Self-Learning Network Intrusion Detection

Selbstlernende Angriffserkennung im Netz

Konrad Rieck, Technische Universität Berlin, laureate of the CAST/GI Dissertation Award IT-Security 2010

Summary Services in the Internet are confronted with a growing amount and diversity of network attacks. Regular instruments of computer security increasingly fail to fend off this threat, as they rely on the manual generation of detection patterns and lack protection from unknown threats. In this article, we present a framework for self-learning intrusion detection, which allows for automatically identifying unknown attacks in the application layer of network communication. The framework links concepts from computer security and machine learning for deriving geometric models of normal network data and identifying attacks as deviations thereof. Empirically, this ability can be demonstrated on real network traffic, where a prototype of the framework identifies 80–97% of unknown attacks with less than 0.002% false positives and throughput rates between 26–60 Mbit/s.

►►► **Zusammenfassung** Dienste im Internet sind einer wachsenden Anzahl und Diversität von Angriffen ausgesetzt. Herkömmliche Instrumente der IT-Sicherheit sind ungeeignet, dieser Bedrohung langfristig entgegen zu wirken, da sie auf der manuellen Erstellung von Erkennungsmustern beruhen und keinen Schutz vor neuen und unbekannten Angriffen bieten. In diesem Artikel wird ein Rahmenwerk zur selbstlernenden Angriffserkennung vorgestellt, das es ermöglicht, unbekannte Angriffe gegen Netzwerkdienste automatisch zu erkennen. Das Rahmenwerk verbindet Konzepte der IT-Sicherheit und des maschinellen Lernens, um Nutzdaten der Dienste geometrisch zu analysieren und Angriffe als Ausreißer zu erkennen. Diese Fähigkeit kann empirisch bestätigt werden, wobei ein Prototyp eine Erkennung von 80–97% der unbekannten Angriffe mit weniger als 0,002% falschen Alarmen und einem Durchsatz von 26–60 Mbit/s erzielt.

Keywords K.6.5 [Computing Milieux: Management of Computing and Information Systems: Security and Protection]; network security, intrusion detection, machine learning ►►► **Schlagwörter** Netzwerksicherheit, Angriffserkennung, maschinelles Lernen

1 Introduction

Over the last years the Internet evolved to a universal communication platform that provides a wealth of services to its users, including electronic commerce, social networks and broadband communication. With this rapid growth, however, crime has found its way to the Internet. Services in the Internet are at steady risk of being compromised and misused for illegal purposes, such as theft of user data, propagation of malicious software or distribution of spam messages. This threat is driven by an underground economy that systematically employs and advances network attacks for compromising network services [3; 15].

Unfortunately, regular instruments of computer security increasingly fail to protect from the threat of network attacks. The majority of protection measures pursues the concept of misuse detection, where attacks are identified using known patterns of misuse. While effective

Dr. Rieck earned his doctorate at the School of Electrical Engineering and Computer Sciences of the Technische Universität Berlin. His dissertation is entitled *Machine Learning for Application Intrusion Detection*. It has been awarded with the Prize 2010 for Best Dissertation in IT-Security by the Competence Center for Applied Security Technology (CAST) e. V. and the Gesellschaft für Informatik e. V. The examiners were Prof. Dr. Klaus-Robert Müller, TU Berlin, Prof. Dr. John McHugh, Dalhousie University, Canada, and Dr. Pavel Laskov, University Tübingen.

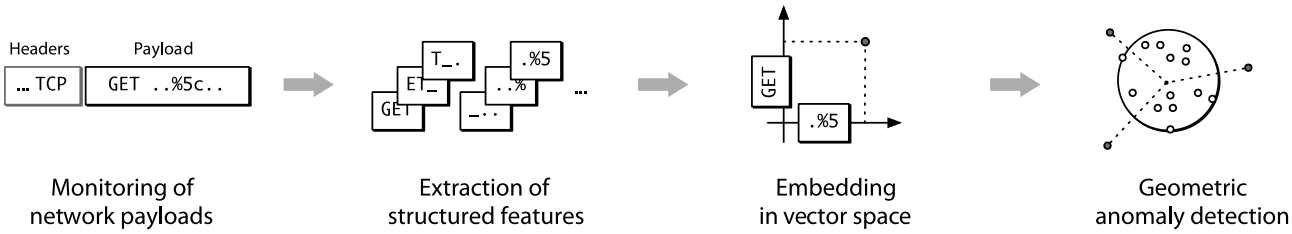


Figure 1 Schematic depiction of self-learning intrusion detection.

against known threats, misuse detection inherently lags behind attack development and fails to protect from novel threats. Crucial time elapses from discovery of a new attack to deployment of a detection pattern, as the attack needs to be manually inspected and an appropriate pattern crafted. Consequently, services in the Internet regularly fall victim to novel attacks and there is an urgent demand for alternative techniques capable of identifying unknown threats.

In this article, we address this problem and present a framework for self-learning intrusion detection, which allows to automatically identify unknown attacks in the application layer of network traffic. The framework builds on linking concepts from computer security and machine learning. To this end, the incoming network payloads of a service are embedded in a vector space, such that their characteristics are expressed geometrically and accessible to means of machine learning. This geometric representation enables learning models of normal communication and detecting unknown attacks as deviation thereof, independently of manually crafted detection patterns. A schematic depiction of this process is shown in Fig. 1, where a network payload is exemplarily embedded and geometrically analysed.

