Improving Resilience against DDoS Attack in Unstructured P2P Networks

MA Xin-Xin, ZHAO Yang, QIN Zhi-Guang

School of Computer Science and Engineering, University of Electronic Science and Technology of China
Chengdu 610054 China

Abstract In unstructured peer-to-peer (P2P) systems such as Gnutella, a general routing search algorithm is used to blindly flood a query through network among peers. But unfortunately, malicious nodes could easily make use of the search approach launching distributed denial of service (DDoS) attack which aims at the whole network. In order to alleviate or minimize the bad effect due to behavior of malicious nodes using the flooding search mechanism, the paper proposes a Markov-based evaluation model which exerts the trust and reputation mechanism to computing the level of trustworthy of nodes having the information requested by evaluation of the nodes’ history behavior. Moreover, it can differentiate malicious nodes as early as possible for isolating and controlling the ones’ message transmitted. The simulation results of the algorithm proposed show that it could effectively isolate malicious nodes, and hold back the transmission of vicious messages so that it could enhance tolerance of DDoS based on flooding in Gnutella-like P2P network.

Key words trust; reputation; distributed denial of service (DDoS); peer-to-peer (P2P)

In recent years, as prevalence in the field of peer-to-peer (P2P) computing, collaborative operation, distributed storage and file sharing, P2P has been under intensive research due to its attributions of scalability, robustness, fault tolerance, self-organization and so on. Based on the routing search mechanism and logical topology structure, the current P2P systems are mainly classified into three different architectures: centralized, decentralized but unstructured, and decentralized and structured. In the centralized model such as Napster, central index items are kept in centralized servers which maintain a directory of shared files stored on peers with such form as <object-key, node-address>. Peers obtain index items from central servers and then directly communicate with the peers having inquired contents, and the performance of those servers would be improved. This P2P architecture is very simple and easy to be implemented and maintained. But it has the problem of single point of failure, and the centralized server nature of the service also makes the index server vulnerable to denial of service attacks (DoS); in decentralized but unstructured model such as Gnutella, flooding is the predominant search technique. The search scope is controlled with time-to-life (TTL) parameter so that the query is executed hop-by-hop until success/failure or timeout. Compared with centralized model, this type of model has no single point failure and provides greater freedom for participating users to exchange information or provide services directly each other. However, the blindly flooding search mechanism makes the whole network subjected to distributed denial of service attacks (DDoS)[1-3]; in decentralized and structured model such as Chord, Pastry, Tapestry, and CAN, a query is also distributed but the logical P2P topology is some structured topology. These structured topologies are usually constructed and tightly controlled using distributed hashing table (DHT) techniques. The query is executed hop-by-hop with deterministic hops under ideal case.

The paper focuses on the decentralized but unstructured P2P network such as Gnutella. In this architecture the general routing search mechanism is used to blindly flood a query to the network among peers. That’s, the query is blindly flooded to all its neighbors and these neighbors continue to flood the query to their neighbors if they have no query content. The purpose of using the blind flooding mechanism is to improve probability of hitting the target as many peers as possible throughout whole P2P overlay network. However it must be pointed out that the
design of search mechanism is based on the assumption that all participating peers in P2P systems trust each other implicitly. In a word, all peers are acting altruistic or obedient for common interest, cooperative, and willing to offer resources. While the fact we have to face is that each peer in P2P systems has no responsibility for its behavior. Self-interesting and autonomous characteristics with strategizing to maximize its owner’s network utility have become huge challenge for P2P scalability, robustness and security. Worse than all, malicious peers could easily make use of the query mechanism launching DDoS attack. Be different with the general DDoS attack which uses multiple comprise systems controlled by the vicious node to target a single system, the execution of DDoS attack in P2P system does not need to invade and control server systems in advance, it just modifies the content of flooding messages obtained and then rebroadcasts and the attack target aims at the whole network so that it brings more disaster than the general DDoS attack to network.

In order to minimize and alleviate the deficiency caused by the flooding search mechanism and improve the resistance to the DDoS attack, we propose the Markov-based evaluation (ME) model which exerts the resistance to the DDoS attack, we propose the Markov-based evaluation (ME) model which exerts the resistance to the DDoS attack. In order to minimize and alleviate the deficiency caused by the flooding search mechanism and improve the resistance to the DDoS attack, we propose the Markov-based evaluation (ME) model which exerts the resistance to the DDoS attack. The Descartes set between the set of peers in the network except the peer and the set of all trustworthy peers’ reputations, R makes up of an evaluation of matrix, which represents the predicting of a requesting peer to a special requested peer. Based on these descriptions a process of Markov-based reputation computing is obtained as follows:

\[ E = TR = [T_1, T_2, \ldots, T_n]^T [R_{i1}, R_{i2}, \ldots, R_{in}] \]

here \( i \) mean the requesting peer, and \( j \) denotes the requested peer. While integer set of 1, 2, \ldots, \( n \) means the peers in the network except the peer \( i \) and \( j \). \( T_{ik} \) denotes the trust value of peer \( k \), \( R_{ij} \) means the recommendation of peer \( k \) on peer \( j \). Then we discuss how to compute the value of these elements of the set of \( T \) and \( R \). Thanks to the reputation or trust respectively focuses on one aspect of evaluation at a peer, and the result of these evaluations is based on a material numerical value (e.g. integer). The process of computing result of evaluation or trust is the set of \( X \).

