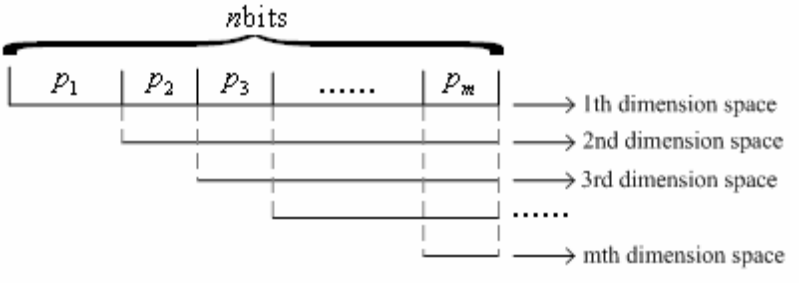

Fig. 3. Space mapping through identifier division

Thus the domain of the ith dimension space's identifier is (*currentID, currentID*+ $2^{n-\sum_{j=1}^{i-1} p_j}$] ($1 \le j \le i$-1, $1 \le i \le m$) or (*currentID, currentID*+ $2^{\sum_{j=i}^{m} p_j}$] ($1 \le i \le j \le m$), here *currentID* is identifier of the peer which is responsible for starting space mapping). In addition, the 1st (i=1) dimension space corresponds to the complete identifier, that is, the initial P2P topological space. The mth dimension space is the final dimensional space. When m=1, it means that the identifier have not been divided, and the related topology is just equal to traditional DHT.

3.4 Quick Location and Initial Security Check

In order to control the routing overhead introduced by Multidimensional Mapping Mechanism, the 1st dimension space still uses traditional DHT routing mechanism. For instance, if it uses Chord's routing algorithm [8], routing table of the 1st dimension space will divide the one-dimensional circle into n parts. Here n is equal to the length of identifier and should meet the constraint $n=\lceil \log N \rceil$ with network scale N. The ith field corresponds to the identifier [(*nodeID*+2^{i-1})/mod2^n, (*nodeID*+2^i)/mod2^n) ($1 \le i \le n$), and chooses the first online peer in this field as the next hop peer of related table entry. Distribution information of sharing resource with identifier *key* is stored by the first peer which meets *nodeID*≥*key*.

On this basis, we propose initial security check measure to avoid security issues in traditional DHT routing process. Peers use the initial security check measure to verify the lookup message they received and consider it as the evidence to judge whether the previous hop peer is malicious or not. Below is detailed procedures for algorithm 1 which define principle of initial security check measure.

1) If peer stores information corresponding to *key* in lookup message, then search successes and send *key* related information to the querier peer. Otherwise, go to 2);
2) If *nodeID* is smaller than *key*, then the peer forwards lookup message. Otherwise, go to 3) If *nodeID* is smaller than *key*, then the peer forwards lookup message. Otherwise, go to 3);
3) If *nodeID* is larger or equal to *key*, traditional DHT will consider it as lookup failure, but this consideration is usually not reasonable. For instance, if there already exists peers responsible for information related to *key* in a peer and its previous hop peer, but the previous hop peer's routing table did not update in time, this is why we do not directly judge it as failure when *nodeID* is equal to *key* and peer does not store *key* related information. We make a rule in such situation, peers search whether peer exists between *key* and *nodeID* according to its routing table, if exists, go to 4); otherwise, go to 5);
4) Peer searches the largest identifier between *key* and *nodeID* according to its routing table, then routes the lookup message to this identifier related peer;
5) Run space conversion mechanism, and continue to search in the mapped space.

Algorithm 1: Algorithm for initial security check measure

```
1: if (store_info(key) is TRUE)
2:   info = get_info(key);
3:   send_info(info,quefierID);
4:   return success;
5: else
6:   if (nodeID<key)
7:     forward(key); //continue routing the lookup mesage
8:   else
9:     if (has_peer_in(nodeID,routing_table) is TRUE)
10:      pre_peer = the peer with the largest peer identifier between key and
                    nodeID;
11:      reroute(key,pre_peer);
12:    else
13:      startup space conversion mechanism;
```

3.5 Space Conversion Mechanism

On the basis of multidimensional mapping mechanism and initial security check measure, space conversion mechanism is introduced to judge on what condition the multidimensional mapping mechanism will be called (Rule 4) to proceed the remainder routing operation in a relatively smaller field (Rule 5).

