1998 International Computer Symposium
Workshop on Cryptology and Information Security
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

A PROBABILITY MODEL FOR RECONSTRUCTING SECRET SHARING UNDER THE INTERNET ENVIRONMENT

Ching-Yun Lee, Yi-Shiung Yeh, Deng-Jyi Chen
Institute of Computer Science and Information Engineering
National Chiao-Tung University, Hsinchu, Taiwan, R.O.C.
Email: {djchen, leecy, ysyeh}@csie.nctu.edu.tw Fax: (03)5724176
Kuo-Lung Ku
Chung-Shan Institute of Science and Technology, Taoyuan, Taiwan

ABSTRACT

Computer and communications world began a new era with the birth of Internet technologies. The use of Internet for various business applications and resource sharing has accelerated in recent years. Owing to such development, Internet security has become a very important issue. In some applications, an important message may be divided into pieces and be allocated at several different sites over the Internet for security access concern. For example, an important map that can be used to access a military base, a vital key that can be used to give a military order or command. To access such an important message, one must reconstruct the divided pieces from different locations. In this paper, we present a novel probability model for reconstructing secret sharing and an evaluation algorithm to measure the probability of secret sharing reconstruction. Also, how to assign the divided shares into different sites over the Internet is studied.

1. INTRODUCTION

The Internet has revolutionized the computer and communications world. The Internet consists of networks interconnected by a set of routers which allow them to function as a single, large virtual network. The number of users in the network increases massively. According to Network Wizards, the number of hosts on the Internet grew from 1,313,000 in January 1993 to 36,739,000 in July 1998. An added feature of the Internet is resource sharing among remote users, in which user can quickly access data from any site. Given its open and easy access for the general public, the Internet is vulnerable to more attacks from intruders than any other network.

A user can achieve a secure secret in Internet environments by adopting strong encryption/decryption algorithms [1] and secure key distribution protocols [2,3] as well as secret sharing schemes [4,5,6]. Sharing secrets among several individuals in a manner that no individual holds all the secrets is highly desired. Shamir [7] and Blakley [8] pioneered the notion of secret sharing and provided the secret sharing schemes. An (m, n) secret sharing scheme is a method which a secret, S, is divided into n shares in such a manner that the secret S cannot be reclaimed unless at least m shares are collected. This scheme is known as (m, n)-threshold scheme.

A secret can be taken and divided into pieces in several ways. To secure a secret, we can divide secret into shares and store shares at different sites over the Internet. The Internet provides resource sharing among remote users, thereby allowing quick access of a message from any site. A user can reconstruct a secret message by sending requests to the service. A service exports a set of commands. After executing a command, the service can transfer the requested shares to that user.

Two types of shares, single-share and multiple-share, play a prominent role in the development of the share assignment method. Each participant could have only one share or a different number of shares. Single assignment secret sharing scheme assumes that each participant holds only one share. Single share protocol is limited in that if any share gets lost or can not be obtained, the secret message can not be reconstructed. Typically, having two or more participants with the same share is desired owing to the availability. If the share is replicated, whether some of the replicas are down or unavailable is unimportant.

Among the promising applications of the Internet include high reliability/availability and resource sharing. The reliability/availability improvement is owing to the redundant techniques used on the Internet. Some errors or other unexpected factors of network may disconnect the communication network, thereby influencing the performance and reliability of the Internet. Several network reliability measures have been defined in addition to related evaluation methods developed as well [9-13]. The evaluation algorithm proposed in [12] with some

modifications can be used to measure the SSRP (Secret Sharing Reconstruct Probability). The computational complexity of the reliability evaluation algorithm is NP-hard [14,15]. Probability generally refers to a system's ability to carry out a requested operation correctly and efficiently. However, allocating shares to appropriate locations for each user over the Internet enhances the secret sharing reconstruct probability. The distribution of shares heavily influences the SSRP. The share assignment problem is involves finding a share distribution such that the SSRP measure is maximal. Exhaustive approach may be used to find the optimal solution with high computation time. In this paper, we present a simple share assignment (SSA) algorithm based on priority search and some heuristics to achieve better shares assignment.

The remainder of this paper is organized as follows. Section 2 presents the notation, definitions, and problem statements used herein. Section 3 thoroughly describes the probability model and SSA algorithm. Next, Section 4 presents some illustrative examples and simulation results to demonstrate the effectiveness of the proposed algorithm. Finally, the conclusion of the paper is stated in Section 5.