\[ X=[x_0, x_1, \ldots, x_k, \ldots, x_n] \]

here \( x_k \) means the evaluation of \( k \)th step and result equals \( p_0 \). In other words, the set of result of evaluation is \( P \).

\[ P = [p_1, p_2, \ldots, p_{n+1}] \]

Here we make assumption that the evaluation of \( N \) process is made. Now we predict the result of evaluation of \( (N+1) \)th step based on previous \( N \) process outcome obtained. Before the process of computing, we classify four evaluated criteria as follows:

1) If the result of nth process of evaluated peer denotes the peer is honest and the evaluating peer satisfy the result \( p_{n+1} = p_n + A; \)

2) If the result of nth process of evaluated peer denotes the peer is honest but the evaluating peer just partly satisfy the result, \( p_{n+1} = p_n + B \). \( A > B; \)
3) If the result of nth process of evaluated peer denotes the peer made the dishonest behavior but the evaluating peer tolerate the result. $p_{n+1} = p_n - C$.

4) If the result of nth process of evaluated peer denotes the peer makes the dishonest behavior and the evaluating peer doesn’t tolerate the result. $p_{n+1} = p_n - D$. ($C < D$).

In the above four aspects, $A$, $B$, $C$, $D$ are defined the weighted ratings and all these ratings are positive integer. The evaluation result of $(n+1)$th step $x_{n+1} = i_{n+1}$ is just based on nth result of evaluation $x_n = i_n$ and independence of $x_0 = i_0$, $x_1 = i_1$, ..., $x_{n-1} = i_{n-1}$. So, the set of $X$ is a Markov process with discrete time and discrete state, that’s, a discrete homogeneous Markov chain.

In the model proposed, we make an assumption that any peer’s evaluated values is zero before it interacts with other peer for the first time. The mathematical formula $x_0 = i_0 = 0$, and the values of $A$, $B$ are set 10 or 5 respectively which represent the level of the dishonest behavior. The value when the value of upper limit is 90, that’s, a discrete homogeneous Markov chain.

$x_{n+1} = \delta_{i_{n+1}} - x_n = i_{n+1} - i_n$.

Based on the above conditions proposed and the equation, we get

$$p_y = \sum \delta_{i=1}^{\frac{1}{2}} P^0_y i = j, \quad \sum \delta_{i=1}^{\frac{1}{2}} P^0_y = 1, \quad i \in X$$

1) $0 \rightarrow 5$, $0 \rightarrow 10$ and $5 \rightarrow 10$, $5 \rightarrow 15$, under these evaluating conditions, once the anticipant peer acts under the above third or forth behavior, it will immediately exclude. So we suppose that the anticipant peer only acts under the first or second behavior neither do under the third or forth behavior. Moreover, the event of anticipant peer’s behavior under the first or second is equiprobable.

$$P_{0,5} = P_{0,10} = P_{5,10} = P_{5,15} = \frac{1}{2}$$

2) $i \in \{10, 15, 20, 25\}$, the anticipant peer not only acts variety behavior but also acts the above the third condition. Similarly, the event is equiprobable.

$$P_{i+5} = P_{i,10} = P_{i,15} = \frac{1}{2}$$

3) $i \in \{30, 35, 40, 80\}$, the anticipant peer can act all above four. Similarly, the event is equiprobable.

$$P_{i,30} = P_{i,35} = P_{i,40} = P_{i,80} = \frac{1}{4}$$

4) $i \in \{85, 90\}$, because the value of upper limit of $i_{n+1}$ is 90

$$P_{85} = P_{85,80} = P_{85,90} = P_{90} = \frac{1}{4}$$

and

$$P_{85,90} = P_{90} = \frac{1}{2}$$

These equations express that the probabilities above third and forth behavior are integrated completely into the transitive probability matrix $P$.

Based on the equation of Kolmogorov-Chapman,

$$P_{y}^{m+n} = \sum \delta_{i=1}^{\frac{1}{2}} P_{y}^{m-n}$$

Because $\forall i \in X, \sum P_{i}=1$, and $X$ is a closed set.

So according the above conditions proposed and the equation, we get

$$P_{y}^{(m+n)} = \sum \delta_{i=1}^{\frac{1}{2}} P_{y}^{m} i = j, \quad \sum \delta_{i=1}^{\frac{1}{2}} P_{y}^{m} = 1, \quad i \in X$$

Here,

$$P_{y}^{(0)} = \delta_{y} = \begin{cases} 1 & i = j, \\ 0 & i \neq j \end{cases}, \quad P_{y}^{(1)} = P_{y}$$

Upon the above conditions and the equation we proposed, the result of $p_y(n)$ is obtain, and the value of $p_y(n)$ is limited the range of $[0, 1]$. 

