DHT Network with Link Access Control using a Social Network

Kimihiko Ando
Graduate School of Technology
Tokyo University of Agriculture & Technology
2-24-16 Naka-cho Koganei-si, Tokyo, Japan
ando@tela.cs.tuat.ac.jp

Atsuo Fukagai
Graduate School of Technology
Tokyo University of Agriculture & Technology
2-24-16 Naka-cho Koganei-si, Tokyo, Japan
afukagai@tela.cs.tuat.ac.jp

Kohta Ohshima
Institute of Symbiotic Science and Technology
Tokyo University of Agriculture & Technology
2-24-16 Naka-cho Koganei-si, Tokyo, Japan
kohta@cc.tuat.ac.jp

Matsuaki Terada
Institute of Symbiotic Science and Technology
Tokyo University of Agriculture & Technology
2-24-16 Naka-cho Koganei-si, Tokyo, Japan
m-tera@cc.tuat.ac.jp

Abstract

Services on Peer-to-Peer (P2P) networks tend to be spread throughout the world. However, there are several problems with Peer-to-Peer networks, such as illegal file sharing or information leaks. These problems are caused by the ability to freely connect many anonymous nodes. The present paper proposes a method of access control applying a social network to a P2P network. We can achieve access control (for example, node A can connect to node B, which is friend of node A, but node A cannot connect to node C, which is friend of node B). We also propose dynamic movement of a node on the DHT circle in order to resolve the problem of increasing path length. This increase is caused by applying a social network to a P2P network. We implemented the proposed method on a pure P2P network using DHT with a circular network structure. In addition, we evaluated the proposed method using a newly developed software simulator. Finally, we demonstrated that expected access control is achieved and that the average number of connections is sufficiently small.

1. Introduction

Social networks have high affinity with computer networks and have various advantages when applied to computer networks. A social network is a graph of real human relationships (e.g., relationships with friends). Services, such as mixi[1] and myspace[2], that apply a social network have seen increasing memberships. These services can identify users and execute access control of content. Internet Phone systems such as Skype are constructed and controlled through a social network of acquaintances. The social network is effective in access control of content based on trust between real friends.

Peer-to-Peer (P2P) networks have some advantages in that they easily provide services, offer scalability and robustness, and are not controlled by any specific institution. However, performing access control on a P2P network is difficult because of the anonymity of the network itself. Therefore, in many cases, P2P networks are used for services such as file sharing systems. Moreover, personal information or secret information is sometimes leaked because of security problems associated with P2P networks.

The present paper describes access control of a P2P network by applying a social network.

2. Problems with Peer-to-Peer Networks

There are a number of problems in the application of P2P networks. Two typical problems are information leaks and copyright infringement by file sharing. These problems are related to the ability to connect to anonymous nodes. All connections in P2P networks may be made transparent, and any node may be accessed from any other node. P2P networks consist of trusted nodes and malicious nodes. When a malicious node relays data, the data may be altered. As a result, the trusted nodes may have cooperated unknowingly in the diffusion of illegal data. Gnutella[8], Winny[10], and Kazaa[14] are networks that are connectable by anyone and are used for file sharing.

In order to solve these problems, it is necessary to create a trusted node and to decide the access control and diffusion

978-0-7695-3297-4/08 $25.00 © 2008 IEEE
DOI 10.1109/SAINT.2008.23

range of data. File sharing is enabled by allowing anyone to access data, which may cause information leaks. Therefore, we assume friends and acquaintances in the real world to be reliable and designate them as trusted nodes.

3. Proposed Network

Figure 1 shows the concept of the proposed P2P network. The top of the figure shows human relationships (social network) that exist in the real world, and the bottom of the figure shows links between computers. The relationships in the social network are transformed into a computer network. This network can construct identifiable trusted routes using trusted friend networks for connections between nodes. This route has the following advantages.