2. NOTATION, DEFINITIONS, AND PROBLEM STATEMENTS

Notation

G(V,E)	An undirected graph in which V represents the set of nodes and E represents the set of edges
S_s	the set of shares such that any m of them can be used to reconstruct the secret, S
SA_i	set of shares available at node i
S_i	single-share, or share i in S_s
M_i	multiple-share in S_s, such as $M_1 = \{S_1 \cup S_2\}$, $M_2 = \{S_3 \cup S_4 \cup S_5\}$
ME	a subset of E that represents the edges merged during the process of finding all MSSTs
$p_i(q_i)$	probability of node, edge, or link i works (fails)
Pr(E)	probability of event E
G – e	the graph G with edge e deleted
G + e	the graph G with edge e = (u, v) contracted such that node u and v are merged into a single node

Definitions

- SST: a share spanning tree that connects the user node (the node that reconstruct the secret under consideration) to some other nodes such that its vertices hold all the required shares.
- MSST: a minimal SST such that there exists no other SST which is subset of it
- HPF-BF Search: The HPF-breadth first search uses the breadth-first search with high probability first, and keep track of nodes visited. In HPF-BF Search, we start from a node u and first visit the node with the highest probability edge incident to u.
- SSRP: The probability that a user can successfully reconstruct a secret over the Internet, i.e., one will be able to access all the shares required from remote sites in spite of faults occurring among the nodes and communication links. The MSSTs connect the user node to other nodes such that these nodes hold all the required shares for the user to reconstruct the secret. The SSRP can be determined by computing the probability that at least one of the MSSTs is operational.

  This can be written as

  $$\text{SSRP} = \Pr\left(\bigcup^{n_{msst}} MSST\right),$$

  where n_{msst} is the number of MSSTs that reconstruct the secret.
- Shadows: We can take any secret and divide it into n pieces, called shadows or shares, such that any m of them can be used to recover the secret.
- Multiple-share: a site with multiple-share (or multiple shares) means it contains more than one share.
- Allocation tree: An allocation tree is a tree that was reconstructed from a network and allocated the share into it.

Problem Statements

The Simple Share Assignment for secret sharing reconstruction is to find a share distribution such that the SSRP measure is maximal. For the share assignment problem, the following information is given:

a) network topology;
b) the shares required to reconstruct the secret;
c) user's location; and
d) the probability of each node and communication link works.

We also assume that the node and link failures are s-independent.

3. THE SYSTEM MODEL AND SSA ALGORITHM

In this section, we present a probability model of secret reconstruction and algorithm to perform shares assignment under the Internet environment. The Internet can be treated as a computer network that is represented by an undirected graph.

3.1 The System Model

Informally, the model is a protocol in which a distinguished processor, or user, selects a secret message and divides it into n pieces, or shares. The divided shares of the secret will be allocated into other processors. A user can access the secret by sending requests to other processors to return the needed shares for reconstruction. Therefore, the model is a probability measure for reconstructing the secret sharing. For example, assume that the commander with anyone of the two vice-commanders is authorized to initiate an order. Herein, we divide a secret message into three shares, S_1, S_2, S_3, and give the commander multiple shares, $\{S_1, S_2\}$ and the vice-commander one each, $\{S_3\}$. Consider the network topology shown in Figure 1 which consists of six nodes with one multiple shares (node 3) and two replicated single-share (node 4, 5).

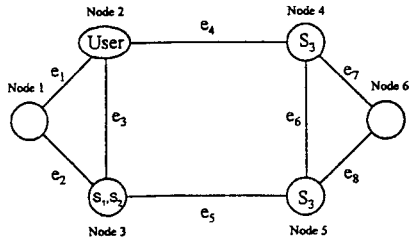

Figure 1. A simple network

A user can either recover the secret or initiate the order by sending a request message to nodes 3, 4, 5 to ask them to return shares S_1, S_2, S_3. In general, the set of nodes and links involved in reconstructing the secret and access its required shares form a tree. A share spanning tree that connects the root node to some other nodes such that its vertices hold all the required shares for reconstructing the secret. Figure 2 shows all MSSTs that represent the site which has shares needed for user to reconstruct the secret for the network application in Fig. 1.

