Motif-Based Attack Detection in Network Communication Graphs

Krzysztof Juszczyszyn, Grzegorz Kolaczek

Motif-based Attack Detection in Network Communication Graphs

Krzysztof Juszczyszyn, Grzegorz Kołaczek

Institute of Informatics, Faculty of Computer Science and Management,
Wroclaw University of Technology,
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
{Krzysztof.Juszczyszyn, Grzegorz.Kolaczek}@pwr.wroc.pl

Abstract. We propose an original approach which allows the characterization of network communication graphs with the network motifs. As an example we checked our approach by the use of network topology analysis methods applied to communication graphs. We have tested our approach on a simulated attacks inside a scale-free network and data gathered in real networks, showing that the motif distribution reflects the changes in communication pattern and may be used for the detection of ongoing attacks. We have also noticed that the communication graphs of the real networks show a distinctive motif profile.

1 Introduction

The most intensively explored approach to unknown threats detection is anomaly detection. We first give a brief list over some earlier traffic anomaly detection methods. The earliest anomaly detection-based approach, proposed by Denning, employs statistics to construct a point of reference for system behavior. The training of an anomaly detection sensor is accomplished by observing specific events in the monitoring environment such as system calls, network traffic or application usage, over a designated time period [12]. The basic problem is what method should be used to measure deviation. The example of statistical anomaly detection is e.g. Haystack [13], Intrusion Detection Expert System (IDES) [14], Next-Generation Intrusion Detection Expert System (NIDES) [15]. Machine learning techniques focus on building a system that improves its performance based on previous results. This type of anomaly detection involves learning the behavior and recognizing significant deviations from the normal. [16] Another machine learning technique that has been frequently used in the domain of machine learning is the sliding window method and Bayesian network-based methods which are frequently used in conjunction with statistical techniques [17,18]. Another approach is- principal components analysis (PCA) which aims to make more efficient anomaly detection process. Principal components analysis allows to reduce the complexity of the analysis [19]. To eliminate the manual and ad hoc elements researchers are increasingly looking at using data mining techniques for anomaly detection [19,20]. This paper proposes an original approach of applying motif analysis to the characterization of network

communication graphs. To best authors' knowledge this is the novel approach to anomaly detection in Internet traffic.

2 Network motifs as local topology patterns

2.1 Network motifs

A biased distribution of local network structures, a.k.a. network motifs is widely observed in complex biological or technology-based networks [8]. The motifs are small (usually 3 to 7 nodes in size) subgraphs that occur far more (or less) often then in the equivalent (in terms of the number of nodes and edges and node degree distribution) random networks [7]. The statistical significance of motif M is usually defined by its Z–score measure Z_M:

$$Z_M = \frac{n_M - \langle n_M^{rand} \rangle}{\sigma_M^{rand}} \tag{1}$$

where n_M is the frequency of motif M in the given network, $\langle n_M^{rand} \rangle$ and σ_M^{rand} are the mean and standard deviation of M's occurrences in the set of random networks [3]. If some motif is not found, its Z-score is not defined. Most algorithms assume full enumeration of all subgraphs. However, the algorithm presented in [4] is asymptotically independent of the network size and enables fast detection of motifs. It will be used in this paper to simplify the detection of network motifs. For biological networks, it was suggested that network motifs play key information processing roles [7,9]. Such results reveal that, in general, we may conclude about function and properties of very large networks from their basic building blocks [2,6]. SPs for small social networks (<100 nodes) were studied in [9], more detailed study is presented in [21]. A web network counting 3.5×10^5 nodes [1] was used to show the usability of motif sampling algorithm [5].

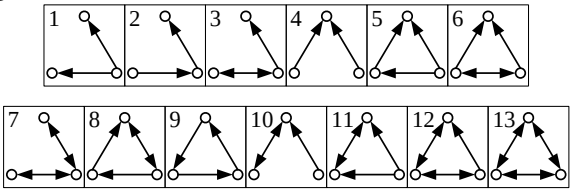

Fig. 1. All possible 3-node directed triads

There are 13 different motifs that consist of three nodes (Fig. 1). Their ID=1,2,…,13 are used in the further descriptions interchangeably with the corresponding abbreviations M1, M2,…, M13. The vector of 13 Z-scores of 3-node motifs measured for the network under consideration will be called (following [8]) a Triad Significance Profile (TSP). The TSP constitutes the individual profile of the network, showing its tendency to form triads of the given type [9]. Our experiments with motif detection were performed with FANMOD tool dedicated for motif

detection in large networks [12][13]. If not stated otherwise, our sets of reference networks always consisted of the 1000 random networks used for structure comparison.