- Since each node knows the communicating target node, access control can be performed by a human operator.
- We can communicate safely and detect a malicious node easily because few particular safety nodes are communicated.
- We can deal with personal information and secret information on the network because we can select nodes that can be accessed and communicate with these nodes without forwarding.
- We can reduce the amount of traffic because queries are sent only to friends.

  Figure2 shows an example of access control. Node A is friend of node D and node E. Node D is a friend of node C. Node E is a friend of node H. Nodes B, G, and F are not friends with anybody on the network. Node A trusts his friends and has allowed them access. In addition, node A trusts node E's friend (node H) and has allowed him access. However, node A does not trust node D's friends, and so node C is denied access by node A.

Although the proposed network has various advantages, the following limitations also apply.

1. Communication with all nodes is not possible.
2. A person can only join a network if they have friends that belong to the network.
3. The friend who has transmitted data knows with whom he is communicating.

With respect to limitations (1) and (2), if several nodes join a network, a node can communicate with all nodes due to the small-world phenomenon. Networks are divided for security reasons, and so exclusive communities arise.

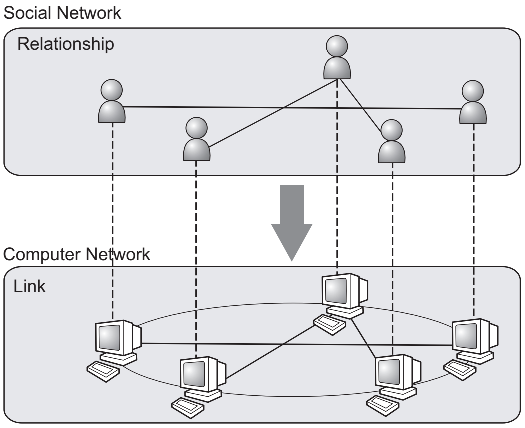


Figure 1. Concept.

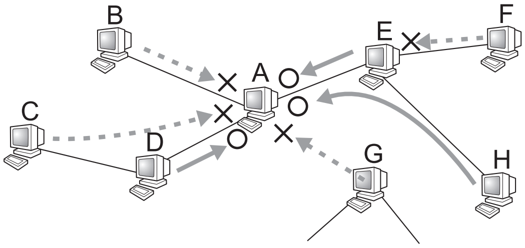


Figure 2. Fine Access Control.

Limitation (3) is a disadvantage. However, we consider that a user that participates in a proposed network needs trusted friends who are allowed to obtain the user's information. If we want to use trusted routes, a trusted friend is required. Then, the network is reliable.

3.1. Network Structure

The proposed network has the following requirements.

1. A node must be able to search friend's nodes even if the IP addresses of some nodes are changed dynamically.
2. The network must have links that can reflect social networks. Since the number of friends is flexible, the number of link must also be flexible.
3. The network does not require a server, because the load is concentrated at the server, and servers are often the target of attacks. If we upload personal information to a server, the administrator of the server can access this information, which increases the risk of leaks.

We chose the DHT-like Chord, which has a circular network structure, as the proposed network structure. Hybrid

P2P networks such as Napster[6] and Kazza were not selected based on requirement (3). Networks that use flooding to send queries, such as Gnutella or Winny, are excluded based on requirement (1). Tapestry[21], Pastry[3], and CAN[18] were excluded due to requirement (2). We constructed the proposed network based on Symphony[13], which is a circular DHT network.

3.2. Structure of the Proposed network

The structure of the proposed network is based on Symphony, which consists of a circular base network and shortcut links (Fig.3). The network consists of nodes and links of a base network and two types of shortcut links. The links are described in detail below.

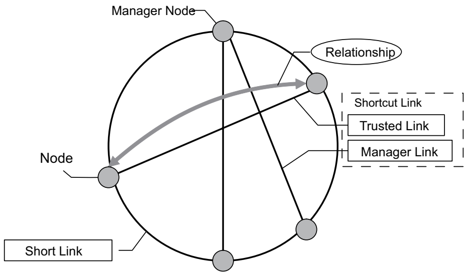


Figure 3. Proposed Network Structure.

