

Attacks against Network Coordinate System: Vulnerable PIC

Xiaohan Zhao, Xiaoxiao Song, Xiao Wang, Yang Chen, Beixing Deng, Xing Li
Tsinghua National Laboratory for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, P.R. China
homeisland03@gmail.com

Abstract

In recent years, network coordinate systems which map nodes into a geometrical space can effectively support overlay applications relying on topology-awareness. However, these systems base on an ideal assumption that the nodes in them are honest to cooperate with each other. Although there have been some studies about attacks on network coordinate systems, the effect of attacks on PIC---one of the representative systems---has not been studied. Moreover, since PIC itself has proposed a security policy, how well it can protect PIC from attacks is another significant problem to be researched. We apply four typical attacks on PIC with security and without security. Our extensive experiments show that PIC is vulnerable by attacks and when the percentage of malicious nodes is more than 40%, PIC with security performs barely better than without security.

1. Introduction

In recent years, network coordinate systems, which can accurately estimate network delay or round trip time (RTT) with low overhead, have been proposed to support the application overlays that benefit from topology-awareness. Up to now, there are several widely studied network coordinate systems: GNP [1], NPS [2], PIC [3], Vivaldi [4] and Pharos [5], where GNP is a centralized coordinate system while the rest are decentralized.

However, since a network coordinate system must run a long time and it is consisted of a large number of nodes, it would attract hackers to attack the system. Moreover, decentralized network coordinate systems [2-5] ideally assume that nodes in them act honestly, in other words, a node in such system will respond other nodes with its true information, such as its coordinates, the RTT between the probing node and itself and so on. But such assumption cannot be achieved in real network and it would make network coordinate systems vulnerable to malicious attacks.

Although it has been verified by Kaafar et al. that Vivaldi and NPS are sensible to attacks from inside malicious nodes [6][11], for PIC, an representative network coordinate system which has been applied in an important P2P system---Pastry [7], the effect of attacks on it has not been studied yet, which leads to the study of attacks on network coordinate systems incomplete. Furthermore, since a security policy has been devised in PIC, which relies on triangle inequality to detect malicious nodes, whether the policy can secure PIC efficiently or not is another significant problem.

In this paper, we study the performance of attacks on PIC and the efficiency of its security policy. We apply four types of attack on PIC, which require fewer restrictions but are more typical than the attack proposed in [3]. The results show that when the percentage of malicious nodes is larger than 40%, PIC with security performs barely better than without security and even worse in Colluding Isolate Attack.

The rest of the paper is organized as follows. Section 2 presents an overview of PIC and classification of attacks. How the simulation is set up and the results of experiment are demonstrated in section 3. Finally, in section 4, we conclude this paper.

2. Related Work

2.1. PIC Overview

PIC is a decentralized coordinate system. Because there are no fixed nodes called infrastructure nodes to assist coordinate computation and every node can be chosen by other nodes as their landmark, which is used to compute the coordinates of other nodes.

In PIC, each node selects L nodes as its landmarks to compute its d -dimensional coordinates ($L > d$) in Euclidean space. When the number of nodes already in the system N is smaller than the number of landmarks L , the joining node selects all nodes as its landmarks, collects measured distances between all pairs of these nodes, which constructs a $N \times N$ matrix and uses a

global optimization algorithm to compute a new set of coordinates for all nodes.

When nodes in the system are more than L , a different method is used to compute coordinates: the joining node chooses L landmarks from all the already-in nodes and after getting information from its L landmarks, it computes its corresponding coordinates by exploiting an optimization algorithm to minimize the error between the predicted distance and the measured distance. The target error function is the sum of the squares of the relative errors:

$$e = \sum_{i=1}^{|L|} \left(\frac{d_i^m - d_i^p}{d_i^m} \right)^2$$

Where d_i^m is the measured distance between the joining node and its i th landmark and d_i^p is the predicted distance between them.