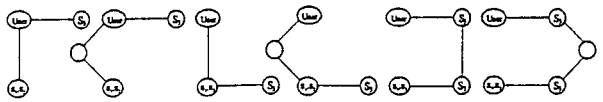

Figure 2. MSSTs for user to reconstruct the secret

Let the probability of all nodes and links being operational be 0.9. The SSRP of secret S can be computed using conditional probability, as shown below.

$$\begin{aligned}SSRP(S) &= p_{N2}\, p_{N3}\, p_{N4}\, p_{e3}\, p_{e4} + q_{e3}\, p_{N1}\, p_{N2}\, p_{N3}\, p_{N4}\, p_{e1}\, p_{e2}\, p_{e4} + (1 - p_{e4}\, p_{N4})\, p_{N2}\, p_{N3}\, p_{N5}\, p_{e3}\, p_{e5} + \\ & q_{e3}\,(1 - p_{e4}\, p_{N4})\, p_{N1}\, p_{N2}\, p_{N3}\, p_{N5}\, p_{e1}\, p_{e2}\, p_{e5} + \\ & q_{e3}\,(1 - p_{e1}\, p_{e2}\, p_{N1})\, p_{N2}\, p_{N3}\, p_{N4}\, p_{N5}\, p_{e4}\, p_{e5}\, p_{e6} + \\ & q_{e3}\, q_{e6}\,(1 - p_{e1}\, p_{e2}\, p_{N1})\, p_{N2}\, p_{N3}\, p_{N4}\, p_{N5}\, p_{N6}\, p_{e4}\, p_{e5}\, p_{e7}\, p_{e8} \\ &= 0.59049 + 0.0478297 + 0.1121931 + 0.00908764 + 0.0129618 + 0.0010499 \\ &= \mathbf{0.7736121},\end{aligned}$$

where p_{Ni} denotes the probability of node i working, p_{ei} denotes the probability of edge i working, $q_{Ni} = (1 - p_{Ni})$, and $q_{ei} = (1 - p_{ei})$.

The probability of reconstructing secret sharing on the Internet can be evaluated by the following two steps: finding all the MSSTs for that user and computing the probability of the MSST to evaluate the SSRP. All disjoint MSSTs can be generated by the following minimal_share _spanning_tree algorithm with some modifications from the reliability algorithm in [12]. The MSST algorithm guarantees no replicated share spanning trees generated during the expansion of computation tree and performs several reduction methods [16] to reduce the size of the problem and to simplify the probability evaluation. The complete MSST algorithm is given below.

Algorithm Minimal_Share_Spanning_Tree *(G, u)*;
Input: G = (V, E, S_s) (the original network), and u (user node)
Output: the SSRP, Secret Sharing Reconstruct Probability
begin
  repeat /* reduce the original network graph G */
    perform degree-1 reduction /* removes degree-1 nodes which contain no required shares and removes their incident edges */
    perform parallel reduction /* Suppose e_a = (u, v) and e_b = (u, v) are two parallel edges in G. We can reduce these redundant edges into a single edge e_c = (u, v) such that $p_{ec} := (1 - q_{ea} * q_{eb})$ */
    perform series reduction /* Let e_a = (u, v) and e_b = (v, w) be two series edges in G such that degree(v) = 2 and node v contains no required shares. We can remove node v and replace edges e_a and e_b by a single edge e_c = (u, w) such that $p_{ec} := (p_{ea} * p_v * p_{eb})$ */
    perform degree-2 reduction /* Suppose node v, with node degree = 2, is a reducible node. Then we can apply series reduction on node v and move the shares within node v to one of its adjacent nodes. */
  until no reductions can be made
  let G' be the network graph after reduction step
  FOUND := ∅
  ME := ∅
  FIND_SST(*G', ME*) /* call FIND_SST to find SSTs */
  for all *s, t* ∈ *FOUND* **do** /* remove the SSTs which are not MSSTs */
    if (t ∩ s) = s **then** remove from *FOUND*
      else if (t ∩ s) = t **then** remove S from *FOUND* **endif**
    endif
    repeat
    nodes are introduced by the intermediary of edged incident to them
    apply the reliability algorithm to all MSSTs in *FOUND* to evaluate SSRP
    output the SSRP
end ***MSST***

