가비지 파일의 수신을 줄여줄 수 있는 효율적인 익명 모바일 P2P 프로토콜

최운봉*, 오희국*, 김상진**
*한양대학교 컴퓨터공학과
**한국기술교육대학교 인터넷미디어공학부
e-mail : wbchoi@infosec.hanyang.ac.kr

An Efficient Anonymous Mobile P2P Protocol Reducing Garbage Files

Yunfeng Cui*, *Heekuk Oh* *, *Sangjin Kim* **
*Department of Computer Science and Engineering, Hanyang University
**School of Internet Media Engineering, Korea University of Technology and Education

Abstract

With the increasing popularity of P2P file sharing and advancement of mobile technologies, mobile P2P has revealed its attraction. Anonymity has become an increasing requirement in mobile networks. To reduce receiving garbage files, file validation and filtering are other requirements in the mobile P2P environment. If there are effective file filtering and validation mechanism, nodes' battery duration will be saved. In this paper, we do an analysis of security and anonymity in P2P file sharing and exchange system in mobile ad hoc environment, and propose a new efficient anonymous protocol, which can provide anonymity by broadcasting with a probabilistic algorith m and hiding real hop count information, the file validation by the file's special hash value and file filtering mechanism through the collaboration of middle nodes.

1. Introduction

Mobile ad hoc network(MANET) is very popular research area due to the rapid advancement of mobile technologies, while the P2P file sharing system is also much more used on the Internet. Due to the lack of fixed infrastructure both in MANET and P2P networks, P2P file sharing seems natural and attractive to be deployed for MANET. With the scarcity of bandwidth, short lifetime of the nodes due to power constraint and dynamic topology caused by the mobility of nodes in mobile ad hoc environment. The ad hoc and heterogeneous nature of mobile P2P systems, however, can present significant challenges to application designers - particularly when it comes to security and privacy. In common P2P system, users have enough energy. If a wrong file was received, users could just download it again. Energy waste is not such a big problem for users. But energy waste means the decrease of the users' lifetime in the mobile P2P system. Therefore, reducing garbage files becomes very important for users in mobile P2P networks.

In this paper, we propose a new P2P file sharing and exchange protocol in mobile ad hoc networks. This protocol provides mutual anonymity, the file validation and file filtering mechanism. It can provide anonymity by broadcasting with a probabilistic algorithm and hiding real hop count information. With the help of file's special hash value, the file validation issue is resolved. Although we can confirm whether the responder has the desired file, the responder also can send the garbage file at transporting stage. To solve this problem, the nodes between initiator and responder can use some file filtering mechanism to check the data to reduce garbage transporting. The rest of this paper is organized as follows: Section 2 presents the major security threats peculiar of the P2P system in mobile ad hoc networks and the privacy problem – anonymity. Section 3 presents the proposed protocol design. Finally we conclude the work in Section 4.

2. Related Work

No preassigned and centralized trust authority, high dynamic network topology and vulnerability of wireless link are the main actors in the mobile P2P networks, which make a lot of security violations increase. For the file sharing and exchange system in MANET, the most import thing is to avoid false download which wastes precious battery power. However in any P2P system you cannot eliminate receiving garbage data completely. Attackers just keep consuming battery energy until the device will power off in the end. So how to reduce garbage file transporting is very important. Another issue is data integrity, as there is limited confidence in the authenticity and quality of the exchanged files. It is impossible to determine whether a download file contains the desired portion until it has been completely received and viewed.

[†] This research was supported by the Ministry of Knowledge Economy, Korea, under the HNRC(Home Network Research Center) ITRC(Information Technology Research Center) support program supervised by the Institute of Information Technology Assessment.

Without proper encryption, anyone can eavesdrop and sniff any data transmitted on the air. To address these problem, here we introduce a P2P file sharing and exchange protocol in mobile ad hoc networks which is called "Divalia"[1]. In this protocol, files are partitioned into segments. So the user can request to download individual segments from various peers, and then reassemble all the segments finally. To address the data integrity issue, the concept of fingerprint is imported. Using the fingerprint information, users can affirm whether the file provided by the other node is correct or not.