- Short Link

  There is a link between adjacent nodes. This link constructs a ling network and manages the proposed network when a node participates in or leaves the network.

- Trusted link

  This link reflects a social network, such as relationship of friends, and becomes a trusted path. Information of a service is passed through this link.

- Manager link

  This link connects a 'manager node' (describe in Section 4.3) to a maintained a node ID.

3.3 Data Types

Transmitted data on the network is divided into two types: service data and control data.

- Service Data

  This data is used for services (such as voices, files, personal information, and diaries) and queries of services. The format of service data is free style, and applications can decide the data format. Service data passes through only a trusted links because of this data relates to privacy or copyrighted material. However, we cannot send this data to all nodes because some nodes cannot be reached through relationships among acquaintances. This data does not send anonymous nodes or denied nodes.

- Control Data

  This data consists of commands for constructing a network, such as searching nodes and making links. Applications do not deal with this data. This data consists of command sets written in XML. In order to reach all nodes, this data is sent by all links (short, trust, manage).

4. Problems and Solutions

Realization of the proposed network requires that certain some problems be solved. In Section 4.1, We describe an authentication method that is used to acquire trusted links between nodes. We describe dynamic movement of a node to solve the problem of long paths in Section 4.2. We describe two types of ID in Section 4.3 and forwarding of queries in Section 4.4 in order to solve the problem of dynamic moving nodes. In Section 4.5, we describe participation in the network and leaving the network.

4.1. Authentication

In the proposed network, a node must be authenticated before communicating service data. Therefore, a node constructs a trusted link called a 'service link'. All of the service data is encrypted and communicated through a service link. Authentication is realized by a public key encryption system with a unique ID and a certification.

A node must record registration information contained in the route (e.g., from node A through node B, C, and D), because we must allow authentication of a node of a 'friend of a friend'. Registration information consists of recorded route information using a node ID. Access control is then realized by the registration information when a node is authenticated. We can set access control finely because each node can be assigned specific access control settings and an authentication method, as described below in detail.

An example of authentication and establishment of a service link is shown Fig. 4. Node A has already established a service link between node C. Then, when node A wants to connect to node B, node C has a list of nodes that is passed by node A to connect to node C, e.g., A, E,... , D. This procedure is shown below.

1. Node A send a connection-request to the service link between node A and node B to node C. Node C adds

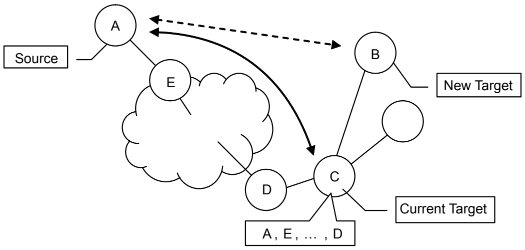


Figure 4. Constructing a Service Link.

the node ID of node A to the registration information of node C. See Fig. 5 (1)

2. Node C send a connection-request from node A and the public-key of node A, as well as the registration information of node C. See Fig. 5 (2)
3. Node B sends the result of authentication to node C. Node C forwards this result to node A. See Fig. 5 (3)
4. Node A establishes a service link with node B if connection is allowed. See Fig. 5 (4)

We can use a trusted route to exclude a spoofed or malicious node when performing the above-described procedure.

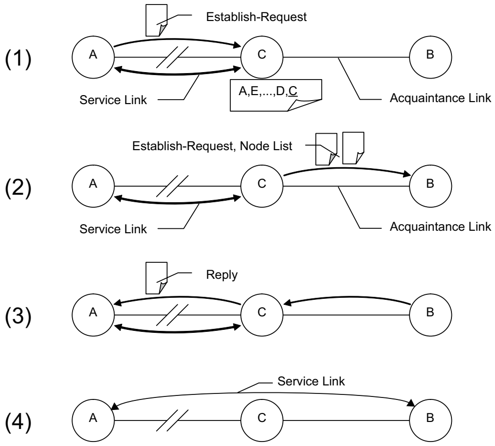


Figure 5. Procedure of Service Link Construction.