Rule 4. Space Conversion Condition: When $|nodeIDI - key| \le \delta$, the inquery information will be routed to higher dimensional space to continue resource lookup.

Here δ, calculated by formula δ=2exp(sum(p_{i+1}, p_{i+2}, p_{i+3},…p_m)), is the threshold which determines whether to run space conversion mechanism. It is defined by current dimension i and the length of identifier related to space with neighboring dimension. According to rule 4, if the difference between two identifiers of current peer and searched resource falls into the range of current space and the neighboring higher dimension space, space conversion will be start by current peer. The following rule 5 and rule 6 give the calculation conditions about the dimension of newly converted space.

Rule 5. The dimension i of the new mapped space should meet

$$2^{\sum_{j+1}^{m} p_j} \le |\, currentID - key \,| \le 2^{\sum_{j=i}^{m} p_j}.$$

Rule 6. $$currentID = \begin{cases} nodeID. & nodeID \le key \\ nodeID'. & nodeID > key \end{cases}$$

The *currentID* is identifier of the peer which starts space convert mechanism when rule 4 works. Otherwise, the first peer with smaller identifier (*nodeID*') but nearest to peer *key* recorded in routing table will be delegated to run space conversion mechanism.

Algorithm briefly generalizes the procedures of space conversion method.

Algorithm 2: Algorithm for space conversion mechanism

```
1: if (|nodeID-key|<=δ)
2:  if (nodeID<=key)
3:    currentID=nodeID;
4:    convert_space(currentID,i);
```

```
5:  else
6:    reroute the lookup message with key to the largest nodeID' that is less or
          equal than key;
7:    currentID=nodeID';
8:    convert_space(currentID,i);
 //convert_space(currentID,i) converts the space to the ith
 //dimension by peer with nodeID equal to currentID
9: convert_space(currentID,i)
10: |currentID-key| should be in the nodeID range of i+1th and ith space;
```

3.6 Routing in Final Dimension Space

During the space conversion process, if the dimension of the new space is *m* calculated under rule 5, the lookup message would be transferred to the final dimension space. According to space mapping mechanism, the final dimension space has the smallest area scale compared with space of other dimensions. Recurring to the neighbor set maintains by each peer, we rule that all the peers which are mapped to the area share routing information among each other. Then one step routing mechanism and final security check mechanism are proposed based on it, we will explain the two mechanisms along with the expatiation of the routing procedures in final dimension space.

Algorithm 3: Algorithm for routing operation in final dimension space

```
1: direct_routing(currentID, init_resultID)
2: if ( share(key, nodeIDs) is TURE)
3:  if (nodeIDs ==1)
4:    init_resultID=the nodeID of the peer found by share();
5:  else
6:    init_resultID=the nodeID of the peer with the updated info.;
          //final security check mechanism by init_resultID
7: if ( store_info(key) is TRUE)
8:  resulted=init_resultID;
9:  send_info(info, querierID);
10:  return success;
11: else
12:  resulted=the nodeID of the peer with nodeID that is larger than and nearest
                to key;
13:  if ( store_info(key) on resulted is TRUE)
14:    send_info(info, querierID);
15:    return success;
16:  else
17:    send_fail(key, querierID);
18:    return false;
```

Algorithm 3 illustrates the main procedures of routing operation in final dimension space. Peer with identifier *currentID* is responsible for mapping remainder routing operation into the final dimension space. Firstly, peer *currentID* checks that if there is

peer storing information inquired by lookup message according to the share information of other peers and routes the lookup message to the most appropriate peer (with identifier *nodeID*) if any. We called this procedure as one step routing mechanism.

As the routing table update will take a certain period of time, there may exists more than one suitable peer that stores the distribution information of searched resource with identifier *key*. So we do not return this peer (with identifier *init-resultID*) as the final lookup result (with identifier *resultID*), but employ it to run the final security check mechanism. If peer *init-resultID* does exactly store the searched resource information, lookup succeeds and *init-resultID* returns the information of *key* to querier (with identifier *requerierID*); else, peer *init-resultID* runs one step routing operation again and choose another appropriate peer as *resultID*. The final result of lookup message will be returned by peer *resultID* according to whether it stores *key* related information or not.