To select landmarks, the author of [3] presents three strategies: 1) landmarks are chosen randomly; 2) only the closest nodes can be selected as landmarks for the joining node; 3) some of landmarks are picked randomly while the rest are picked from the closest ones. The results in [3] show the third strategy performs best and it is used in this paper.

Considering the existence of malicious nodes, PIC proposes a security policy based on triangle inequality. The main idea is that each node uses triangle inequality to verify every landmark and rejects the one which violates the triangle inequality mostly. Specifically, after a node n receives information from its landmarks, it computes the two metrics below for every landmark i :

$$upper_i = \sum_{j=1}^{|L|} \begin{cases} d_i^m - (d_j^m + d_{i,j}^p), & \text{if } (d_j^m + d_{i,j}^p) < d_i^m \\ 0, & \text{otherwise} \end{cases}$$

$$lower_i = \sum_{j=1}^{|L|} \begin{cases} (d_j^m - d_{i,j}^p) - d_i^m, & \text{if } (d_j^m - d_{i,j}^p) > d_i^m \\ 0, & \text{otherwise} \end{cases}$$

Where d_i^m is the measured distance between node n and landmark i , so is d_j^m . And $d_{i,j}^p$ is the predicted distance between landmark i and landmark j . Then, the maximum values of both metrics are found out, the corresponding node is eliminated and then the joining node computes its coordinates with the remaining landmarks until the above process is repeated for required times or the average relative error between the joining node and the remaining landmarks exceeds the settled threshold.

As mentioned in [3], PIC has been used in Pastry [7], a generic, scalable and efficient substrate for peer-to-peer applications. The usage of PIC in Pastry can largely reduce the control traffic. Studies in [3] also

show that PIC works well in a churn environment where nodes join and leave the overlay continuously. In other words, PIC has demonstrated its usefulness in Proximity-aware P2P overlays.

2.2. Attack Classification

[6] classifies attacks on network coordinate systems into four classes:

(1) Isolation: Malicious nodes select several nodes as their targets and then inveigle themselves into a remote area. These targets seem to be isolated from other nodes so that they would probably choose malicious nodes as their neighbors because the malicious ones are their closest nodes in the remote zone. Thus, malicious nodes can play tricks on these targets.

(2) Repulsion: In order to reduce the consumption of its resources, e.g. bandwidth, a malicious node provides other nodes with false information either by forging coordinates or delaying the probes to pretend its position is rather far away.

(3) Disorder: The aim of this attack is to cause high error or even non-convergence in coordinate systems. In order to realize the attack, malicious nodes provide fake information to others.

(4) System control: In this attack, malicious nodes try to be in higher hierarchy to influence as many nodes as possible.

3. Performance Evaluation

3.1. Performance Metric

We use the mean of *average relative error* as the performance metric. In each computation, average relative error can reveal the accuracy of coordinate computation of all the nodes in the system. However, there are many random factors to influence the accuracy so that the average relative error of each computation fluctuates. In order to better evaluate the performance of the system, we run the computation for several times and compute the mean of average relative error to eliminate random factors. The smaller the mean of average relative error is, the more accurate computation is. Average relative error is computed using the first equation as follow and the mean of it uses the second one.

$$\bar{e} = \frac{\left(\sum_i \sum_{j(i \neq j)} \frac{d_{i,j}^m - d_{i,j}^p}{d_{i,j}^m} \right)}{M}$$

$$E = E(\bar{e}) = \frac{\sum_{i=1}^N \bar{e}_i}{N}$$

3.2. Experiment Set Up

We used two kinds of data which were collected from real Internet. One is the “King” data that contains measured RTT between any two nodes of Internet 1740 DNS servers using the King method [8]; the other one is a data of measured RTT between 226 nodes of PlanetLab [9].

We developed the PIC simulator based on the description of [3]. In the simulator, nodes were mapped into a 7-dementional Euclidean space, each node had 16 landmarks, of which 4 landmarks were the closest ones, and Simplex Downhill [10] was exploited as the optimization algorithm. For the security policy, the threshold of the relative error mentioned in 2.1 was set to 5% and the repeat time of security policy was set to 5 times. When the security policy is used in PIC, we call it security on; otherwise, it is called security off.