Function ***Find_SST(G, ME)***
begin
if there are no SSTs in G **then return**(0) **endif**
if there exists one node v such that $SA_v \supseteq S_s$ **then**
FOUND := *FOUND* $\cup$ {*ME*} **return endif**
for all $e_i \in$ {the set of edges incident to the node that contains the user} **do**
FIND_SST(G + e_i, *ME* $\cup$ {e_i})
G := G - e_i
remove the irrelevant components from G
if there is no SSTs in G **then return endif**
repeat
end *FIND_SST*

The previous example in Fig.1 is used in the following to illustrate the MSST algorithm. Figure 3 depicts the process of finding all MSSTs. The set of all MSSTs of user to reconstruct the secret are {(e_4, e_9, N2, N3, N4), (e_4, e_5, e_{10}, N2, N3, N4, N5), (e_5, e_9, N2, N3, N5)}. Applying the terminal reliability algorithm, e.g. [10,12] to the above MSSTs allow us to compute the SSRP of secret S, as shown below.

$$\begin{aligned} SSRP(S) &= p_{N2}\, p_{N3}\, p_{N4}\, p_{e4}\, p_{e9} + q_{e9}\, p_{N2}\, p_{N3}\, p_{N4}\, p_{N5}\, p_{e4}\, p_{e5}\, p_{e10} + (1 - p_{e4}\, p_{N4})\, p_{N2}\, p_{N3}\, p_{N5}\, p_{e5}\, p_{e9} \\ &= 0.6383197 + 0.0140117 + 0.1212807 \\ &= \mathbf{0.7736121} \end{aligned}$$

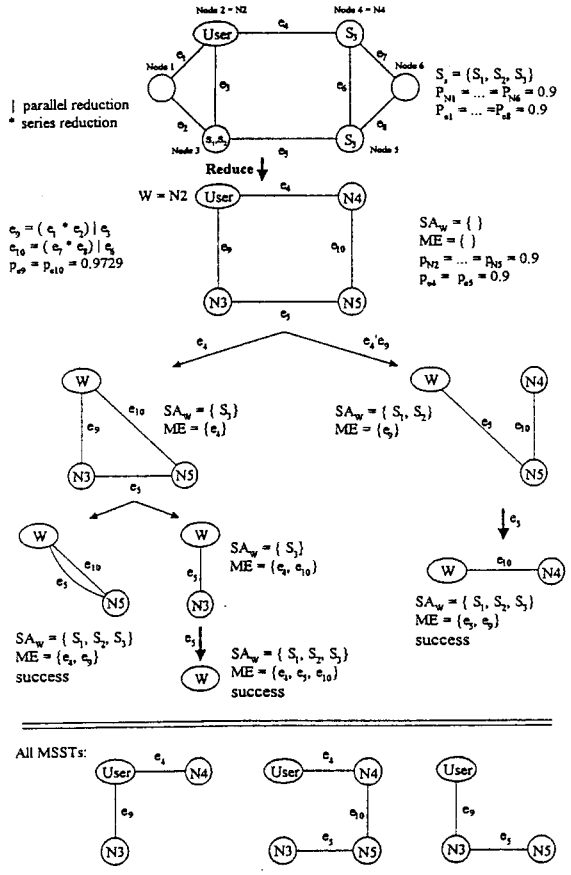

Figure 3. The process of finding all MSSTs

3.2 A Simple Example of SSRP Analysis

Example 1

Herein, a secret is divided into three shares, S_1, S_2, S_3. Assume that one participant is more important than the others and give that participant multiple shares, {S_1, S_2}. We allocate the shares into different sites over the Internet. To recover the secret, the user must obtain the three shares together. Consider the network topology shown in Figs. 4 and 5 which allocated one multiple-share and three different single-share in redundant manner. Let the probabilities of all nodes and links being operational be 0.9. Figure 4 depicts an optimal share assignment to this network topology and Figure 5 is a poor one. These allocations differ only in the share assignments, while the other parameters are identical.