3 Experiment 1: Seq. Scanning (SS) and Hit-list (HL) attacks

3.1 Attack patterns

Internet worms are programs that self-propagate across a network exploiting security or policy flaws in widely-used services. There are two major classes of them, scanbased worm" and email worms. Email worms propagate through emails and they do it relatively slowly; scan-based worms propagate by generating IP addresses to scan vulnerable target, in result the act much faster - Slammer worm in January 2003 infected 90% of vulnerable computers in the Internet within just 10 minutes [11].

In this paper, we concentrate on scan-based worms. Internet worms use many scanning strategies [18]. The basis of our Internet worm modelling is the classical epidemic model [12]. The experimental testbed is assumed to be a homogeneous network — any infectious host has the equal probability to infect. Once a host is infected by a disease, it is assumed to remain in this state. We simulate:

1. Sequential Scanning: This scenario lets each newly infected system choose a random addresses and, then scans sequentially from there.
2. Hit-list worm: A hit-list worm first scans and infects all vulnerable hosts on the hit-list first, then randomly scans the entire network to infect others. It means that at first, worm tries to infect the hosts which previously communicated with the infected node.

The following parameters were assumed for our experiments:

- N=1000 : The total number of host hosts in experimental network
- V=30 : The population of vulnerable hosts in experimental network, the number of vulnerable hosts is approximately 3% of all population [10].
- I=100 : Average scan rate. The average number of scans an infected host sends out per unit time (time window).

We have assumed relatively high scan rate, in order to track short time, massive attacks. The normal communication for experimental network has been modeled using Barabási scale-free network model [13] with $\gamma = 3$; 1000 nodes and 1971 edges. Communication is being observed in time windows. We have modelled the perturbations during normal network operation - 25% of links disappear and emerge between time windows. The network consisted of 1000 nodes and a subset of the initial set of 1971 edges, with average value 1210 edges. During experiments the worm related communication patterns has been added to these "normal" patterns according to the abovementioned Sequential-scanning and Hit-list scenarios.

3.2 The results for SS and HL attacks

First, we have computed the Triad Significance Profiles (see sec. 2.1) for the test network working at normal conditions. As our test network is generated a random procedure (preferential attachment), all the Z-scores should be close to 0, which is

exactly the result we have obtained: the absolute values of all of them were below 3 with the missing Z-scores for motifs M11-M13, especially M13 (if a subgraph is not found in the network, its Z-score is undefined). Typically, this concerns subgraphs which contain many edges, hardly found in networks generated by random schemes. Next, we have checked the TSP for the network under attack. We have assumed an infection of single host during T=1, so the first results of an attack are visible in T=2. The TSPs during an SS attack are shown on the Fig.2. Note the visible difference in the values of most Z-scores, especially the appearance of dominant value for M13. During a worm attack, the communication graph becomes more dense, as the worm communicates randomly with potential victims. The following characterize the network under attack:

- Growing values of Z-scores of all the motifs (each edge added to the network may constitute several new subgraphs of the size 3, so we experience growing values in the entire TSP).
- The appearance of Z-scores of the motifs M12-M13, with the value for M13 being dominant in the TSP.

The simulation of an HL attack leads to the similar results (Fig.3). The result for M13 is still significant, with additional appearance of M11 (which may be associated with probing two communicating nodes by the worm).