4.2. Movement of Nodes

The proposed network is inefficient with respect to path length, because the construction of a trusted link by a social network is made without an efficiency of a network. As a result, the path length is increases tremendously. Nodes are dynamically moved to better positions (Fig. 6). If any nodes have moved, left, or participated in the network, then the node is moved to the correct position in order to reduce the path length of queries. The correct position is decided by the ID of all nodes that are connected directly to a trusted link.

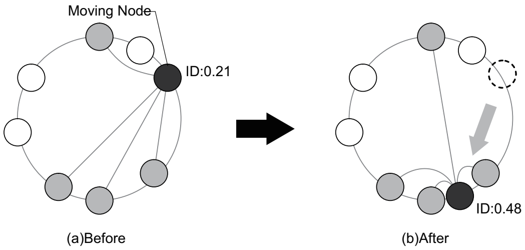


Figure 6. Node Repositioning.

4.3 Dynamic ID and Static ID

When each nodes moves due to the network condition, a node ID that indicates the node position on the network also changes. Thus, when a node moves, we cannot access the node because we identify a node by an ID of the node. Therefore, we use two types of ID (Fig. 7) and a 'manager node' (Fig.8). The manager node connects to managed nodes by manager links and changes the ID of the connected node into an associated ID.

- Dynamic ID

  This ID is decided randomly whenever the node participates in the proposed network. This ID indicates the current position of a node. If the state of the network is changed, for example, if certain nodes participate in or leave the network, then this ID is also changed automatically. When a query is forwarded, a node decides a forward node based on this ID.

- Static ID

  This ID is unique to each node. Each node is identified by this ID. This ID does not change if a node participates in or leaves the network. When a node wants to send a query to a destination node, the node sets the static ID of the destination node.

- Manager Node

  This node converts a static ID into a dynamic ID. All nodes become manager nodes. The range of static IDs covered by a manager node is from the dynamic ID of the node to the dynamic ID of the adjacent node to the right.

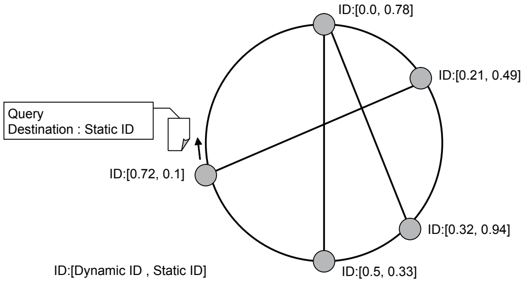


Figure 7. Static ID and Dynamic ID.

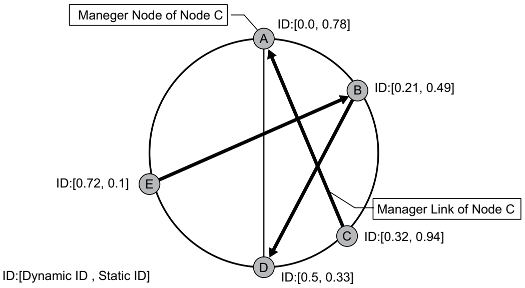


Figure 8. Manager Node.

4.4. Improvement of the Method of Sending Control Data

When a node moves, it must be possible to send control data with certainty, even if a node moves dynamically. We solve this problem using the procedure shown Fig.9. In this case, we use all of the links because the sent data is control data. Figure 9 shows that node B sends a query to node C, and node A is a manager node of node C.

1. Node B makes a query that includes the static ID of a node, the IP address of Node B, and the port number of Node B.
2. Node B sends the query to node D because the dynamic ID of node D is nearest the static ID of the destination node among nodes connected by trusted links.
3. Node D forwards the query to node E, the dynamic ID of which is nearest the static ID.
4. The query is forwarded from node E to node F and then from node F to node A in a manner similar to that described above.
5. Node A forwards the query to node C by a manager link because node A is a manager node of node C.
6. When node C receives the query, node C communicates directly with node B.