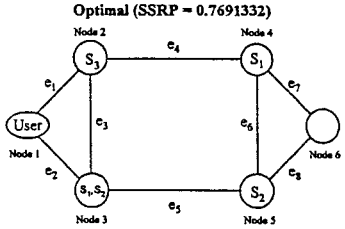

Figure 4. The optimal share assignment of Example 1

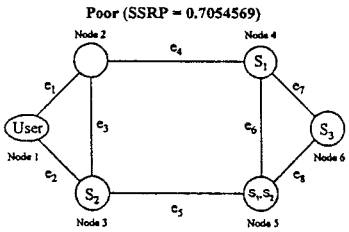

Figure 5. The poor share assignment of Example 1

This example demonstrates hoe share assignments affect the SSRP. Thus, the means of share assignment significantly affects the probability of secret sharing reconstruction. The following section presents a simple share assignment algorithm such that the probability of secret sharing reconstruction can be maximized.

3.3 Basic Strategy of the Simple Share Assignment

3.3.1 Basic Concept

The concept of full set is introduced as follows. Full set is defined as *a set containing all the required shares to reconstruct the secret.* For an (*m, n*)-threshold scheme, the full set of secret *S* is the set of all shares {S_1, S_2,..., S_n} which is a unique case of $m = n$. Full set is the basic requirement of the SSRP. Recall the definition of the

minimal share spanning tree. The MSST for the secret must contain all the elements in the full set. This paper also concerns the combination of the multiple shares for the participants. By doing so, we attempt to get larger number of groups with full set. Thus, the proposed approach is carried out to obtain an acceptable SSRP value. If more full sets are in the network, the number of MSSTs is also larger. Hence, SSRP should be greater owing to $P(A \cup B) \geq P(A)$ by the set theory.

Closely study the above example reveals the following:

a) The transfer probability for a share is data link dependent. The fewer the data links capable of transferring the required data implies a higher transfer probability;
b) The MSST with less number of nodes and links is more reliable; and
c) The SSRP is the probability of the union of the number of MSSTs and can be enhanced by grouping the required shares as close to the user node as possible.

Based on above observations, we propose a heuristic approach for efficiently share assignment. Underlying concepts of the heuristic method are as follows:

1) Allocate the more influential shares first. (Share M_1 is more influential than share M_2 if and only if the number of shares of M_1 is greater than that of M_2);
2) Complete the full set which occupy minimum number of nodes;
3) Allocate the shares to the more reliable sites; and
4) Allocate all shares as close to the user's location as feasible.

The above strategies are applied to develop our share assignment algorithm. In general, the algorithm can be summarized in the following two parts.

Part 1: Classify and determine the order of the shares.

Part 2: Use the HPF-BF algorithm to reconstruct the network into a tree topology, then sequentially allocate these shares into the reconstructed tree according to the order obtained in Part 1.

3.3.2 Reconstructing Network into Tree topology

The network is reconstructed into a tree topology by using the HPF-BF search algorithm to travel all the edges. In HPF-BF search, we start from a node u (the user node) and visit the node from the highest to the lowest probability edge incident to u. Then all nodes adjacent to those nodes are visited, and so on. The HPF-BF algorithm uses the breadth-first search with high probability first to build a HPF-BF tree by including only edges that lead to newly visited nodes. All those edges cumulatively form a tree with user node as its root.

The complete HPF-BF search algorithm is given below.

Algorithm HPF-BF Search *(G, u)*;
Input: G = (V, E) (an undirected graph), and u (the user node).
Output: T = (V, E'): a tree T with node set V and edge set E'.

```
begin
  S := {u}
  L := {e_i | ∀ e_i ∈ E and ∃ v_i ∈ S, such that e_i is incident to v_i}
  E := E - L
  V := V - S
  E' := ∅
  S := ∅
  repeat
     repeat
        e := {e_i | select an e_i ∈ L which has the maximum p_ei}
        L := L - {e}
           if E' + {e} does not have cycle then E' := E' + {e}
           endif
           S := S + {v_i | ∀ v_i ∉ S and e is incident to v_i }
     until L = ∅
  L := {e_i | ∀ e_i ∈ E, ∀ e_i ∉ E', and ∃ v_i ∈ S, such that e_i is
       incident to v_i}
  V := V - S
  S := ∅
  until V = ∅
end HPF-BF
```

3.3.3 An Example of HPF-BF algorithm

Figure 6 depicts a simple example of using HPF-BF algorithm to construct the allocation tree. The parenthesized numbers associated with the nodes denote the order in which the nodes were searched by HPF-BF search algorithm.