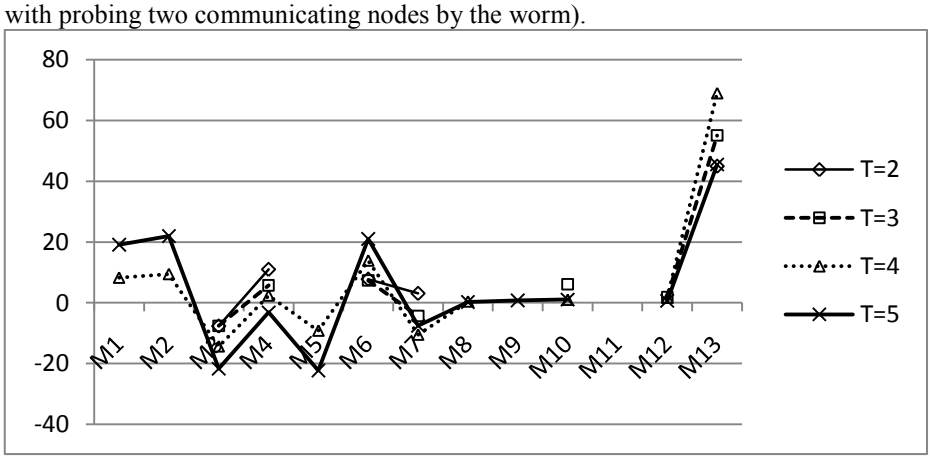

Fig. 2. TSPs for 4 consecutive phases of SS attack.

As the motifs M11-M13 show maximum link density, we may conclude that in sparse graphs an attack leads to the emergence of new links which are responsible for the high values of their Z-scores. Next we have checked the sampling strategy. For TSPs the sampling provides much faster way to obtain them. Network sampling procedure used the approach proposed in [5] and was based on checking the neighbourhood of randomly chosen network nodes. It [5] it was shown that it may be enough to check around 10% of existing triads to build a TSP with good accuracy.

The results below show the TSPs for the two already discussed attacks (we have used exactly the same data) – only 10% of existing triads were checked. The results of assessing the TSPs with network sampling are presented on the Figs 4 and 5 and should be interpreted in the same way as those from Figs 2 and 3.

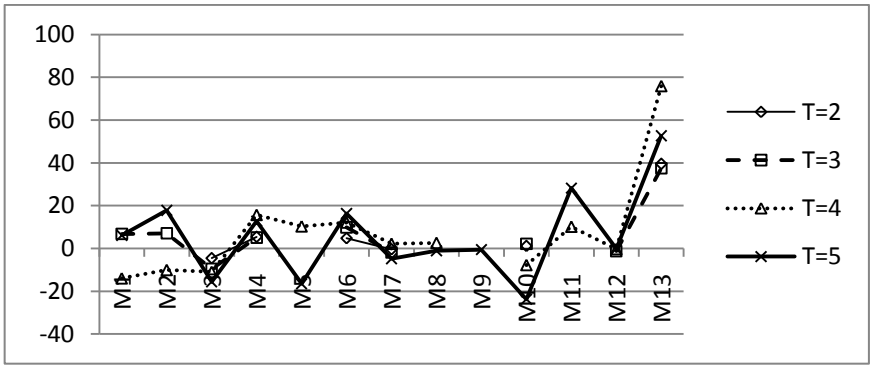

Fig. 3. TSPs for 4 consecutive phases of HL attack.

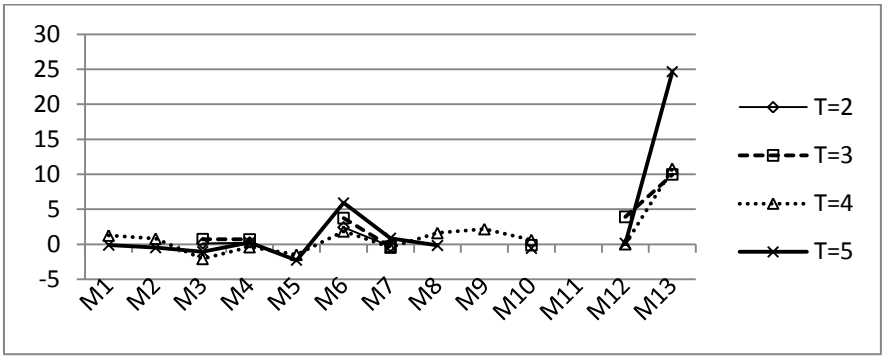

Fig. 4. TSPs for 4 phases of SS attack. – estimation with network sampling algorithm.

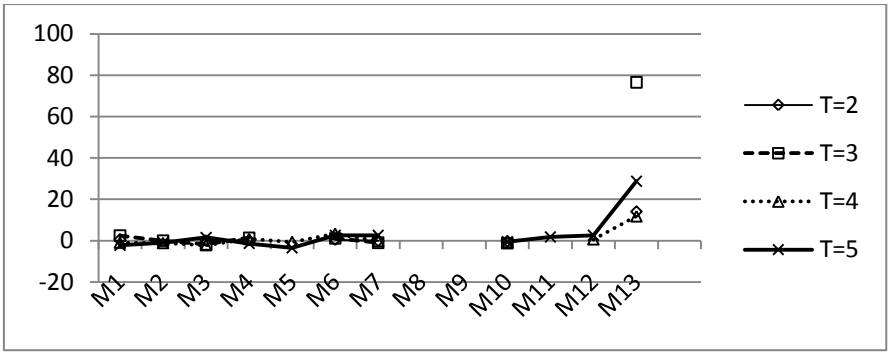

Fig. 5. TSPs for 4 phases of HL attack. – estimation with network sampling algorithm.