We can send a query to a node that moves dynamically by following the above procedure.

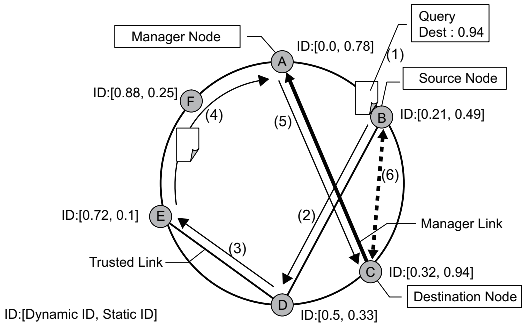


Figure 9. Extended Query Sequence.

4.5. Participating of a Node

Since the proposed network is a pure peer-to-peer network, there is no login server. We must obtain an initial node list that is list of nodes that already participate in the network and then send a participate-request to a participating node. The procedures for participation are shown below.

1. The dynamic ID of new node is decided at random.
2. The node establishes a link between itself and a participated node.
3. The node moves to a dynamic ID position.
4. The node searches a manager node and establishes a manager link.
5. The node searches acquaintance nodes and establishes trusted links.

If a node participates in the network, trusted links between the node and acquaintance nodes are established because the above procedure is performed using all links that include short links.

4.6. Leaving of a Node

When a node leaves the proposed network, the node terminates a link with no additional actions to close because the network is maintained even if a node is disconnected accidentally. However, a node that is connected to a node that has left the network must maintain its connections to the network. This is described in detail below.

- A node connected by a short link

  The node constructs a new short link between itself and the new adjacent node immediately upon detecting a disconnection, because situations such as a short link being disconnected are unfavorable with respect to maintaining the network. Each node connects to three backward-and-forward nodes in order to reconnect immediately.

- A node is connected to a trusted link

  The node does nothing.

- A node is connected to a manager link

  If the node is a manager node of a disconnected node, the node does nothing. If a disconnected node is a manager node, the managed nodes must search for a new manager node and connect to the manager node by a manager link.

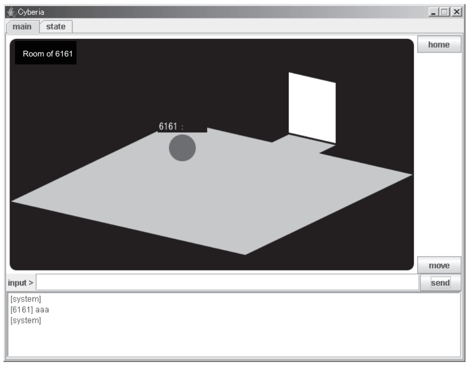
The above procedure is performed even if a number of nodes are disconnected.

5. Implemented Application

We implemented a prototype application in order to confirm the behavior of the proposed network (Fig. 10). The application is an instant messaging system with an SNS function. Each user has a room and performs the above-described actions on a user's avatar in the room.

- Send a message
- Receive a message from other avatars in the room
- Move to another user's room

A move operation is limited to access by the proposed network. An avatar can only be moved to one's own room or to the rooms of acquaintance's of the owner of the room that is being visited. An owner can limit access to own room by checking a moving records of a visiting node.



```
[system]
[6161] aaa
[system]
```

Figure 10. Screenshot of Implementation.

6. Evaluation

We evaluated the proposed network using a newly developed simulator. The example social network used for simulation has the feature whereby most of the nodes have few acquaintances and some of the nodes have several acquaintances (Fig. 11). This feature follows a power law in a social network[15].