4 Performance Evaluation

To satisfy routing information requested by the multidimensional mapping mechanism proposed in this paper, peer mapped into new space needs to get some additional information except the current routing table. Suppose that all the other dimension spaces, except the final dimension space, follow the same routing table structures with current DHT protocol (take Chord for example here), then after comparing with table entries [5,7,8] and according to multidimensional space mapping (rule3), it can be concluded that the first dimension space is just corresponding with the entire previous structured P2P network, and the routing table needs no extra information. The maximum number of peers in space with dimension from 2 to m-1 is decreasing gradually, so as the entries of routing table. And the routing table of the i+1th dimension space just partially tallies with the ith dimension space's. Besides, the initial security check measure guarantees that only when difference between *nodeID* and *key* is relatively small the space conversion mechanism is called, i.e., the times of space conversion operation will be relatively few and this also further limits costs of routing table update introduced by secure routing method proposed in this paper.

As to the routing overhead, 1) if no malicious exists in the whole routing path, the routing path will be same with traditional DHT before falling into the final dimension space. According to the one step routing mechanism, the routing operation for lookup message only takes one step. But to the traditional DHT routing algorithm, it will need at least one routing operation when most related peers are online. Thus, routing overhead of secure routing method by this paper will not exceed the traditional DHT ($O(\log N)$); 2) if malicious does commit routing attacks during routing operation, lookup message will be routed to the relatively closest peer to *key* according to initial security check mechanism. This will effectively control the increase of routing overhead before mapping routing operation into the final dimension space not more than m-2 (here m represents the highest dimension of rule 2). Thus, routing overhead for this security method is no more than $O(\log N+m-2)$. Let p be the probability that a peer shows malicious behavior when routing lookup message, then the maximum routing cost of security method proposed in this paper is $(1-p)O(\log N)+pO(\log N+m-2)$, which also could be expressed as $O(\log N)+p[O(\log N+m-2)-O(\log N)]$. This value will gradually approach to $O(\log N)$ as network scale increases.

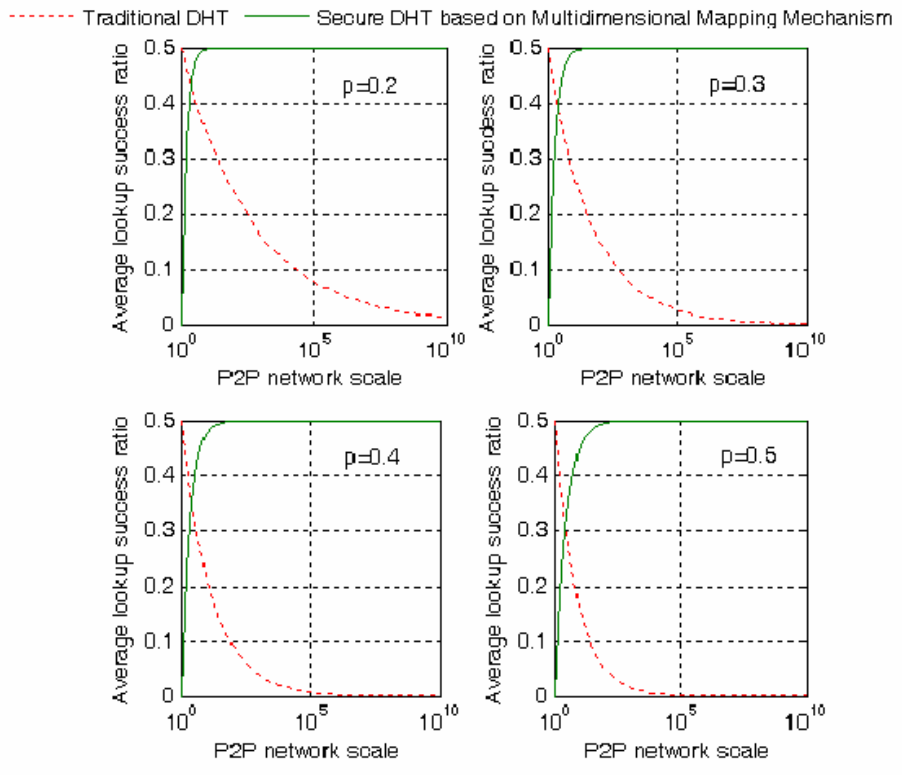

Fig. 4. Comparison of average lookup success ratio under various network environments

