Knowledge discovery from network logs*

Tuomo Sipola

Abstract Modern communications networks are complex systems, which facilitates malicious behavior. Dynamic web services are vulnerable to unknown intrusions, but traditional cyber security measures are based on fingerprinting. Anomaly detection differs from fingerprinting in that it finds events that differ from the baseline traffic. The anomaly detection methodology can be modelled with the knowledge discovery process. Knowledge discovery is a high-level term for the whole process of deriving actionable knowledge from databases. This article presents the theory behind this approach, and showcases research that has produced network log analysis tools and methods.

1 Network anomaly detection

A modern communications network is a collection of interconnected computers. The number of these computers and the detailed communications paths are usually unknown in the global view, they are just anonymous parts of a huge machinery. It is very challenging to know who are connected to the network and what kind of clients and servers there are at a certain time point. Therefore, many hiding places exist for malicious clients. There is a growing need to find these attackers and prevent their actions by cyber security methods.

Dynamic web services are vulnerable to unknown intrusions that grant access to systems that are not meant for the public audience. The trust between the network

Tuomo Sipola
c/o Tapani Ristaniemi
Department of Mathematical Information Technology, University of Jyväskylä
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland
e-mail: tuomo.sipola@iki.fi

* This article is partly based on the author's dissertation [19]. Author's current affiliation is with CAP Data Technologies.

participants is broken as legitimate features can be used for such unwanted access [14]. However, server logs contain interesting information about network traffic. Finding attacks from these logs naturally improves security of the services. So called intrusion detection systems analyze the logs in order to find malicious traffic and the attackers [15].

1.1 Fingerprinting

Traditional cyber security measures are based on fingerprinting, or pre-defined rules. Incoming traffic is compared to a database of known attacks, and then filtered according to the rules. All firewalls, IP blacklists and traditional intrusion detection systems are based on this approach. It is an effective way of controlling the network but with today's dynamic web services and protocols it is not possible to control everything with static rules. Moreover, the existing channels can be used for attacks, and blocking them would inhibit the normal use of the network.

1.2 Anomaly detection

Anomalies, or outliers, are data samples that differ vastly from the baseline traffic. Anomaly detection techniques are used in many application areas, one of which is security. There are many anomaly detection methods, each having their own advantages [2]. Anomalies in networks are in a sense very common occasion. There is always happening something new and the whole network as a system is evolving as time advances. However, not all new events are the kind of anomalies that system administrators are interested in. It is a natural state that the network changes.

Therefore, there is a need to build a profile, or baseline, of clean network functioning. Anomaly detection differs from fingerprinting in that it finds events that differ from the baseline traffic even if such deviations are not previously known. Fingerprinting on the other hand reveals and prevents only known threats.

2 Network environment

Figure 1 shows the different components of a model network. The Internet connects all the computers to each other. Normal users make queries to the servers and get responses with content. Attackers, on the other hand, try to access the servers, collect information or disrupt operations. Some servers are not protected at all while others rely on traditional protection, such as firewalls. The anomaly detection system is deployed to the protected server in addition to traditional measures.

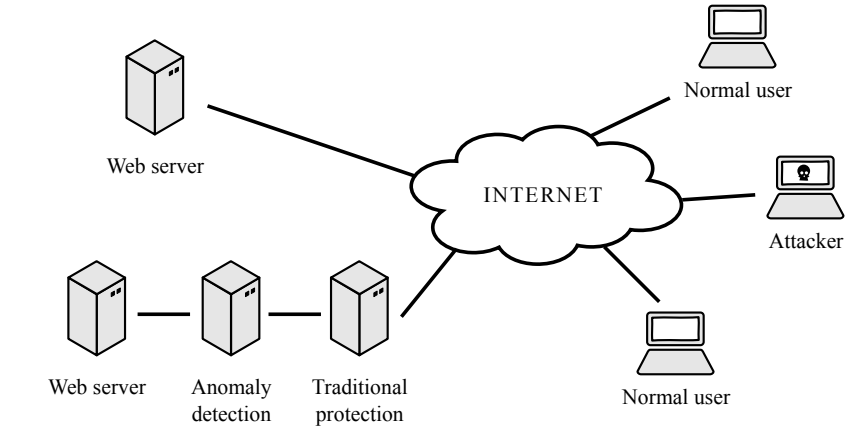

Fig. 1 Schematic network environment.