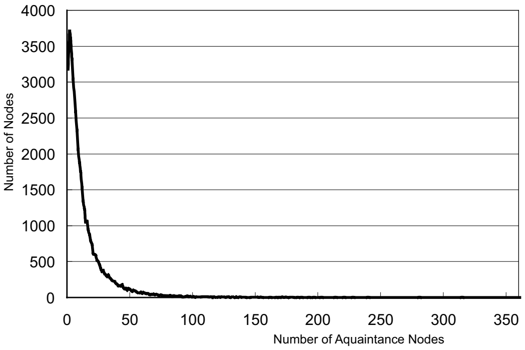


Figure 11. Example of a Social Network.

6.1. Number of Trusted Links

We simulate the number of trusted links of each node, and examine how the number of trusted links changes when the total number of nodes is changed. In a social network, the number of trusted links is the number of acquaintance nodes. Figure 12 shows the results of the simulation.

In the results of the simulation, the total number of trusted links increases in proportion to the size of the network. Similarly, the average number of links increases in proportion to the number of nodes. But an increase rate is small. The average number of links is 0.53, because the network consists of 1,000 nodes. Then, the number of links is 14.72, because the network consists of 50,000 nodes. This value is approximately half that for Chord, which is rather small. Therefore, the load of some nodes that have many acquaintances is large.

6.2. Access control

Table 1 show a comparison of access control methods for P2P networks. In a table 1, 'Server' refers to a paper[20] and 'Resident Program' refers to a paper[12].

Access control of the proposed network is performed by not only one's own node, but also friend's nodes and forwarded nodes. Users may assume that a node that is connected by a malicious node is a malicious node. Therefore, malicious nodes or lazy nodes that do not limit access are

Table 1. Comparison of Access Control Methods

Method	Proposed Network	Server	Resident Program
Network Type	PureP2P	HybridP2P	PureP2P
Access Control Target	Static ID, Certificate	Secret Key	File Name, Extension
Action	Refuse/Establish Connection	Impossible Decode File	Refuse Access to File
Control Timing	Establishing Connection	Decoding File	Accessing File by Applications
Access Controller	Own Node, Forwarding Node	Owner Node, Server	Own Node
Target Object	Node	File	File

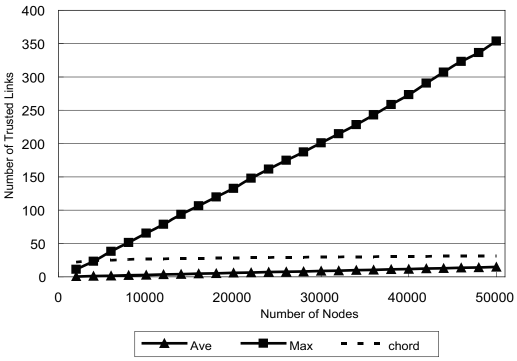


Figure 12. Number of Trusted Links.

refused by some nodes. As a result, the security of the proposed network is ensured automatically.

Spoofed nodes do not appear in the proposed network because access is limited based on static IDs and certificates. If a node pretends to be another node, the node is not permitted to join the network because the node has no certificate. If a malicious node is refused by all nodes, the malicious node will change its static ID. However, the new node cannot access the network because the user of the node is no longer trusted by his or her friends, and so the node has no trusted links. Thus, the malicious node is banned by the other nodes.

The proposed network can achieve access control on a pure peer-to-peer network. Since servers are often the target of attacks, we consider a server-less network to be more secure.

The proposed network can prevent DoS attacks because this network can reject the establishment of links from malicious nodes. Rejection occurs at not only a target node, but also at forwarding nodes. Therefore, we assume that bandwidth and node resources are rarely consumed by improper requests.

Based on the above considerations, we believe that the proposed network can efficiently perform access control.

7. Related Research

Several access control methods for P2P networks have been investigated. However, these methods differ from the proposed access control method in that, in the previous methods, access control is handled by a file stored in a computer[12], by a server, by a USB memory[16, 20], or by LAN restriction[11].