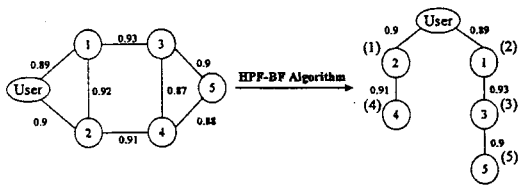

Figure 6. An example of the HPF-BF Search algorithm

3.4 The Complete SSA Algorithm

The SSA algorithm consists of two steps. First, the algorithm determines the allocation order of the shares. Next, the algorithm executes the HPF-BF search and sequentially allocates the shares into the tree. The allocation order of the shares is determined by using the following procedures.

a) Calculate the total number of shares of each multiple-share.
b) Allocate the multiple-share with more unallocated shares first.

c) Calculate the number of unallocated shares of each multiple-share.
d) Select and allocate the share which contains more unallocated shares to complete the full set. Select the multiple-share that contains more shares first to break the tie of the number of unallocated shares.
e) If all shares have been set to "allocated" then set all shares to "unallocated". A share assignment, which has the greatest number of full set groups, is our preferred choice.
f) Repeat steps c, d, and e until all the shares have been allocated.

Algorithm Simple_Share_Assignment;
Input: G = (V, E) (an undirected connected graph), u (the user node), and M (the set of shares to be allocated).
Output: an allocation tree.
begin
S := {v | v is the node that contains the user}
M' := IN_OEDER_OF(M)
V := HPF-BF(S)
repeat
v := FIRST_NODE(V)
V := V - {v}
m := FIRST_SHARE(M')
M' := M' - {m}
allocate multiple-share (or share) m into node v
until M' = ∅
end *SSA*

Function *IN_ORDER_OF(M)*
begin
M' := ∅
repeat
for ∀ m ∈ M **do**
count the number of unallocated share in multiple-share (or share) m
m := {m'| m' is in set M with the largest number of unallocated share}
M' := M'+{m}
M := M - {m}
if all shares have been set to "allocated" **then**
set all shares to "unallocated" **endif**
until M = ∅
return(M')
end *IN_ORDER_OF*

Function ***FIRST_SHARE(M');***
begin
return the first multiple-share (or share) in an order set M'
end *FIRST_SHARE*

Function ***FIRST_NODE(V)***
begin
return the first node in an order set V
end *FIRST_NODE*

3.5 Time Complexity Analysis

The computational complexity of the SSA algorithm is the sum of the run time of determining the allocation order of shares and finding the next location for share to be allocated. First, arranging the allocation order of shares take O((k+n)n) comparisons in the worst case. Second, the HPF-BF search algorithm takes O((e+v)logv) to construct the tree in the worst case. Therefore, the time complexity of algorithm SSA is O((k+n)n)+(e+v)logv), where v denotes the number of vertices, e denotes the number of edges, n denotes the number of single-share, and k denotes the number of multiple-share.

4. ILLUSTRATIVE EXAMPLES AND SIMULATION RESULTS

In this section, some examples and numerical results are employed to illustrate the effectiveness of our approach.

4.1 An Eight-node Communication Network

Consider the network topology in Fig. 7 which consists of eight nodes and eleven communication links. Node 2 contains the user and the other nodes are regarded as share-storage locations. Assume that the probabilities of all nodes and links being operational are set to 0.9.

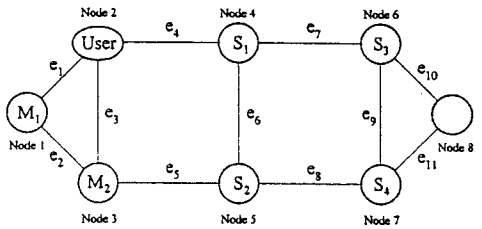

Figure 7. An eight-node communication network

The SSRP of the secret S is evaluated by applying the algorithm presented in Sec. 3.1. In this example, the secret is divided into four shares. Assume that four single-share and two multiple-share are to be allocated. The multiple-share may contain more than one single-share. They are listed as follows.