2 Simulation Methodology

2.1 Simulation Setup

The simulation of the proposed approach in the paper has been implemented in the PeerSim simulation environment. In this section, we describe how to
construct simulation setup, parameter settings and hops of message transmission controlled by TTL. In order to simulate the real P2P network, all peers are designated artificially to share amount of files information in our simulation, and every peer chooses a few of files which are not stored in its local area to transmit to network within a fixed interval. Upon receiving a query message, a node checks if any locally stored information match the query. If so, the node sends a query response back towards the query originator and the query will not be further forwarded again from this responding peer. A query message will also be dropped if the query message has visited the peer before. But if the peer is bad one, it will tamper the message with no destination arriving at and then retransmit message to the network. In our simulation, there are 2000 nodes altogether, and every neighbor is setup 3 and TTL is assigned to 3. The max messages numbers which every node can process in one cycle is 300. That’s to say, the whole network capability of messages numbers is 600 million. If each peer acts as good behavior the curve of message rises smoothly, but the curve would ascend sharply along with the increase of the number of bad peers. This shows that the resource of network is used up rapidly.

To test the degree of damage produced by collusive malicious peers and evaluate our ME algorithm resisting DDoS attack aiming at the whole network, we propose a simulation setup where once a peer joins the network, it would not leave the network, hence, the network composed by these peers is a static network.

2.2 Simulation Results Analysis

The simulation results are shown in Fig.1, Fig.2 and Fig.3. Fig.1 shows the diagram of consume network resources of malicious peers’ collusion. X axis presents the malicious message numbers in a cycle and Y axis does the percentage of malicious message numbers produced by malicious peers in different cycle. In Fig.1, there are five curves mean variety of network resources consume when the percentage of malicious peers among peers in networks is 20, 30, 40, 50, and 60 separately. As shown in Fig.1, the change of these curves accord with power law. Moreover, with the number of the collusive malicious peers increasing, the consuming of network resources also appeared a trend of augmentation sharply so that network resources would be used up in short cycle time. E.g. in the diagram, the curve which presents the percentage of malicious peers’ is 50 may reach 80% of the whole network capability within only 4 cycle time. It shows that the malicious peers make use of the flooding inquire mechanism launching easily DDoS attacks to the whole network in Gnutella-like P2P network, The damage of this attack is huge. Even worse, the network itself had no valid defending measure to this type of attack.

Fig.2 shows the diagram of consume network resources of malicious peers’ collusion using our Markov-based evaluation algorithm. In this diagram, there’re also five similar curves to Fig.1. However, be difference with the latter, the consuming of network resources exerted by malicious peers’ collusion falls distinctively. E.g. when the ratio of malicious peers are within 20 through 40 per, the numbers of malicious peers’ messages just occupy about 10% of all network message numbers, and it is decreased by 400%~800% compared with bad messages shown in Fig.1. Even when the ratio of malicious peer numbers is up to 50%~60%, the number of malicious messages is also reduced by 100%.

Fig.3 shows the comparison of malicious message numbers in network in histogram. As shown in Fig.3, when the percentage of malicious peers in the simulation environment are 20, 30 and 40 separately, the malicious message numbers using ME algorithm fall sharply, that is to say, the malicious peers are identified and isolated, and the bad messages from these malicious ones are also controlled with local area. Even when the percentage of malicious peers in the simulation environment are up to 60, our ME algorithm is also reduced the number of bad messages to 50%. It shows that the Markov-Based evaluation algorithm proposed could efficiently improve resilience against DDoS attacks based on flooding in Gnutella-like P2P network.
flooding inquire and notify its’ other neighbors to isolate the malicious peer. Moreover, it could effectively reduce the number of malicious messages down to 50 per, and the average cost of each query to reach the same scope of nodes is reduced by about 65 percent when using our proposed ME algorithm in a Gnutella-like P2P network.

References


3 Conclusions

In this paper, we propose a Markov-Based evaluation algorithm which exerts the trust and reputation mechanism to enhancing resilience against DDoS attack in Gnutella-like P2P Networks. Our simulation results show that the proposed evaluation algorithm could make a peer distinguish neighbor malicious peers from its’ neighbor peer launching flooding inquire and notify its’ other neighbors to isolate the malicious peer. Moreover, it could effectively reduce the number of malicious messages down to 50 per, and the average cost of each query to reach the same scope of nodes is reduced by about 65 percent when using our proposed ME algorithm in a Gnutella-like P2P network.

Brief Introduction to Author(s)

MA Xin-xin (马新新) was born in Xian, Shannxi Province, China, 1975. He is now a candidate for doctorate in the School of Computer Science and Engineering of University of Electronic Science and Technology of China (UESTC). His

(Continued on page 28)