3 Knowledge discovery process

The anomaly detection methodology can be modelled with the knowledge discovery process. Knowledge discovery is a high-level term for the whole process of deriving actionable knowledge from databases. Presenting data mining as a part of the knowledge discovery process places the technical challenges in the broader scope. The knowledge discovery process from databases (KDD) suggests the steps that are needed to extract business knowledge from available data [9, 8, 10, 1].

Figure 2 shows the schematic workflow during the knowledge discovery process. The steps are described in more detail below.

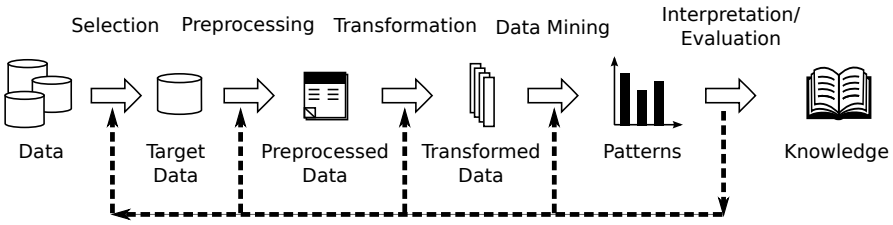

Fig. 2 Steps of the knowledge discovery process according to Fayyad et al. [9].

3.1 Databases

In the context of cyber security, the relevant databases include server and user logs, captured packet data and ultimately any collected information about the functioning

of the system. The technical implementation of the data store is a challenge since the volumes of such data masses are huge. However, if all the data is not needed for later reference, storing is not a big problem.

3.2 Selection

In the data selection step the most relevant data sources are selected. It is usually the case that too many data sources and data features are available. Those sources that might contain important information should be selected, which requires domain knowledge. The selection process heavily depends on the knowledge discovery task. In cyber security tasks this would mean that the log files and the time frames should be selected. Some other sources might also be needed to give background information.

3.3 Preprocessing

Once the target data is defined, it needs to be preprocessed. Besides preprocessing, data cleaning is also a relevant. Converting and collecting the data to correct formats takes effort. Noise removal is a typical preprocessing step. Incomplete data entries and known large changes need to be accounted for. Combining and cleaning various databases is a rather mechanical process but sometimes poses problematic situations. The amount of work needed at this stage is usually underestimated in practical work.

3.4 Transformation

Preprocessed data is transformed to a more suitable form for clustering and classification. This involves feature extraction and selection, dimensionality reduction and other transformations. The selected features depend on the data mining case, as does the number of needed final features.

3.5 Data mining

The data mining step itself tries to extract patterns from the transformed data. Summarization, classification, regression and clustering are some common tasks at this stage. The goal of the knowledge discovery process is matched with the relevant data mining methods. Often, this includes exploratory analysis of the data, which

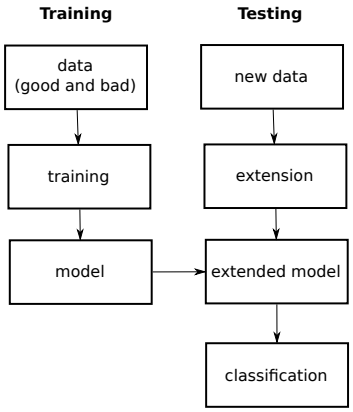

Fig. 3 Machine learning: training creates a model, while testing classifies new data using the model.

helps in deciding the most suitable models and parameters for the methods. The end products of this stage are rules, decision trees, regressions and clustering of the data.

3.6 Interpretation and evaluation

Finally, the business value of the results should be understood. In this step, interpreting the patterns creates the actual final result of the knowledge discovery process. Evaluation of the obtained results is also important because some assumptions might have been wrong, the data might have been insufficient or there might have been some problem during the previous steps. Not all results have business significance even if they present some new information that has scientific or statistic novelty.

This research focuses on the transformation and data mining steps of the knowledge discovery process. The other steps are intimately connected to the utilized data, but the data mining methods are usually separate modules that can be described as independent systems. This is not to say, however, that in all cases the selection of data mining method is independent of the data and knowledge discovery problem. Correct tools should be used with differing datasets.

The actual data mining step usually uses a machine learning method. A supervised machine learning system is first trained using known data labeling (hence supervision), and then the performance of the system is tested. The testing results reveal the quality of the learned model, provided that the testing material adequately reflects the data in a real situation. This idea of training and testing is illustrated in Figure 3.