Nowadays anonymity has become an increasing requirement in wireless networks. Also every node in mobile P2P must communicate with its neighbors. As no node can know the other nodes two or more hops, partial anonymity can be achieve at least between those non-neighboring nodes in the mobile ad hoc P2P environment. Hence, in mobile P2P system both sender and receiver anonymity cannot be achieved in each node very local environment. However, if mutual anonymity (both sender and receiver) can be provided in this environment among these non-neighboring nodes, it is good for protecting the privacy of users. There are several protocols which can provide mutual anonymity for P2P file sharing system, such as P5 (Peer-to-Peer Personal Privacy Protocol)[2] and AFPS(anonymous Peer-to-Peer File Sharing)[3]. The basic idea of P5 is that all the nodes in the hierarchy channel send fixed length encrypted noise messages at a fixed rate as if all nodes are grouped in a logic ring. As it is based on the assumption that the initiator knows the public key of query responder, it is difficult to be used in mobile P2P networks. APFS uses Onion [4] as the base to build their protocol. In APFS, volunteering peers should be taken in, which can affect the performance of P2P system. This change make the coordinator should examine each volunteering peer before assigning the task, which will increase the communication overhead. Here we introduce a file sharing and exchange protocol in mobile ad hoc networks, called Secret-sharing-based Mutual Anonymity (SMA) [5]. In SMA, both Shamir's (k, n) secret sharing scheme (SSS)[6] and the information dispersal algorithm(IDA)[7] is used together with onion routing to achieve the mutual anonymity. Using secret sharing to transport the file data can achieve the sender (responder) anonymity, but the overhead will be too high. Furthermore, mutual anonymity is not good for receiver in mobile P2P to find out the node who always sends the wrong file because receiver cannot identify any sender. Another novel protocol MAPCP [8] goes a different way to achieve anonymity by broadcasting with a probabilistic algorithm to control packet flooding in the data transmission phase.

3. Protocol Design

Similar to most P2P applications, communication in our proposal consists of two phases: the query phase and the data transmission phase.

A. Notation

In this paper, the following notations are used to describe protocols.

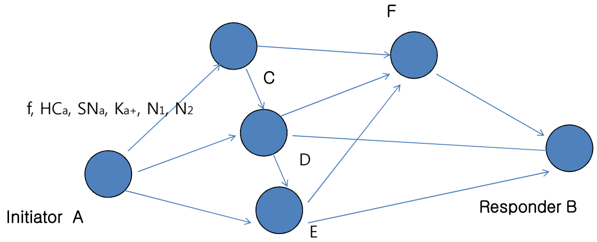
- F: the desired file
- K_{X+}, K_{X-}: a public and private key of a principal X.
- (M).K: Encryption of a message by using a key K.
- SN_X: the sequence number of query initiated by X which is 128-bit random nonce
- HC_X: hop count which is a random positive integer generated by X
- SK_{mn}: the session key with node n which is generated by sender m
- PID_X: the pseudo ID of sender or responder
- F_i: one segment of the real file
- H_i: the hash value of F_i
- $H_{[j,k]}$: the hash of the fragment of F starting at j-th byte and ending at the k-th byte
- N: the real name of the file
- L: the length of the file
- H: the hash value of the entire file
- $\{H_i\}$: the hash value set for each segment of the file

B. Query Phase

When the initiator Alice wants to query a file, she first should take more information about the file, which can identity that file such as the hash value of the file, the length of the file and so on. Initiator A generates $HC_a>1$. The query message should be created as following:

$A \rightarrow * : f,\ c,\ K_{a+},\ HC_a,\ N_1,\ N_2$

where f is the description of the desired file, N_1 and N_2 are random integers with the same bit length chosen by A (Figure.1).



node	Sour. SN	Dest.SN	min（HCa）	min（HCb）
C	SNa	Null	Hca	MAX
D	SNa	Null	Hca	MAX
E	SNa	Null	Hca	MAX
F	SNa	Null	HCa+1	MAX

Figure.1 Query Phase

The initiator A broadcasts the message and keeps entry {null, null, HC_a, MAX, null} in its own destination table, where MAX is a very large positive integer. When intermediate node i receives the message, it increases HC_a by 1 and forwards it to its neighbors. And then i checks whether the query could be satisfied. If it cannot be satisfied, i saves entries { SN_a, null, min(HC_a), Max } into its own destination table by the format {Source SN, Destination SN, min(HC_a), min(HC_b)}. Otherwise, if yes (here i is responder B), it generates SN_b, PID_b, SK_{ba}. Here, SK_{ba} can only be used for further communication with the initiator A. Then it broadcasts to his neighbors the query reply as following:

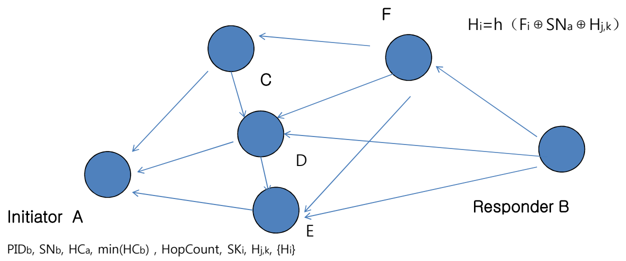
$A \rightarrow *$: SN_a, SN_b , HC_b, (PID_b, min(HC_a), SK_{ba}, F's info).K_{a+}

where F's info consists of {$\{H_i\}$, L, $H_{[j,k]}$}. Here, j and k can be calculated through the function G(η, θ, min(HC_a), L)

= jk. $\{H_i\}$ is the hash value for checking each segment of the file. $\{H_i\}$ is not simply the hash vale of file segment, while it is calculated as follows:

$$H_i = h(F_i \oplus SN_a \oplus H_{[j,k]})$$

After sending reply message, responder keep entries$\{SN_a, SN_b, PID_b, \min(HC_a), HC_b, SK_{ba}\}$ in its destination table. When intermediate node i receives the message, it increases HC_b by 1 and forwards it to its neighbor. If j is not A, it updates the entry { SN_a, null, min(HC_a), Max } in its path table to{ SN_a, SN_b, min(HC_a), min(HC_b)}, where min(HC_b) is the minimum HC_b value among all received query replies. Otherwise, if j = A, it decrypts the encrypted part with K_{a-} to get PIDb, min(HCa), SKba, F's info, and calculates Hop Count = min(HC_a)- HC_a as responder sends min(HCa) which it received. Then A updates the entry {null, null, HCa, MAX, null} in its destination table to {PID_b, SN_b , HC_a, min(HC_b), SK_{ba}, HopCount, F's info}.


F
H_i=h（$F_i \oplus SN_a \oplus H_{j,k}$）
C
D
Initiator A
E
Responder B
PID_b, SN_b, HC_a, min(HC_b) , HopCount, SK_i, $H_{j,k}$, {Hi}

node	Sour. ID	Dest.ID	min（HC_a）	min（HC_b）
C	SN_a	SN_b	H_{ca}	HC_b+1
D	SN_a	SN_b	H_{ca}	HC_b
E	SN_a	SN_b	H_{ca}	HC_b
F	SN_a	SN_b	HC_a+1	HC_b

Figure.2 Query Reply

C. Data Transmission Phase

Once node S collects enough query replies, data transmission between A and each file holder R can be done anonymously as follows. Because of knowing the value of Hop Count for every responder R, A can choose several responders to request the data from all responders. Here, we choose B as the responder. Therefore, A looks up B's pseudonym PID_b from its destination table to get PID_b, SN_b, HCa, min(HC_b), SK_{ba}, HopCount, F's info, and broadcasts a data message to its neighbors, which contains PID_b, SN_b, a positive number α = HC_a+min(HC_b), and a SK_{ba}-encrypted part consisting of SN_a, request (e.g. a request for file). This can be written as

$$S \rightarrow * : PID_b, SN_b, \alpha, \{H_i\}, (SN_a, \text{request}).\ SK_{ba}$$

When an intermediate node i, (i is not B), receives a non-duplicate data message, it looks up SN_b in its path table to get min(HC_a) and min(HC_a), and calculates its rebroadcast probability p_j as:

$$\mu = \alpha / (\min(HC_a) + \min(HC_a))$$

$$p_j = \begin{cases} \mu\lambda^{\min(HCa)\ +\min(HCa)-\ \alpha}, & \text{if } \mu < 1, \\ \text{Otherwise} & \text{(a)} \end{cases}$$

where $0 \le \lambda \le 1$ is a real number selected by the protocol. Then, node j forwards this message according to its rebroadcast probability.

