Semantic Anomaly Detection in Computer Networks

Henry Clausen, November 23, 2018

Intrusion detection and semantic models

Cyber Security relies on a range of defensive techniques, including sophisticated intrusion detection systems and firewalls that try to detect and prevent attacks against software subsystems. Malicious software still remains the biggest threat to computer users, and its detection is of utmost importance.

Program analysis methods is used widely to automatically test software for particular characteristics and behaviour, and thus identify malicious instances. Close attention has to be paid at the type of features and models that quantify the behaviour of software, as a lack of general program representation leads to low robustness against new or polymorphic malware, and consequently a poor classification performance. Chen et al. (2016) [3, 2] demonstrated convincingly that semantic features are suitable to reflect the nature of software, benign or malware, in an accurate manner.

In modern computer networks, it is often impossible use software analysis on a large-scale basis to prevent network intrusions through malicious software. Here, network intrusion detection systems play a vital role in protecting computer networks from malicious access. Current detection methods are predominantly based on the analysis of previously identified attack signatures, which provides great reliability and low false alert rates. However, these methods are dependent on an updated attack signature database and provide no protection against previously unseen attacks.

Another approach that has recently gained traction in commercial deployment is based on detecting malware and other undesired activity as anomalous behaviour when compared to benign computer activity. For that, models that quantify the behaviour of normal network are trained on attack-free traffic data. Observed behaviour that significantly deviates from the trained model is then denoted as anomalous, and the intrusion detection system is taking further steps to investigate a possible intrusion.

Generally, network traffic is usually collected either on a atomic level as raw network packets, or in a more structured way as network connection summaries, also called network flows. Both formats consist of symbolic sequence of events with different types of attributes. These can be of numeric, such as the number of bytes transferred, or categorical nature, such as the specific connection port, or entirely different such as the contacted IP address or the checksum of a package. Such formats are not appropriate as direct input for a generative traffic model. Instead, features have to be mined from the raw traffic in order to provide a suitable data format. These features have represent the traffic properties that should be quantified by the designed traffic model. The design of an appropriate model and the choice of corresponding features for capturing the different aspects of network traffic behaviour is not trivial, and there exists a substantial body of research concerned with this topic. I discuss this issue more thoroughly in the literature review attached below.

After reviewing literature, I found that there exists little work that attempts to model network traffic on a semantic level. This is both true on a packet level inside a connection, as well as on a network flow level. Furthermore, I identified the lack of meaningful features

provided by widespread network flow collection tools as a severe problem that fails to be addressed by the majority of flow-based approaches. These two topics are related and corresponding research can be combined into one PhD-project aimed at providing network modelling tools based on semantic traffic features.

1 Semantic modelling of connection communication and feature generation

Network communication between two hosts usually takes place in the form of TCP-connections. Such a connection is established by one host (client) sending a request packet to the other (server), and upon acceptance of the request, the two hosts communicate by exchanging network packets containing data or relevant instructions. The exchange of packets often follows distinct communication protocols, which are defined by the corresponding software application. Although fact is largely overlooked in intrusion detection, the field of traffic classification uses the existence of distinct communication patterns to identify the type of traffic a connection belongs to, even when the content of the transmitted packets is encrypted. These semantic patterns manifest themselves in different forms, some of which are listed below:

1. The first few packets in a connection usually transmit the necessary user authentication and other important information about the data to be transmitted. Therefore, they follow a distinct order of client and server replying each others using packets with distinct properties such as size, flags, or interarrival time. Similarly, the end of a connection is often equally shaped by specific packet order [1].

2. Data sent from client to server or vice-versa is normally broken into multiple chunks, each of which needs to be acknowledged by the receiver. The size and direction of these chunks as well as the distance between chunk transfers can be characteristic of the connection type [4].

3. Idle times in a connection are often used by applications to optimise traffic flow or await further information. Their length and frequency is therefore governed by the specific implementation and can therefore be used as information criteria [5].

4. Interarrival times of packets are often related to the type of content in the packet and take different values over a connection [5].