4 Some proposed approaches

This section introduces some approaches to knowledge discovery from network logs. These research articles provide one point of view to the problem, but they are not the only ones.

Diffusion map -based approaches have been used for SQL injection detection and network traffic classification [5, 7, 6]. These methodologies extract the relevant features from the data and create a low-dimensional presentation of the structure of the data, which can be clustered and classified to identify the anomalous traffic.

The goal of [20] is to find security attacks from network data. The proposed anomaly detection scheme includes n-gram feature extraction [4], dimensionality reduction and spectral clustering style linear clustering [17, 13]. It could be used for query log analysis in real situations. In practice the boundary between normal and anomalous might not be as clear as in this example. However, the relative strangeness of the sample could indicate how severe an alert is. The data in question is rather sparse and the discriminating features are quite evident from the feature matrix. This is the merit of the n-gram feature extraction which creates a feature space that separates the normal behavior in a good manner. The features describe the data clearly, and they are easy to process afterwards. The presented anomaly detection method performs well on real data. As an unsupervised algorithm this approach is well suited to finding previously unknown intrusions. This method could be applied to offline clustering as well as extended to a real-time intrusion detection system.

These results are elaborated in [21]. The dimensionality reduction framework adapts to the log data. It assumes that only few variables are needed to express the interesting information, and finds a coordinate system that describes the global structure of the data. These coordinates could be used for further analysis of characteristics of anomalous activities. The practical results show that abnormal behavior can be found from HTTP logs. The main benefits of this framework include:

- The amount of log lines that needs to be inspected is reduced. This is useful for system administrators trying to identify intrusions. The number of interesting log lines is low compared to the total number of lines in the log file.
- The unsupervised nature and adaptiveness of the framework. The proposed methods adapt to the structure of the data without training or previous knowledge. This makes it suitable for exploration and analysis of data without prior examples or attack signatures. This means that the framework may also detect zero-day attacks.
- It works on the application layer in the network. The attacks themselves must in some way target the actual applications running on the computer. These logs might be more available than pure low-level network packet data.
- Visualization of text log data. It is much easier to analyze the structure of traffic using visualizations than it is to read raw textual logs.

The feature extraction from the web log is currently done with n-grams. However, this is only one method for it and other text-focused features might better describe

the dataset. Furthermore, the dimensionality reduction scheme could be developed to adapt to this kind of data more efficiently, and the quality of the reduction could also be evaluated. Finally, automated root cause detection would make the system more usable in practice.

In [11] a framework for preprocessing, clustering and visualizing web server log data is presented and used for anomaly detection, visualization and explorative data analysis. The results indicate that there are traffic structures that can be visualized from HTTP query information. Traffic clustering can give new information about the users. They could be categorized with more accuracy, and individual advertising or content could be offered. Using data mining methods, underlying structure and anomalies are found from HTTP logs and these results can be visualized and analyzed to find patterns and anomalies.

Article [12] deals with extracting rules from the clustering results provided by a diffusion map training framework. Modern data mining technology in network security context does not always create understandable results for the end users. Therefore, this so-called black box system is not a desirable end goal. Simple conjunctive rules [3, 16] are easier to understand, and rule extraction from the complex data mining techniques might facilitate user acceptance. The main benefit of this framework is that the final output is a set of rules. No black box implementation is needed as the end result is a simple and easy to understand rule matching system. The training data may contain intrusions and anomalies, provided that the clustering step can differentiate them. In addition, rule matching is a fast operation compared to more complex algorithms. The proposed framework is useful in situations where high-dimensional datasets need to be used as a basis for anomaly detection and quick classification. Such datasets are common nowadays in research environments as well as in industry, because collecting data is widespread. Our example case has been network security, which bears real benefits to anyone using modern communication networks. The provided tools are useful for network administrators who are trying to understand anomalous behavior in their networks.

In [22] another approach is taken to create a more online system. The training phase is computationally expensive in machine learning algorithms. Evolving datasets require updating the training. The proposed method updates the training profile using the recursive power iterations algorithm [18] and a sliding window algorithm for online processing. The algorithms assume that the data is modeled by a kernel method that includes spectral decomposition. A web server request log where an actual intrusion attack is known to happen is used to illustrate the online processing. Continuous update of the kernel prevents the problem of multiple costly trainings.

