JACET Journal of Advances in Computer Engineering and Technology, 1(2) 2015

A Genetic Programming-based trust model for P2P Network

Mahdi Sattarivand[1]

Received (2015-02-23)
Accepted (2015-04-05)

Abstract ***- Peer-to-Peer systems have been the center of attention in recent years due to their advantage. Since each node in such networks can act both as a service provider and as a client, they are subject to different attacks. Therefore it is vital to manage confidence for these vulnerable environments in order to eliminate unsafe peers. This paper investigates the use of genetic programing for achieving trust of a peer without central monitoring. A model of confidence management is proposed here in which every peer ranks other peers according to calculated local confidence based on recommendations and previous interactions. The results show that this model identifies malicious nodes without the use of a central supervisor or overall confidence value and thus the system functions.***

Index Terms - peer-to-peer systems, confidence, genetic programing, malicious nodes.

I. INTRODUCTION

IN the past decade with the growth of Peer-to-Peer (P2P) systems, malicious activities have turned in to an important security issue in these networks. Due to the openness of P2P systems, unsafe users may occupy a great part of the population of P2P system. Confidence management in such an open area is a problem and an important subject. The aim of the confidence models is generally to eliminate unsafe peers. Although maintaining communication by recognizing malicious peers from safe ones without prior knowledge is a difficult task. Therefor most of the proposed models in this research offer an approximate decision-making guideline about peers.

Confidence management can be carried out by a central supervisor like eBay. But a central supervisor is not compatible with a P2P environment. Peers must prepare themselves for management and data storage about their safe connections [1, 2, 3]. In pure P2Ps like Gnutella, peers send great volumes of confidence queries to the network in order to gain confidence information about others [4]. In this model all the peers save confidence data about their neighbors according to previous interactions [2, 5, 6]. Queries are able to gather recommendations about asked peers and decide accordingly. Some models use distributed hash tables (DHT) to save information. Every peer saves confidence data about other specified peers using a DHT algorithm which enables them to effectively access information [1, 3, 7]. In this way peers can access information of overall confidence about other nodes without sending so many queries about the overall network.

[1] School of Electrical and Computer Engineering, Guilan University, Guilan, Iran (sattarivand.it@gmail.com)

Confidence management in P2P systems is a problem due to the absence of a central supervisor and unsafe gathered data from peers. Confidence models must be able to detect the complex behavioral pattern of unsafe peers and make a smart decision for recognizing them from safe nodes. Using artificial intelligence techniques can be a useful method for a complex subject as this one.

Nowadays artificial intelligence has been used in many sciences, e.g., to recognize the quality of a football goal-keeper and football talent fuzzy expert systems and A Novel Fuzzy Approach for Determining Best Position of Soccer Players [8, 9] Also, the fuzzy theory based on the uncertainty were used in geographic information systems [10]. AI classification algorithms were used in medicine and psychology [11], and its important applications in the field of bioinformatics [12] or use expert systems in risk managements [13].

In this paper a genetic program based on confidence management model is proposed. This model utilizes features extracted from peers to detect malicious peers from safe ones. Peers save previous interactions with other peers and gather recommendations about neighboring peers. Two types of information are gathered by peers: interactions and recommendations, which are the ground elements of features. A confidence model including these features and using genetic programing to measure the level of confidence of peers is proposed. The rest of this research is continued as follows: in the second part the proposed confidence model is presented. In part three and four the results of simulation and conclusion are presented.

II. THE PROPOSED CONFIDENCE MODEL

In The proposed model in this paper utilizes genetic programing to make a confidence decision. Genetic programing is an evolutionary calculation technique which was offered by Koza for machine learning community [14].

In genetic programing functions include operators, phrases, etc. and terminals include features and constants build up a genetic programing (GP) tree. Every GP tree is unique. A group of these unique trees which are possible solutions to a problem are produced by GP of every generation.

1. A. Operators and Features

The Choosing a set of proper features is a difficult subject and also the key point for achieving successful results in GP and other learning machine techniques [9]. In our model the gathered data from previous interactions and recommendations of neighbors build the set of features. Interactions are gained through previous experience of peers with other peers. These experiences have happened directly among two peers that have interacted before and can include any special activity in P2P networks like sharing files sharing CPU or memory. The table below shows features based on interactions.