The set of shares of secret S: $S_s = \{S_1, S_2, S_3, S_4\}$
Four single-share: S_1, S_2, S_3, S_4
Two multiple-share: $M_1 = \{S_1, S_3\}$, $M_2 = \{S_2, S_4\}$

To reconstruct the secret message, the user must access all four divided shares ($m = n$) together. We use HPF-BF search, starting from user node, the root of the tree, to construct the allocation tree. Then we determine the allocation order of shares which is $M_1, M_2, S_1, S_2, S_3, S_4$. A tree that is constructed from a network and allocated

shares into it is called an allocation tree. Figure 8 depicts the allocation tree and SSRP value for this communication network.

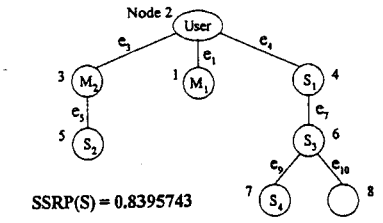

Figure 8. Allocation tree for the eight-node network

Results from our algorithm are compared to those of a Random allocation algorithm. We computed the SSRP of all random distributions. Table 1 summarizes the maximum and minimum SSRP of simulation results based on 5,000 samples.

Table 1: Simulation results of eight-node network

Network topology	SSA algorithm	Random algorithm
Eight-node network	SSRP = 0.8395743	Max. = 0.8395743 Min. = 0.5464987

4.2 The Pacific Basin Network

Figure 9 displays the Pacific Basin network topology, which consists of nineteen sites and twenty-five communication links. Site 7 contains the user and the other sites are considered share-storage locations. Let the probabilities of all nodes and links being operational for the Pacific Basin network be 0.9.

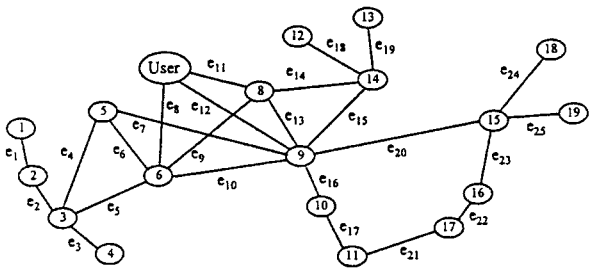

Figure 9. Pacific Basin network

In this example, we divide the secret into seven shares. Assume that seven single-share and four multiple-share are to be allocated, and at least three qualified multiple-share participants can recover the secret. They are listed below.

The set of shares of secret S: $S_s = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7\}$
Seven single-share: $S_1, S_2, S_3, S_4, S_5, S_6, S_7$
Four multiple-share: $M_1 = \{S_2, S_4, S_6, S_7\}$, $M_2 = \{S_1, S_2, S_3\}$, $M_3 = \{S_3, S_4, S_5, S_6\}$, $M_4 = \{S_1, S_5, S_7\}$

The user must access all seven divided shares together to recover the secret. We first use the HPF-BF search to reconstruct the network into a tree topology. Then we obtain the allocation order of shares which is M_1, M_3, M_2, M_4, S_1, S_3, S_5, S_2, S_7, S_4, S_6. Figure 10 depicts the allocation trees for the Pacific Basin network.

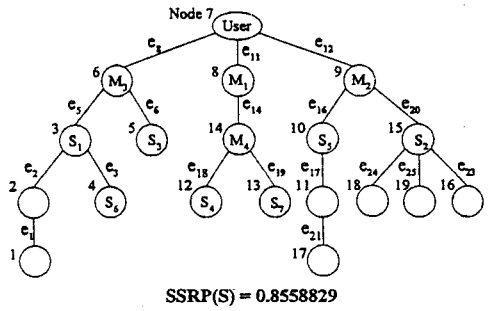

Figure 10. Allocation tree for the Pacific Basin network

Results of our algorithm are compared to those of a Random allocation algorithm. We computed the SSRP of all random distributions. Table 2 summarizes the maximum and minimum SSRP of simulation results based on 5,000 samples.

Table 2: Simulation results of Pacific Basic network

Network topology	SSA algorithm	Random algorithm
Pacific Basic network	SSRP = 0.8558829	Max. = 0.8669108 Min. = 0.3054878

Tables 1 and 2 show that the proposed algorithm can obtain near optimal solutions. Results from our algorithm in the Pacific Basin network indicates a slightly less than the optimal due to the use of the HPF-BF search algorithm. The HPF-BF algorithm can not guarantee that we will obtain the maximal number of MSSTs. Exhaustive approach may be used to find the optimal solution with very expensive price. Previous investigations have demonstrated that exhaustive approach is an NP-hard problem [15,17].