The probability p we introduced above ranges from 0 to 1. If taking $0.5\log N$ as the average routing hops for traditional DHT protocol [12], and supposing that resource *key* being searched has equal probability of really exists in current network or not, then average lookup success ratio of traditional DHT will be $F_1(p)=0.5*(1-p)^{0.5\log N}$, which means that only when the searched resource exists and no malicious peer commits routing attack along whole routing path will the lookup be success. For secure routing mechanism proposed in this paper, the average lookup success ratio turns to be $F_2(p)=0.5*(1-p^{0.5\log N+\lambda})> F_3(p)=0.5*(1-p^{0.5\log N})$, i.e., if resource required by lookup message does exactly exist, our proposed security method will fail the resource lookup only when all the peers involved in routing path commit routing attack (here λ represents routing costs introduced by space conversion). Fig. 4 illustrates comparison of average lookup success ratio when p=0.2, 0.3, 0.4, 0.5, respectively. With increasing of the network scale, DHT hiring secure routing mechanism proposed in this paper can achieve relatively higher success ratio comparing with traditional DHT.

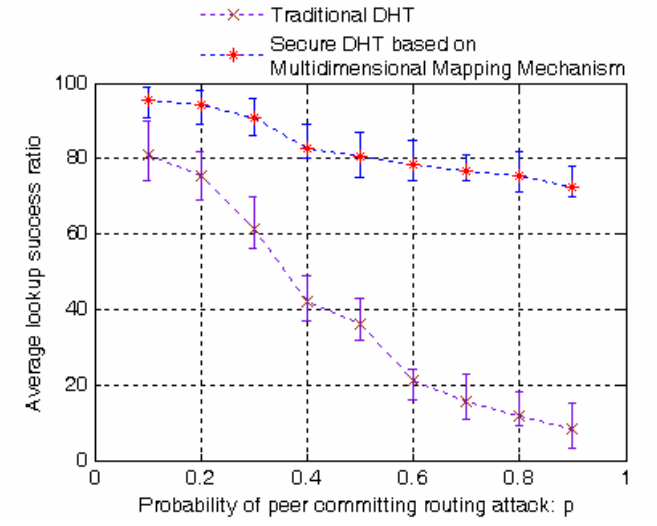

Fig. 5. Simulation results of average lookup success ratio

We also verify the analysis above through experiments in a simulated P2P network including 1000 online peers with 10bits (2^{10}=1024>1000) identifiers. We set 3bits for the identifiers group corresponding to final dimension space. For each routing experiment, a random *key*∈[0, 1023] is firstly generated and store its information on right peer according to the routing algorithm, and then randomly generate a peer identifier *lookup_nodeID* which starts the lookup message to search resource *key*. Lookup success message will be returned if no malicious routing behavior took place during whole routing process. For experiments of traditional DHT, we suppose that probability of peer committing malicious routing behavior directly proportional to lookup failure caused by this. To compare traditional DHT and secure DHT based on multidimensional mapping mechanism, we repeat 10 groups experiments under each $p \in$ [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9] and 100 times for each group for two kinds of DHT protocols. Fig. 5 shows the statistical result of traditional DHT and secure DHT. Lookup success ratio of both kinds of DHT protocol will decrease as probability p increases, but compared with traditional DHT, multidimensional mapping mechanism based secure routing method proposed in this paper can keep strong defense ability to malicious routing attacks and relative high average lookup success ratio as well.

5 Conclusion

In this paper, we propose a novel multidimensional mapping mechanism and secure DHT routing measures based on it against the routing security issues of structured P2P and limitations of current research methods. Multidimensional Mapping Mechanism simplifies the design and implementation through mapping routing operation into a smaller space topologized under the same geometry with current DHT topology. And

this also guarantees the nice generality of our proposed method for applying to multiple types of DHT protocols. The results of simulation experiments show that after employing the multidimensional mapping mechanism based secure routing method, DHT can achieve a higher average lookup success ratio with limited cost of routing overhead.