We applied four types of attack: Random Attack, Fixed Point Attack, Colluding Isolate Attack and Combined Attack. In each attack, we repeated 20 times to carry out our experiment by randomly choosing 0%, 10%, 20%, 30%, 40%, 50%, 60% and 70% nodes from all nodes as malicious nodes except the first 16 joined nodes called basic landmarks. Specifically, each attack begins when the number of joined nodes is larger than the required number of basic landmarks.

3.3. Experimental Result

3.3.1. Random Attack

Random Attack is a simple attack implemented by malicious nodes independently, which can be classified into disorder attack mentioned in 2.2. In this attack, malicious nodes have no special aims except causing high computation error in the system. When malicious nodes are chosen as landmarks by other nodes, they will generate random coordinates and inflate measured distances between them and their victims. In specific, every dimension of random coordinates is in [-250,250] and the inflated distances are 1.5 times more than the true ones.

Fig. 1 and 2 show the mean of average relative error on PlanetLab data and King data when there are different percentage of malicious nodes in PIC. From them, we observe that when the number of malicious nodes increases, the mean of average relative error rises, which is consistent with our intuition. When

malicious nodes are more than 20% in both Fig.1 and 2, the error is larger than 1, while the error is less than 0.2 of PIC without malicious node. In other word, Random Attack of malicious nodes would result in high computation errors.

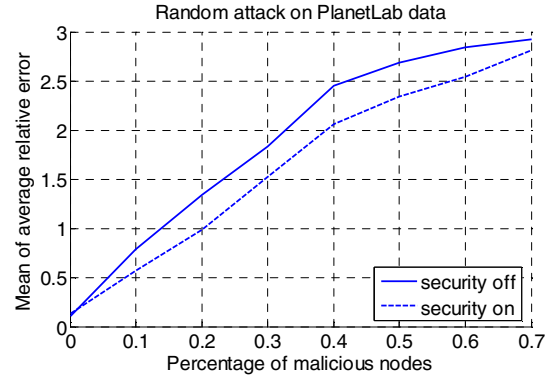


Fig.1. Mean of Average Relative Error of Random Attack on PlanetLab data

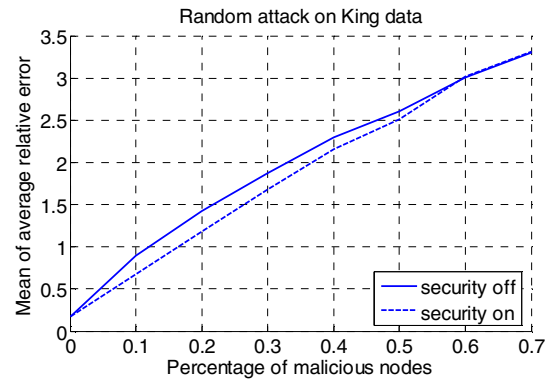


Fig.2. Mean of Average Relative Error of Random Attack on King data

Comparing PIC with security and without security, we find that the security policy can protect PIC from Random Attack. However, when more than 40% malicious nodes present in the system, the difference of the two curves gets smaller with the increase of malicious nodes. Especially in Fig. 2, when there are more than 50% malicious nodes in the system, the accuracy of PIC with security is almost the same as without security. The probable reason for this is that although malicious nodes only attack the nodes choosing them as landmarks, the high computation errors caused by their attack can propagate throughout the whole system. When the number of malicious nodes is less than 40%, their action is distinguished from other nodes and the security policy can detect them easily. So the difference between PIC with security and without security is obvious. However, the increase of malicious nodes leads to more chaos in the

system and causes more honest nodes' coordinates inaccurate, which results in that the failure rate in distinguishing malicious and honest ones increases. Thus, when the number of malicious nodes increases in the system, the security policy becomes less effective.