Traffic classification methods often use statistical or machine-learning models to learn the distinct semantic patterns of different types traffic from collected training data in an automated. In a similar fashion, we want to introduce an intrusion detection model that can learn the different patterns present in the traffic of a network from training data. If a connection is observed that deviates strongly from the extracted connection patterns, the system should identify it as an anomaly and flag it correspondingly. Additionally, it would be desired to generate features for each connection that summarise its semantic nature. As mentioned earlier, connection summaries currently severly lack meaningful features, so the generation of semantic summary features can potentially improve flow-based models greatly.

Chen et al. [3, 2] in their approach used *timed behaviour automata* to extract sequences of reoccurring events in Android call graphs. An automata encode patterns of short-term

interactions of a system and is a representation of symbolic sequences, often corresponding to state transitions, which it encodes in a state transition function. Pellegrino et al. [6] recently demonstrated that a probabilistic automata model can capture behaviour sequences in network traffic, however with a different objective. Timed behaviour automata are therefore a promising candidate to model the initial communication patterns between clients and the servers. In order to extract the long-term interaction patterns in connections, the use of *recurrent neural networks* with a long-short-term-memory cell appear to be suitable.

2 Temporal modelling of network flows

Despite network traffic being a stream of events, most anomaly-based intrusion detection approaches neglect any temporal features. Nevertheless, malicious behaviour most often is composed of a series of related computer and network events and have a distinct temporal and semantic profile [8]. Since a machines network traffic is the collective stream of multiple processes accessing the internet, individual traffic sequences are mixed with others, which makes the modelling of temporal dependencies a non-trivial task. Any suitable model must be able to incorporate noise and variation in the traffic patterns or temporal correlation it captures.

Initial work on creating a temporal model of short network flow sequences has been conducted by David Aspinall et al. [7]. The authors test three different approaches, namely a recurrent neural network, behaviour automata mining, and a simple markov chain, to predict the properties of successive flow events in small time intervals. The work demonstrated that it is possible to create a predictive model of the existing features in the noisy enviromnent of network traffic, and that such a model can be used to identify anomalous behaviour.

As I described in the attached literature review, current network flow generators provide only little informative features about the summarised connection, none of which contain any semantic information. Therefore, even the best designed predictive model of network flows might not be able to represent a machine's traffic sufficiently enough to identify stealthy attacks. As I described above, semantic models of network connections can be used to generate abstract semantic features which should represent its purpose and behaviour much better. My work could extend the work of Aspinall et al. in a straight forward manner by extending the tested methods to incorporate such additional semantic features. This will enable the identification of more distinct flow sequences and represent them in a more detailled manner, and thus combine these two modelling approaches at different levels into one PhD-project.

References

[1] Laurent Bernaille, Renata Teixeira, Ismael Akodkenou, Augustin Soule, and Kave Salamatian. Traffic classification on the fly. *ACM SIGCOMM Computer Communication Review*, 36(2):23–26, 2006.

[2] Wei Chen, David Aspinall, Andrew D Gordon, Charles Sutton, and Igor Muttik. More semantics more robust: Improving android malware classifiers. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, pages 147–158. ACM, 2016.

[3] Wei Chen, David Aspinall, Andrew D Gordon, Charles Sutton, and Igor Muttik. On robust malware classifiers by verifying unwanted behaviours. In *International Conference on Integrated Formal Methods*, pages 326–341. Springer, 2016.

[4] Felix Hernandez-Campos, Andrew B Nobel, F Donelson Smith, and Kevin Jeffay. Understanding patterns of tcp connection usage with statistical clustering. In *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2005. 13th IEEE International Symposium on*, pages 35–44. IEEE, 2005.

[5] Anthony McGregor, Mark Hall, Perry Lorier, and James Brunskill. Flow clustering using machine learning techniques. In *International Workshop on Passive and Active Network Measurement*, pages 205–214. Springer, 2004.

[6] Gaetano Pellegrino, Qin Lin, Christian Hammerschmidt, and Sicco Verwer. Learning behavioral fingerprints from netflows using timed automata. In *Integrated Network and Service Management (IM), 2017 IFIP/IEEE Symposium on*, pages 308–316. IEEE, 2017.

[7] Marc Sabate, Gudmund Grov, Wei Chen, and David Aspinall. On robust anomaly detection with behavioural models. 2018. Under review.

[8] Nong Ye et al. A markov chain model of temporal behavior for anomaly detection. In *Proceedings of the 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, volume 166, page 169. West Point, NY, 2000.