TABLE 1
INTERACTION-BASED FEATURES

Feature	Symbol
No. of interactions	F1
No. of successful interactions	F2
Average size of the downloaded files	F3
Mean difference of the last two interactions	F4
Average weight	F5
Average satisfaction	F6

The satisfaction and weight parameters are calculated like [16]. Successful interactions are the ones in which a file download is finished successfully. Satisfaction is calculated according to average bandwidth, agreed bandwidth before interaction and the size of the online and offline periods of loader:

$$S= \begin{cases} (\frac{AvB}{AgB}+\frac{On}{On+Off})/2 & \text{if AvB > AgB} \\ (1+\frac{On}{On+Off})/2 & \text{Otherwise} \end{cases} \qquad (1)$$

In which AvB is average bandwidth, AgB is agreed bandwidth and on and off show periods of online and offline status of the loader respectively.

Weight is calculated according to file size, the number of downloaded file loaders and maximum number of files loaded by loaders.

$$W=(\frac{s}{100MB}+\frac{NU}{U_{\max}})/2 \quad \text{if s< 100 MB}$$

$$=(1+\frac{NU}{U_{\max}})/2 \quad \text{otherwise} \qquad (2)$$

In which s is the file size, NU is number of file loaders and Umax is the maximum amount of uploaded files.

The second set of features is based on recommendations. When a peer wants to interact with another he asks his neighbors about their experience with that peer. The neighbors which have an experience of interaction with that peer send their recommendations. Actually the experience about other nodes is called recommendation. A recommendation includes the following information: average number of successful transactions, the average satisfaction of the interactions, the average weight of the transactions, and the amount of calculated confidence for the specific peer. The features based on recommendation are shown in the table below.

TABLE 2
RECOMMENDATION-BASED FEATURES

Feature	Symbol
Number of recommendations	F7
Average number of neighbors' successful transactions	F8
Average satisfaction of the neighbors	F9
Average weight of the neighbors	F10
Average confidence values	F11

In the specified genetic model, simple operators are used to produce a formula for confidence calculations. The used operators include addition, subtraction, division, multiplication, inverse, logarithm, square root and square:

B. Compatibility Function

Compatibility function is one of the most important factors affecting the performance of evolutionary computation techniques. It determines how well a program can function in solving a problem [8.11]. In the developed confidence model, compatibility function is used as reduced number of attacks. In other words if N_{att}^{Tr} shows the number of attacks with the confidence model and N_{att}^{NoTr} shows the number of attacks without the confidence model then the compatibility function would be as follows:

$$F = \frac{N_{att}^{NoTr}}{N_{att}^{Tr}} \qquad (3)$$

If the produced trees can reduce the number of attacks, the amount of compatibility function reduces and the success of the model increases. So in the genetic model, the aim is minimizing the compatibility function. At the end of evolution, the most successful tree is the one chosen as the solution. GP algorithm is shown below:

```
Initial determination of population
While (present population < = maximum generation
}      for( all trees in the present generation)
          {simulation run;
{  ; evaluating compatibility function
; genetic operators run
creating a new population ; }
```

Although malicious peers may download unidentified files or receive unjust recommendation to harm the system, the goal of a confidence management model is to reduce unsafe and injected files or unjust recommendations.

In this confidence model, nodes are supposed to show two types of behavior: Naïve or hypocritical.

Naïve: Attackers usually download unidentified or injected virus files and gives an unjust recommendation to other nodes [18].

Hypocritical: Attackers carry out attacks by downloading unidentified files or unjust recommendations with x% probability. In other words they act as good peers [3, 5].

III. RESULTS OF SIMULATION

In the test, the model is initially tested for all kinds of attackers. Testing with network settings is carried out with 10 percent malicious peers, the best result out of ten for each attack is chosen. Then the model is tested with 10, 30 and 50 percent malicious peers in the network. The probability of hypocritical peers in considered 20% in all interactions. If a peer downloads an unidentified or a virus file it is considered an attack on the file. Simulation is first carried out with the confidence model.