3.3.2. Fixed Point Attack

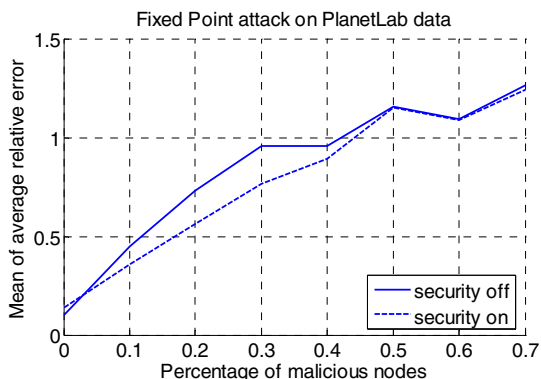


Fig.3. Mean of Average Relative Error of Fixed Point Attack on PlanetLab data

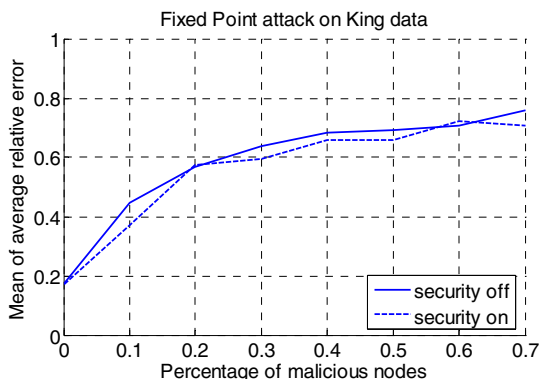


Fig.4. Mean of Average Relative Error of Fixed Point Attack on King data

In this attack, malicious nodes consult a fixed point as their coordinates before attack and they don't communicate with each other after joining the system. If they are selected as landmarks by other nodes, they will inform the nodes with the fixed coordinates and the real distances between them. In our experiment, we set each dimension of the fixed point to 0.

Fig. 3 presents the results of Fixed Point Attack on PlanetLab data and Fig. 4 depicts the results of King data. Comparing the mean of average relative error when there are malicious nodes in PIC with that of PIC without malicious nodes, we find that Fixed Point Attack causes higher error in coordinate computation. Meanwhile, the error increases with the growth of malicious nodes. In Fig.3, when more than 40%

malicious nodes attack PIC, the mean of average error gets larger than 1.

Moreover, in Fig. 3, when there are more than 50% malicious nodes, the two curves are almost the same. In Fig. 4, the performance of PIC with security is barely better than without security. Even when the number of malicious nodes is 60%, the performance of PIC with security is a little worse than without security. Both of the figures show the security policy cannot protect PIC well.

3.3.3. Colluding Isolate Attack

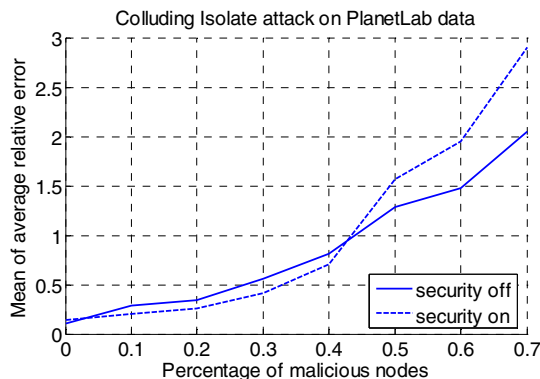


Fig.5. Mean of Average Relative Error of Colluding Isolate Attack on PlanetLab data