The consecutive phases of attacks are visible – the abnormal Z-scores of M13 and, partially, M12 show that the communication graphs are deformed. However, the measured Z-scores are lower, due to sampling accuracy. Additionally, in the case of SS attack, its profile is not detected for T=2 (Fig.4).

4 Experiment 2: network logs

4.1 Experimental data

In this part we are dealing with the communication graph built from the real network data gathered in short periods of time. This has serious consequences in the context of structural analysis of these graphs. During short intervals we observe a broadcast-type communication, which results in graphs reach in hubs and isolated nodes. Additionally, the nodes often change their roles – the network hub may become isolated node in the next time window. In result, the typical structural network measures are not effective. For this experiment, traffic logs have been taken from MAWI (Measurement and Analysis on the WIDE Internet) database samplepoint-D [22]. The data include week-long record (from 25-31.01.2009) among more than 3000 IPv6 nodes.

4.2 The TSP change during the DDoS attack

Contrary to the former experiment, the data for this one were collected during short periods of time, which resulted in extremely sparse graphs. Fig.6 presents the TSPs for the networks based on the communication data. The Z-scores are low, however, in the case of real network their values are not random-like (this happens only when the network is generated by randomly-driven algorithm) but show a distinctive pattern (always negative scores of M5 and M8, positive for M4 and M7 etc.) which feature will be later used in more detailed analysis during our research.

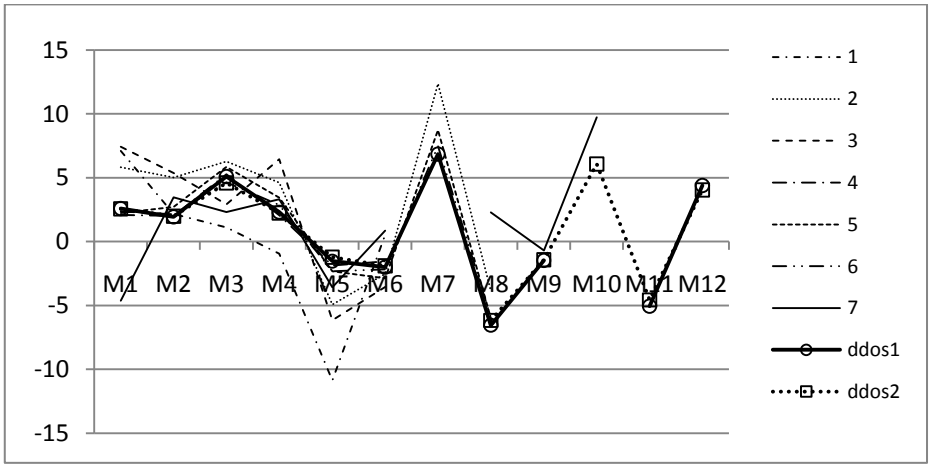

Fig. 6. The TSPs of networks during normal operation and a DDoS attack.

The attack data (series ddos1 and ddos2 on Fig.6) were analyzed for time windows of the same length as normal network data (series 1-7). Motif M13 was not detected, which can be explained by the sparsity of the network. The TSPs for attack networks are not distinguishable for M1-M10, but there is a difference for M11 and M12. The

occurrences of these motifs were not found in normal networks, they appear only in the case of the attacks. Moreover, the Z-score of M11 is negative, showing its frequency below average, the M12 Z-score is positive. Their values constitute a fingerprint of the attack, visible in the Z-score.

The last experiment was to apply network sampling to the DDoS attack data, but in this case, due to the sparsity of the network and the number of links hardly exceeding the number of nodes, the sampling procedure did not return any significant results.