Then simulation is carried out for the developed confidence model, the success of the confidence model in preventing attacks is emphasized with this model. Table 3 shows the rates of success of the confidence model against personal attacks to file-based attacks.

TABLE 3
CONFIDENCE MODEL SUCCESS RATE

	10%	30%	50%
Naïve	83.8	78.9	73.6
Hypocritical	71.8	57.7	47.1

As the table shows, the model has achieved noticeable success against Naïve attackers. That is why identifying naïve attackers become easy after first interaction.

Our model shows a good rate of success for hypocritical attackers, in a network with 10% malicious peers this amount is 71.8% and in a network with 50% malicious peers the model can prevent half of the attacks. In a network with this amount of malicious nodes, this is good rate of success.

Speed of convergence of the confidence model is an important item in recognizing an attack in a reasonable time. Figure 1 shows how the number of attacks by naïve and hypocritical attackers has reduced.

Figure 2 shows reduce in attacks based on recommendation over time.

In the second part of the simulation, simulation is carried out for collaborative malicious peers. These peers act as a team to create a good recommendation for each other.

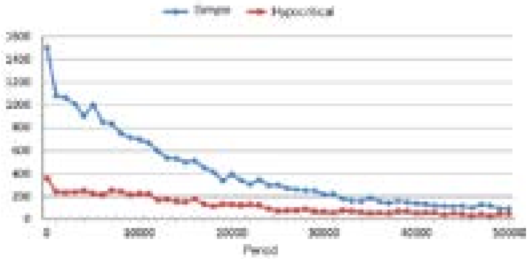

Fig. 1. Attacks based on file

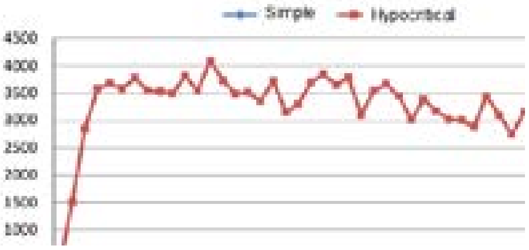

Fig. 2. Reduce in attacks based on recommendation

Table 4 presents rates of success for this state which shows a reduce in success for both naïve and hypocritical peers ith respect to the previous state.

TABLE 4
RATE OF SUCCESS FOR COLLABORATIVE MALICIOUS PEERS

	10%	30%	50%
Naïve	79.3	75.1	71.9
Hypocritical	61.7	46.3	39.5

Figure 3 depicts the number of attacks based on file over time for a network with 10 percent collaborative mallicious peers. The model reduces the number of effective attacks by naïve and hypocritical peers.

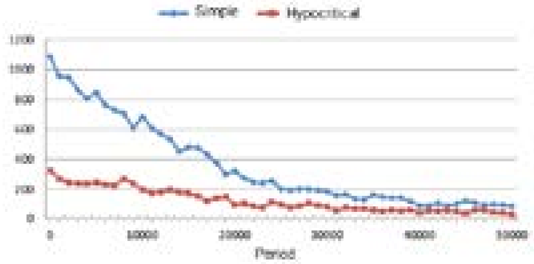

Fig. 3. File-based attacks with collaborative malicious peers

Figure4, shows attacks based on recommendation which are carried out by fellow attackers. Cooperation between malicious peers increases unjust recommendations.

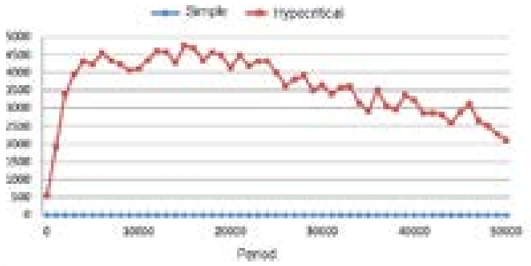

Fig. 4.attacks based on fellow peers' recommendations

IV. CONCLUSIONS

This paper presented a developed model of confidence using genetic programing; rates of confidence for peers have been calculated by a formula produced by this model. Experimental results state that the model is able to differentiate between malicious and pure peers. Naïve and hypocritical attackers were also studied separately and the results showed that the model is considerably successful for naïve attackers and fairly good for hypocritical ones.