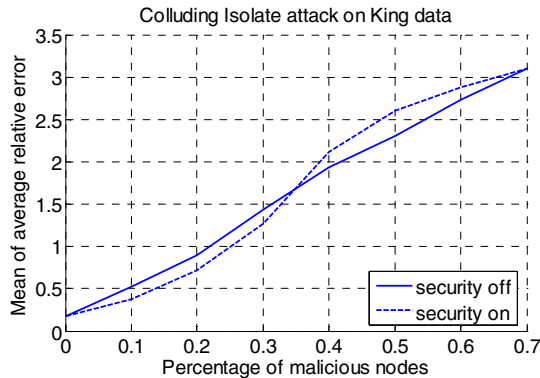


Fig.6. Mean of Average Relative Error of Colluding Isolate Attack on King data

In [6], there are two methods to realize Colluding Isolate Attack: one is that malicious nodes move all honest nodes away from the target nodes; the other one is that malicious nodes attract targets into a specific area. We utilize the latter one in this paper: malicious nodes try to inveigle the victims to a settled point in a remote area, each dimension of which is 230. In real network, nodes can delay probes to augment measured distances. Thus, when a node requests information from a malicious node, the malicious node would compute the predicted distance between it and the

settled point by using the coordinates. If the predicted distance is less than the real distance between the malicious node and its victim, the malicious node will compute new coordinates to make the predicted distance larger than the real one. Otherwise, it will compute coordinates to get close to the settled point but keep the predicted distance larger than the real one. In both computations, the coordinates change along the line between the malicious node and the settled point. Finally, the malicious node would send the victim its elaborately computed coordinates and the predicted distance between it and the settled point.

Fig.5 and 6 show that the mean of average relative error rises when the number of malicious nodes increases, which is similar with the results of Random Attack and Fixed Point Attack. We also observe that Colluding Isolate Attack is powerful in causing high computation errors. When there are more than 40% malicious nodes in PIC on both PlanetLab data and King data, the error is higher than 1, which is at least 5 times more than that of PIC without malicious nodes. That is, PIC is vulnerable by Colluding Isolate Attack.

Meanwhile, the significant results show that either on PlanetLab data or on King data, Colluding Isolate Attack can confuse the security policy successfully because this attack attempts to accord with the triangle inequality to avoid being detected. From Fig. 5 and 6, we find that PIC with security performs even worse than PIC without security when there are more than 40% malicious nodes in the system. Indeed, when less than 40% malicious nodes attend in PIC, although the security policy can protect the system from the attack, the difference between the two curves in each figure is less than 0.2, which is rather small. When malicious nodes are more than 40%, the capability of the security policy becomes worse---Colluding Isolate Attack on PIC with security results in higher error than without security. Thus, Colluding Isolate Attack is not only powerful in leading to high computation error in PIC but also efficacious to defeat the security policy based on triangle inequality.

3.3.4. Combined Attack

Previously, we study the effect of three attacks on PIC with different percentage of malicious nodes. In Internet, since there are many hackers to attack the system with different intentions, there would be more than one type of attack. The most practical attack should be consisted of different attacks, which we call Combined Attack. In our experiment, we combine three attacks above into this attack and fairly separate malicious nodes into three parts to carry out different attacks. The parameters of each attack are the same as above.