A P2P network in which a social network was applied was also investigated. Most of the research in this area[5, 17, 4, 19] has examined the reduction of download time or search time. These studies differ from the preset research in that our goal is access control. In the present study, a social network was used to increase reliability[7], i.e., to prevent route DoS attacks.

An example of SNS by a P2P network is Afferio[9], which realizes an SNS based on web servers that are linked by a P2P network and perform access control. Affelio is similar to the proposed network in that a social network is used for access control. However, the goal of Affelio is to construct an SNS, whereas the goal of the present study was to construct a P2P network platform.

8. Conclusion

In the present paper, we proposed a P2P network with a user base access control function intended for application to a social network. We developed active movement of nodes in order to solve the problem of long path length. We implemented a prototype application and evaluated the application using a newly developed software simulator. We confirmed fine access control and demonstrated the average number of connections to be sufficiently small.

Acknowledgment

This research is partly supported by MEXT Fund for Promoting Research on Symbiotic Information Technology.

References

[1] mixi. http://mixi.jp/.

[2] mixi. http://www.myspace.com/.

[3] R. A. and D. P. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peersystems. In *IFIP*, pages 329–350. ACM, 2001.

[4] C. Anwar, W. Yurcik, V. Pandey, A. Shankar, I. Gupta, and R. H. Campbell. Leveraging 'social-network' infrastructure to improve peer-to-peer overlay performance: Results from orkut. In *Networking and Internet Architecture*. ACM, 2005.

[5] H. CHEN, M. YANG, J. HAN, H. DENG, and X. LI. Maze: a social peer-to-peer networking. In *CEC'04-East*. IEEE, 2004.

[6] S. FANNING. Napster. http://www.napster.com.

[7] S. M. P. Ganesan and H. Garcia-Molina. Dht routing using social links. In *IPTPS*, pages 100–111. IEEE.

[8] Gnutella. The gnutella protocol specification v0.4. http://dss.clip2.com/GnutellaProtocol04.pdf.

[9] A. Inc. Affelio: The open social network. http://open.affelio.jp/.

[10] I. Kaneko. *The Technology of Winny*. ASCII BOOKS, 2005.

[11] T. Kashima, R. Uda, M. Ito, S. Ichimura, K. Tago, T. Hoshi, and Y. Matsushita. A proposal of secure and low-priced p2p distributed file system with pc share. In *DICOMO2004*, pages 261–264. Information Processing Society of Japan, 2004.

[12] K. Kida, H. Sakamoto, H. Shimazu, and H. Tarumi. A proposal if file access control software agent toward using p2p file sharing system in safet. *Information Processing Society of Japan*, 48(1):200–212, 2007.

[13] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed hashing in a small world. In *4th USENIX Symposium on Internet Technologies and Systems*, 2003.

[14] S. Networks. Kazaa. http://www.kazaa.com.

[15] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. In *PNAS*, volume 99, pages 2566–2572, 2002.

[16] K. Ohtsu, R. Uda, M. Ito, S. Ichimura, K. Tago, T. Hoshi, and Y. Matsushita. Ap proposal of a p2p file sharing system with access control mechanism. In *DICOMO2004*, pages 265–268. Information Processing Society of Japan, 2004.

[17] J. Pouwelse, P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. Epema, M. Reinders, M. van Steen, and H. J. Sips. Tribler: A social-based peer-to-peer system. Technical report, Delft University of Technology, 2006.

[18] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content addressable network. In *SIGCOMM*, pages 161–172. ACM, 2001.

[19] Y. Upadrashta, J. Vassileva, and W. Grassmann. Social networks in peer-to-peer systems. In *International Conference on System Sciences (HICSS'05)*, 2005.

[20] Y. Watanabe and M. Numao. Access control for encrypted data in p2p data shareing. *Information Processing Society of Japan*, 44(10):2437–2443, 2003.

[21] B. Y. Zhao, L. Huang, J. Stribling, A. D. J. S. C. Rhea, and J. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *Selected Areas in Communications. IEEE*, 22(1), 2004.