5. CONCLUSION

Rapid expansion of the Internet is fueled by its ability to promote information sharing and to offer high availability. The Internet today is a widespread information infrastructure. As the Internet evolved, one of the major challenges is how to answer the request efficiently and reliably. Information on the Internet can be stored close to its normal point of use, thereby reducing both response times and communication costs. Effectively distributing shares to appropriate cites is the basic consideration for share assignment problem. In this paper, we present a probability model for secret sharing reconstruction and an

algorithm for shares assignment and deal with two types of shares. Illustrative examples demonstrate the underlying concept of the model and the feasibility of our approaches. According to simulation results, the proposed algorithm obtains approximate solutions efficiently. Particularly the (m, n)-threshold scheme is realized and evaluated based on the probability of secret sharing reconstruction.

6. EFERENCES

[1] R. L. Rivest, A. Shamir and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Comm. ACM, vol. 21, no. 2, pp. 120-126, Feb. 1978.

[2] W. P. Lu and M. K. Sundareshan, "Enhanced Protocols for Hierarchical Encryption Key Management for Secure Communication in Internet Environments," IEEE Trans. on Communications, vol. 40, pp. 658-670, 1992.

[3] T. Hwang and W. C. Ku, "Reparable Key Distribution Protocols for Internet Environments," IEEE Trans. on Communications, vol. 43, pp. 1947-1949, 1995.

[4] M. Ito, A. Saito and T. Nishizeki, "Secret Sharing Scheme Realizing General Access Structure," Proc. IEEE Globecom'87, Tokyo, pp. 99-102, 1987.

[5] C. S. Laih, L. Harn, J. Y. Lee and T. Hwang, "Dynamic Threshold Scheme Based on the Definition of Cross-Product in an N-Dimensional Linear Space," Journal of Information Science and Engineering 7, pp. 13-23, 1991.

[6] H. M. Sun and S. P. Shieh, "On Dynamic Threshold Schemes," Information Processing Letters, 52, pp. 201-206, 1994.

[7] A. Shamir, "How to Share a Secret," Comm. ACM, vol. 22, pp. 612-613, June 1979.

[8] G. R. Blakley, "Safeguarding Cryptographic Keys," Proc. of AFIPS 1979 National Computer Conference, New York, vol. 48, pp. 313-317, 1979.

[9] K. K. Aggarwal, S. Rai, "Reliability Evaluation in Computer Communication Networks," IEEE Trans. on Reliability, vol. R-30, pp. 32-35, 1981.

[10] S. Hariri, C. S. Raghavendra, "SYREL: A Symbolic Reliability Algorithm Based on Path and Cutset Methods," IEEE Trans. Computers, vol. 36, pp. 1224-1232, Oct. 1987.

[11] C. S. Raghavendra, V.K.P. Kumar, S. Hariri, "Reliability Analysis in Distributed Systems," IEEE Trans. Computers, vol. 37, pp. 352-358, Mar. 1988.

[12] M. S. Lin and D. J. Chen, "New Reliability Evaluation Algorithms for Distributed Computing Systems," Journal of Information Science and Engineering 8, pp. 353-391, 1992.

[13] D. J. Chen and T. H. Huang, "Reliability Analysis of Distributed Systems Based on a Fast Reliability Algorithm," IEEE Trans. on Parallel and Distributed Systems, vol. 3, no. 2, pp. 139-154, Mar. 1992.

[14] A. Rosenthal, "A Computer Scientist Looks at Reliability Computations," in: Reliability and Fault tree Analysis SLAM, pp. 133-152, 1975.

[15] L. G. Valiant, "The Complexity of Enumeration and Reliability Problems," SIAM J. Computing, vol. 8, pp. 410-421, 1979.

[16] O. R. Theologou and J. G. Carlier, "Factoring and Reductions for Network with Imperfect Vertices," IEEE Trans. on Reliability, vol. 40, pp. 210-217, June 1991.

[17] M. O. Ball, "Computational Complexity of Network Reliability Analysis: An Overview," IEEE Trans. on Reliability, vol. R-35, pp. 230-239, Aug. 1986.