The developed model proved that genetic programing can be used in Peer-to-Peer networks for building a confidence model.

REFERENCES

[1] Kang, Xin, and Yong dong Wu. "A trust-based pollution attack prevention scheme in peer-to-peer streaming networks." Computer Networks 72 (2014): 62-73.

[2] Hendrikx, Ferry, Kris Bubendorfer, and Ryan Chard. "Reputation systems: A survey and taxonomy." Journal of Parallel and Distributed Computing 75 (2015): 184-197.

[3] England, Philip, et al. "A Survey of Trust Management in Mobile Ad-Hoc Networks." Proceedings of the 13th annual post graduate symposium on the convergence of telecommunications, networking, and broadcasting, PGNET. 2012.

[4] Qi, Xiaoguang, et al. "A complex network model based on the Gnutella protocol." Physica A: Statistical Mechanics and its Applications 388.18 (2009): 3955-3960.

[5] Fan, Xinxin, et al. "Behavior-based reputation management in P2P file-sharing networks." Journal of Computer and System Sciences 78.6 (2012): 1737-1750.

[6] Zhou, Runfang, Kai Hwang, and Min Cai. "Gossiptrust for fast reputation aggregation in peer-to-peer networks." Knowledge and Data Engineering, IEEE Transactions on 20.9 (2008): 1282-1295.

[7] Sattarivand, Mahdi, Reza Ebrahimi Atani, and Yashar Fekri Sustani. "Anew METHOD FOR SECURITY IN PEER TO PEER NETWORK USING MODIFIED PARTICLE SWARM ALGORITHM." Chemical and Process Engineering)2014(, 435-441.

[8] Bazmara, Mohammad, Shahram Jafari, and Fatemeh Pasand. "A Fuzzy expert system for goalkeeper quality recognition." arXiv preprint arXiv:1309.6433 (2013).

[9] Mohammad Bazmara,"A Novel Fuzzy Approach for Determining Best Position of Soccer Players", IJISA, vol.6, no.9, pp.62-67, 2014. DOI: 10.5815/ijisa.2014.09.08

[10] Mohammadi, Fereshteh, and Mohammad Bazmara. "A New Approach of Fuzzy Theory with Uncertainties in Geographic Information Systems." International Journal of Mechatronics, Electrical and Computer Technology 3.6 (2013): 1001-1014.

[11] M. Bazmara, S. Vahedian and S. Ramadhani, "KNN Algorithm for Consulting Behavioral Disorders in Children," Journal of Basic and Applied Scientific Research, 3(12) 2013, pp. 981-986

[12] Pal, Sankar K. "Computational theory perception (CTP), rough-fuzzy uncertainty analysis and mining in bioinformatics and web intelligence: a unified framework." Transactions on Rough Sets XI. Springer Berlin Heidelberg, 2010. 106-129.

[13] Fereshteh Mohammadi, Mohammad bazmara, Hatef Pouryekta,"A New Hybrid Method for Risk Management in Expert Systems", IJISA, vol.6, no.7, pp.60-65, 2014. DOI: 10.5815/ijisa.2014.07.08

[14] Koza, John R. "Survey of genetic algorithms and genetic programming." Wescon Conference Record. WESTERN PERIODICALS COMPANY, 1995.

[15] Hall, Mark A. Correlation-based feature selection for machine learning. Diss. The University of Waikato, 1999.

[16] Wang, Li. "SoFA: An expert-driven, self-

organization peer-to-peer semantic communities for network resource management." Expert Systems with Applications 38.1 (2011): 94-105.

[17] Cramer, Nichael Lynn. "A representation for the adaptive generation of simple sequential programs."Proceedings of the First International Conference on Genetic Algorithms. 1985.

[18] Dellarocas, Chrysanthos. "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior." Proceedings of the 2nd ACM conference on Electronic commerce. ACM, 2000.