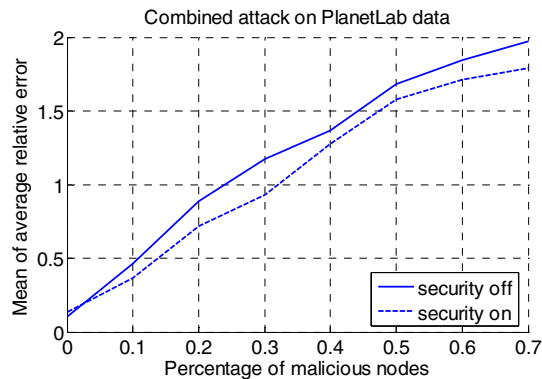


Fig.7. Mean of Average Relative Error of Combined Attack on PlanetLab data

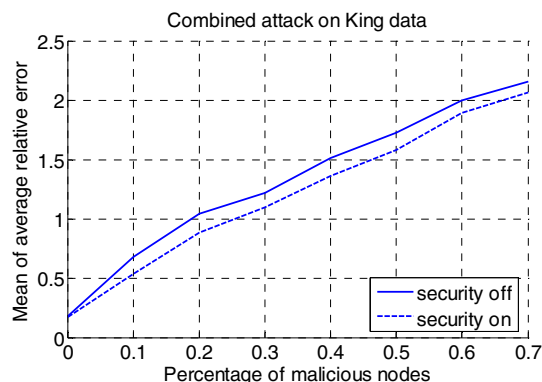


Fig.8. Mean of Average Relative Error of Combined Attack on King data

Fig. 7 and 8 show that the effect of Combined Attack is impactful. When malicious nodes are more than 20%, the mean of average relative error is higher than 1. Moreover, we find that the error grows almost linearly with the increase of malicious nodes. That is, the more malicious nodes attack PIC, the higher error is produced. Combined Attack reveals that in network when different attacks attend in PIC at the same time, the accuracy of coordinate computation in PIC would decline obviously.

4. Conclusion

In this paper, we study the effect of four typical attacks by implementing them on PIC, a representative network coordinate system. Moreover, since PIC proposes a security policy based on triangle inequality, we research into how well the security policy can protect PIC from attacks. From the results, we get two conclusions as follows.

First, from the results of four types of attack on PIC, we find that PIC is vulnerable to these attacks and each attack would lead to higher error than PIC without

malicious nodes. Moreover, the average relative error rises with the increase of malicious nodes, which would attract hackers to control as many nodes as possible to attack PIC effectively.

Secondly, by analyzing the difference between PIC with security and without security in different attacks, we found that although security policy is useful when less than 40% malicious nodes attack PIC, the security policy becomes less powerful in protecting PIC from attacks when there are more than 40% malicious nodes. Especially in Colluding Isolate Attack, when malicious nodes are more than 40%, PIC with security performs even worse than without security.

In the future, we will focus our attention on designing an effective security policy based on our study of attacks on network coordinate systems and deploying it in real network.

Acknowledgement

This work is supported by the National Basic Research Program of China (No.2007CB310806) and the National Science Foundation of China (No.60473087, No.60703052).

References

- [1] T. S. Eugene Ng and Hui Zhang. *Predicting internet network distance with coordinates-based approaches*. In Proceedings of the IEEE INFOCOM, New York, June 2002.
- [2] T. S. Eugene Ng and Hui Zhang. *A Network Positioning System for the Internet*. In Proceedings of the USENIX annual technical conference, Boston, June 2004.
- [3] M. Costa, M. Castro, A. Rowstron and P. Key. *Practical Internet coordinates for distance estimation*. In Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS), Tokyo, March 2004.
- [4] F. Dabek, R. Cox, F. Kaashoek and R. Morris. *Vivaldi: A decentralized network coordinate system*. In Proceedings of the ACM SIGCOMM, Portland, August 2004.
- [5] Y. Chen, Y.Q. Xiong, X.H Shi, B.X. Deng and X. Li. *Pharos: A Decentralized and Hierarchical Network Coordinate System for Internet Distance Prediction*. In Proceeding of the 50th Annual IEEE Global Telecommunications Conference (GLOBECOM), Washington, D.C., November 2007.
- [6] M. A. Kaafar, L. Mathy, T. Turletti and W. Dabbous. *Real attacks on virtual networks: Vivaldi out of tune*. In Proceedings of the SIGCOMM workshop on Large Scale Attack Defense (LSAD), Pisa, September 2006.
- [7] A. Rowstron and P. Druschel. *Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems*. In Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms, Heidelberg, November, 2001.
- [8] K. P. Gummadi, S. Saroiu, and S. D. Gribble. *King: Estimating Latency between Arbitrary Internet End Hosts*. In Proceedings of SIGCOMM Internet Measurement Workshop (IMW), Pittsburgh, November 2002.
- [9] PlanetLab: <http://www.planet-lab.org/>
- [10] J. A. Nelder and R. Mead. *A simplex method for function minimization*. Computer Journal, 7:308–313, 1965.
- [11] M. A. Kaafar, L. Mathy, T. Turletti and W. Dabbous. *Virtual networks under attack: Disrupting internet coordinate systems*. In Proceedings of Second CoNext Conference, Lisbon, 2006.