5 Conclusions and future work

We have presented an original approach which allows the characterization of network communication graphs with the network motifs. We have tested our approach on a simulated attack inside a scale-free network showing that the TSPs reflect the changes in communication pattern and may be used for the detection of ongoing attacks. In the next step we have evaluated our method on the logs collected during real attacks, with additional restriction of very short time windows, for which the networks were created. The above results are preliminary and open a way for further development of our method:

- Discovering the attack type by the network TSP analysis.
- Characterizing the TSP of networks during normal operation.
- Checking the possibilities of applying sampling procedures to attack detection.
- Merge the sampling idea with a model of distributed multiagent system dedicated for attack detection.
- Checking the approach in various classes of networks (WAN, LAN, wireless…)

The above concepts will be also tested in real environments starting from the Wroclaw University of Technology network and its network security laboratory.

6 Acknowledgements

This work was supported by the Polish Ministry of Science and Higher Education, grant no. N N516 518339.

7 References

1. Barabasi A.-L. Albert R. (1999) Emergence of scaling in random networks. Science, 286, 509–512.
2. Chung-Yuan H., Chuen-Tsai S., Chia-Ying C., Ji-Lung H. (2007) Bridge and brick motifs in complex networks, Physica A, 377, 340–350.
3. Itzkovitz S., Milo R., Kashtan N., Ziv G., Alon U. (2003) Subgraphs in random networks. Physical Review E., 68, 026127.
4. Kashtan N., S. Itzkovitz S., Milo R., Alon U. (2004) Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. Bioinformatics, 20 (11), 1746–1758.
4. Mangan S. Alon U. (2003) Structure and function of the feedforward loop network motif. PNAC, USA, 100 (21), 11980–11985.

5. Mangan S., Zaslaver A. Alon U. (2003) The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks. J. Molecular Biology, 334, 197–204.
6. Milo R., Itzkovitz S., Kashtan N., Levitt R., Shen-Orr S., Ayzenshtat I., Sheffer M., Alon U. (2004) Superfamilies of evolved and designed networks. Science 303(5663), 1538–42.
7. Milo R., et. al. (2002) Network motifs: simple building blocks of complex networks. Science, 298, 824–827.
8. Shen-Orr S., Milo R., Mangan S., Alon U. (2002) Network motifs in the transciptional regualtion network of Escherichia coli. Nat. Genet., 31, 64–68.
9. Wernicke.S. (2006) Efficient detection of network motifs. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 3 (4), 347–359.
10. Wernicke S., Rasche F. (2006) FANMOD: a tool for fast network motif detection. Bioinformatics, 22 (9), 1152–1153
11. S. E. Smaha, Haystack: An intrusion detection system, IEEE Fourth Aerospace Computer Security Applications Conference, Orlando, FL, 1988, pp.37–44.
12. D. Anderson, et. al., Next generation intrusion detection expert system (NIDES), SRI International, USA, TR SRI-CSL-95-0, 1994.
13. C. Kruegel, D. Mutz, W. Robertson, F. Valeur, Bayesian event classification for intrusion detection, 19th CSA Conference, Las Vegas, NV, 2003.
14. W.W. Cohen, Fast effective rule induction, in: Proceedings of the 12th International Conference on Machine Learning, USA, 1995, pp. 115–123.
15. C. Warrender, S. Forrest, B. Pearlmutter, Detecting intrusions using system calls: alternative data models, IEEE Symposium on Security and Privacy, Oakland, CA, USA, 1999, pp. 133–145.
16. A. Valdes, K. Skinner, Adaptive model-based monitoring for cyber attack detection, Recent Advances in Intrusion Detection, Toulouse, 2000, pp. 80–92.
17. M. L. Shyu, S.C. Chen, K. Sarinnapakorn, L. Chang, A novel anomaly detection scheme based on principal component classifier, IEEE Foundations and New Directions of Data Mining Workshop, Melbourne, FL, USA, 2003, pp. 172–179.
18. W. Lee, S.J. Stolfo, K.W. Mok, A data mining framework for building intrusion detection models, IEEE Symposium on Security and Privacy, Oakland, CA, 1999, pp. 120–132.
19. S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, ACM SIGMOD, Dallas, USA, 2000, pp. 427–438.
20. M. Breunig, H.-P. Kriegel, R.T. Ng, J. Sander, LOF: identifying density-based local outliers, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, Dallas, TX, 2000, pp. 93–104.
21. K.Juszczyszyn, K.Musial, P.Kazienko, B.Gabrys: Temporal Changes in Local Topology of an Email-Based Social Network. Computing and Informatics 28(6): 763-779 (2009).
22. MAWIlab http://mawi.wide.ad.jp/mawi/