When responder node B receives the data query message identified by PID_b, it decrypts the encrypted part with session key SK_{ba} to get SNa and the data. Likewise, if node B intents to send the file desired to A (e.g. the requested file), it looks up SN_a from its destination table to get SN_a, SN_b, min(HC_a), HC_b, SK_{ba}, and then broadcasts a data message containing SN_a, SN_b, a positive number α'= HC_b+min(HC_a) and the requested file segment data to its neighbors.

$$B \rightarrow * : SN_a, SN_b, \alpha', \{H_i\}, Fi \oplus H_{[j,k]}$$

When receiving the data message, each intermediate node calculates its rebroadcast probability using (a) and forwards the data message according to p_j.

The selection of λ represents the tradeoff between anonymity and performance. If λ = 1, the system has the highest anonymity but lower forwarding efficiency, since dummy packets contribute to collision. If λ is close to zero, the system generates the fewest dummy packets and has higher forwarding efficiency.

D. File Filtering

Although the truth that responder B possesses the file is confirmed, B still can send other packets data to A. If there is not an efficient filtering mechanism, A cannot know that B is sending wrong file to her until it gets the packets. It means wasting A's battery duration. In our design, every segment's hash value is broadcasted, intermediate nodes between A and B has the file hash set. Consequently, the nodes between A and B can check whether B honestly send the right file data by calculating the hash values of the data and comparing them with the hash values which is received at the File Query stage. For example, C receives one segment data and calculate H'_i =h($F_i \oplus$ sn $\oplus H_{[j,k]}$)and then compares H'i with H_i. $H_{[j,k]}$ is only known by A except the nodes who possess the real file. So no one can acquire the data F_i except A.

4. Conclusion

In this paper, we analyze security and anonymity of the P2P file sharing and exchange system in mobile ad hoc networks. Then we propose a new protocol, which can not only provide anonymity for receiver and initiator by broadcasting probabilistically and hiding real hop count information but also reduce users' receiving garbage files by file's hash values. To prevent responder's sending wrong segment data at transporting stage, file filtering mechanism is implemented by the nodes between initiator and responder. By calculating hop count value, the initiator can choose better ones from several responders. Our future work is to find a more efficient approach to provide anonymity for all the nodes and more efficient file filtering and trust management mechanism by the collaboration of members in system.

Reference

[1] Ryan Vogt, Ioanis Nikolaidis, Pawel Gburzynski, “Divalia: a practical framework for anonymous peer-to-peer file exchange in wireless ad-hoc networks,” Fourth Annual Conference on Communication Networks and Services Research, pp. 149-156, May 2006.

[2] R. Sherwood, B. Bhattacharjee, and A. Srinivasan, “P5: A Protocol for Scalable Anonymous Communication,” in Proceedings of IEEE Symposium on Security and Privacy, pp. 58 - 70, 2002.

[3] V. Scarlata, B. N. Levine, and C. Shields, “Responder Anonymity and Anonymous Peer-to-Peer File Sharing,”the 9th International Conference of Network Protocol(ICNP), pp. 272-280, 2001.

[4] P. F. Syverson, D. M. Goldschlag, and M. G. Reed, “Anonymous Connections and Onion Routing,” in Proceedings of IEEE Symposium on Security and Privacy, pp. 44-54, 1997.

[5] Jinsong Han, Yanmin Zhu, Yunhao Liu, Jianfeng Cai, Lei Hu, “Provide privacy for mobile P2P systems,” Distributed Computing Systems Workshops 25th IEEE International Conference, pp. 829-834, June 2005.

[6] R. J. McEliece, D. V. Sarwate, “On sharing secrets and Reed-Solomon codes," Communications of the ACM, pp. 583-584, 1981.

[7] A. Shamir, “How to share a secret,” Communications of the ACM, pp. 612-613, 1979.

[8] Chao-Chin Chou, Wei, D.S.L., Kuo, C.-C.J., Naik, K. , “An efficient anonymous communication protocol for peer-to-peer applications over mobile ad-hoc networks,” Selected Areas in Communications, IEEE Journal on volume 25, Issue 1, Jan. 2007