5 Conclusion

Knowledge discovery from network logs is a step forwards when trying to prevent attackers from using previously unknown attack types. This approach complements

the traditional technologies that use fingerprints to detect the attackers. Anomaly detection finds the unusual traffic from networks and alerts the operators or uses automated countermeasures that prevent the attack from happening. This protection enhances the security of the web and creates a more efficient communications network for all participants.

References

1. Brachman, R.J., Anand, T.: Advances in knowledge discovery and data mining. chap. The process of knowledge discovery in databases, pp. 37–57. American Association for Artificial Intelligence (1996). URL http://dl.acm.org/citation.cfm?id=257938.257944
2. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys (CSUR) **41**(3), 15 (2009)
3. Craven, M., Shavlik, J.W.: Using sampling and queries to extract rules from trained neural networks. In: ICML, pp. 37–45. Citeseer (1994)
4. Damashek, M., et al.: Gauging similarity with n-grams: Language-independent categorization of text. Science **267**(5199), 843–848 (1995)
5. David, G.: Anomaly Detection and Classification via Diffusion Processes in Hyper-Networks. Ph.D. thesis, Tel-Aviv University (2009)
6. David, G., Averbuch, A.: Hierarchical data organization, clustering and denoising via localized diffusion folders. Applied and Computational Harmonic Analysis (2011)
7. David, G., Averbuch, A., Coifman, R.: Hierarchical clustering via localized diffusion folders. In: Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) Fall Symposium Series (2010)
8. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery in databases. AI Magazine **17**(3), 37–54 (1996)
9. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: The KDD process for extracting useful knowledge from volumes of data. Communications of the ACM **39**, 27–34 (1996)
10. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., et al.: Knowledge discovery and data mining: Towards a unifying framework. In: KDD, vol. 96, pp. 82–88 (1996)
11. Juvonen, A., Sipola, T.: Adaptive framework for network traffic classification using dimensionality reduction and clustering. In: IV International Congress on Ultra Modern Telecommunications and Control Systems 2012 (ICUMT 2012), pp. 274–279. St. Petersburg, Russia (2012)
12. Juvonen, A., Sipola, T.: Combining conjunctive rule extraction with diffusion maps for network intrusion detection. In: The Eighteenth IEEE Symposium on Computers and Communications (ISCC 2013), pp. 411–416. Split, Croatia (2013)
13. Meila, M., Shi, J.: A random walks view of spectral segmentation. In: 8th International Workshop on Artificial Intelligence and Statistics (AISTATS) (2001)
14. Mukkamala, S., Sung, A.H.: A comparative study of techniques for intrusion detection. In: Tools with Artificial Intelligence, 2003. Proceedings. 15th IEEE International Conference on, pp. 570–577. IEEE (2003)
15. di Pietro, R., Mancini, L.V.: Intrusion detection systems. Springer (2008)
16. Ryman-Tubb, N.F., d'Avila Garcez, A.: Soar sparse oracle-based adaptive rule extraction: Knowledge extraction from large-scale datasets to detect credit card fraud. In: Neural Networks (IJCNN), The 2010 International Joint Conference on, pp. 1–9. IEEE (2010)
17. Shi, J., Malik, J.: Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on **22**(8), 888–905 (2000)
18. Shmueli, Y., Wolf, G., Averbuch, A.: Updating kernel methods in spectral decomposition by affinity perturbations. Linear Algebra and its Applications **437**(6), 1356–1365 (2012)

19. Sipola, T.: Knowledge discovery using diffusion maps. Ph.D. thesis, University of Jyväskylä (2013)
20. Sipola, T., Juvonen, A., Lehtonen, J.: Anomaly detection from network logs using diffusion maps. In: L. Iliadis, C. Jayne (eds.) Engineering Applications of Neural Networks, *IFIP Advances in Information and Communication Technology*, vol. 363, pp. 172–181. Springer Boston (2011)
21. Sipola, T., Juvonen, A., Lehtonen, J.: Dimensionality reduction framework for detecting anomalies from network logs. Engineering Intelligent Systems **20**(1–2), 87–97 (2012)
22. Yaniv Shmueli Tuomo Sipola, G.S., Averbuch, A.: Using affinity perturbations to detect web traffic anomalies. In: Proceedings of the 10th International Conference on Sampling Theory and Applications (SampTA 2013), pp. 444–447. EURASIP, Bremen, Germany (2013)