For future work, more effort will be made to improve routing information update in newly mapped space to better control the routing overhead introduced by space conversion mechanism. Besides, final security check mechanism and determination of the lookup result in final dimension space also need further investigation.

Acknowledgments

This research work acknowledges the supports by the National Natural Science Foundation of China (60973140), the National Natural Science Foundation of Jiangsu Province of China (BK2009425), the Innovation Fund for Technology Based Firms (08C26213200495), the Key Technologies R&D Program of Jiangsu Province of China(BE2007058),the National Natural Science Research Program of Colleges of Jiangsu Province of China (08KJB520005).

References

1. Rowstron, A., Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In: Guerraoui, R. (ed.) Middleware 2001. LNCS, vol. 2218, pp. 329–350. Springer, Heidelberg (2001)
2. Zhao, B.Y., Huang, L., Stribling, J., Rhea, S.C., Joseph, A.D., Kubiatowica, J.D.: Tapestry: A Resilient Global-scale Overlay for Service Deployment. J. IEEE Journal on Selected Areas in Communications 22, 41–53 (2004)
3. Wallach, D.S.: A Survey of Peer-to-Peer Security Issues. In: International Symposium on Software Security, pp. 31–338. Springer, Heidelberg (2003)
4. Lua, E.K., Crowcroft, J., Pias, M., Sharma, R., Lim, S.: A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. J. IEEE Communications Survey and Tutorial 7, 72–93 (2005)
5. Rescorla, E.: Introduction to Distributed Hash Tables. Technical report, IAB Plenary, IETF-65 (2006)
6. Sit, E., Morris, R.: Security Considerations for Peer-to-Peer Distributed Hash Tables. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 261–269. Springer, Heidelberg (2002)
7. Klemm, F., Girdzijauskas, S., Le Boudec, J.-Y., Aberer, K.: On Routing in Distributed Hash Tables. In: 7th IEEE International Conference on Peer-to-Peer Computing, pp. 113–122. Springer, Berlin (2007)
8. Stoica, I., Morris, R., Karger, D., Kaashoek, F., Baladrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications. In: Annual Conference of the Special Interest Group on Data Communication, pp. 149–160. ACM Press, New York (2001)
9. Hildrum, K., Kubiatowicz, J.: Asymptotically Efficient Approaches to Fault-tolerance in Peer-to-Peer Networks. In: Fich, F.E. (ed.) DISC 2003. LNCS, vol. 2848, pp. 321–336. Springer, Heidelberg (2003)

10. Castro M., Druschel P., Ganesh A., Powstron A., Wallach D. S.: Secure Routing for Structured Peer-to-Peer Overlay Networks. In: 5th Symposium on Operating Systems Design and Implementation (2002)
11. Castro, M., Druschel, P., Charlie Hu, Y., Rowstron, A.: Topology-aware Routing in Structured Peer-to-Peer Overlay Networks. Technical report, Microsoft Research (2002)
12. Wang, P., Hopper, N., Osipkiv, I., Kim, Y.: Myrmic: Secure and Robust DHT Routing. Technical report, DTC Research Report (2006)
13. Kamvar, S.D., Schlosser, M.T.: EigenRep: Reputation Management in P2P Networks. In: 12th International World Wide Web Conference, pp. 123–134. ACM Press, New York (2003)
14. Marti, S., Ganesan, P., Garcia-Molina, H.: DHT Routing Using Social Links. In: Voelker, G.M., Shenker, S. (eds.) IPTPS 2004. LNCS, vol. 3279, pp. 100–111. Springer, Heidelberg (2005)
15. Singh, M.G.: Routing Networks for Distributed Hash Tables. In: 22nd Annual Symposium on Principles of Distributed Computing, pp. 133–142. ACM Press, New York (2003)
16. Rarnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A Scalable Content-Addressable Network. In: Annual Conference of the Special Interest Group on Data Communications, pp. 161–172. ACM Press, New York (2001)
17. Yu, Z.H.: Analysis of Malicious Behaviors in Peer-to-Peer Trust Model. J. Computer Engineering and Applications 43, 18–21 (2007)