The ability to detect unknown network attacks can be empirically demonstrated on real network traffic and attacks. In these experiments, a prototype of the framework called SANDY significantly outperforms the popular intrusion detection system SNORT and state-of-the-art anomaly detection methods by identifying 80–97% of unknown attacks with less than 0.002% false positives. During operation, SANDY attains a throughput of 26–60 Mbit/s, rendering it readily applicable for protection of small and medium network services. Moreover, it possesses the ability to visualize detected anomalies for further forensic analysis. These results demonstrate that self-learning intrusion detection provides a valuable instrument for protecting network services – despite recurring preconceptions in the security community [4; 14].

The framework for self-learning intrusion detection is systematically developed in the dissertation [9]. We herein provide a brief introduction to its main contributions: the generic embedding of network payloads (Sect. 2) and the geometric detection of attacks (Sect. 3). Additionally, we present some of the results obtained with real network traffic and attacks (Sect. 4).

2 From Payloads to Vector Spaces

The syntax and semantics of communication with network services are defined by standard protocols, such as HTTP, DNS or FTP. Although these protocols precisely specify the form of transferred content, this representation of data is not directly suitable for analysis using machine learning, as learning methods usually operate on numerical vectors. Hence, a key to self-learning detection of attacks is the embedding of network payloads in a vector space.

For this embedding, we consider a network payload as a string of bytes sent to a service, which depending on the granularity of analysis may correspond to the content of a network packet, request or connection. This string can be characterized using different classes of features, which range from simple numerical measures to sequential and syntactical constructs, such as sets of strings and parse trees. If we associate the features of one such class with the dimensions of a vector space, we obtain a generic way for associating payloads with vectors, where individual dimensions reflect the occurrences of features in the payloads.

To understand how this embedding works, let us consider the feature class of n-grams as an example. An n-gram is essentially a string of length n and the network payloads are mapped to a vector space, whose dimensions are associated with all such strings. As shown in Fig. 1 for $n = 3$, this embedding is carried out by first extracting all n-grams from a payload and then generating a vector, such that all dimensions associated with the extracted n-grams are set to 1 and all other dimensions to 0. In this way, the content and structure of the payload are reflected geometrically and thereby they are accessible to means of machine learning.

This embedding, however, imposes a dilemma: On the one hand, the accurate detection of attacks requires an expressive vector space with as many features as possible, while on the other hand efficiently operating with millions of dimensions easily turns intractable. Fortunately, for several feature classes the embedding is sparse, that is, the vast majority of dimensions is zero. For example, the vector space induced by n-grams comprises 256^n dimensions, yet a network payload of m bytes contains at most m unique n-grams. This sparsity can be exploited to derive linear-time algorithms for extraction and comparison of vectors [10]. Instead of operating with full vectors,

only the non-zero dimensions are considered during analysis of network payloads.

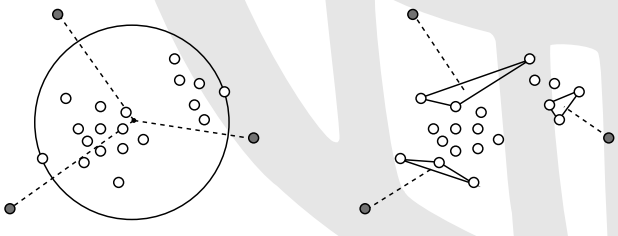
3 Geometric Anomaly Detection

The embedding of network payloads induces a geometry in the vector space. Payloads that share several features lie close to each other and form dense clouds in the vector space, whereas payloads with few shared features are scattered in different regions. These geometric relations can be exploited to derive statistical models of normal communication, which enable identifying all attacks that deviate in their features from normality. Thus, as the second key contribution, the dissertation [9] introduces concepts for geometric anomaly detection and learning models of network payloads.

Network attacks often significantly deviate from normal communication. For example, many buffer overflow attacks contain uniform byte patterns, which infrequently occur in legitimate payloads. Such deviation can be identified by a *global model* of normality, where the model captures features shared by the majority of payloads. An intuitive geometric shape reflecting this concept is a hypersphere (a sphere in a high-dimensional vector space). Normality is modeled by placing a hypersphere around the embedded payloads and the deviation from this model is determined by the distance from the hypersphere [13]. Figure 2a shows a global model enclosing a set of points.

However, normality can not always be cast into a global model. For example, if a web server provides multiple virtual hosts, the network payloads partially share features and are scattered across different regions in the vector space. This problem can be addressed by a *local model* of normality. Geometrically, a local perspective can be derived from the concept of k-nearest neighbors [2]. Normality is modeled by the local neighborhood of an embedded payload and the deviation from this model is determined by the distance to the k-nearest neighbors. Figure 2b illustrates the local detection of anomalies for a set of points.

In contrast to previous work, both models for anomaly detection are solely defined in terms of geometry and can be applied for arbitrary network services and embeddings. As a result, realizations of this detection technique have been successfully applied for various network services, for example using the protocols HTTP, FTP, SMTP, SMB and SIP. Another essential advantage of geometric models is their robustness to noisy training data. In particular, the hypersphere model can be extended with a "soft margin", which allows to compensate unknown attacks in the learning data.



(a) Global model (b) Local model

Figure 2 Geometric anomaly detection.

4 Experiments and Applications

The proposed framework provides a link between computer security and machine learning by modeling intrusion detection as a geometric problem. In practice, however, it is not an elegant design but the sheer performance of a security tool that matters. Consequently, a prototype of the framework called SANDY has been developed and evaluated for detection performance, robustness and network throughput. SANDY employs the feature class of n-grams and a global model of normality, as this combination best balances detection and run-time performance [9].

For the evaluation, 10 days of consecutive network traffic have been acquired for the application-layer protocols HTTP and FTP. The HTTP traces have been recorded at the web server of Fraunhofer Institute FIRST, while the FTP traffic has been obtained from the public FTP server of Lawrence Berkeley National Laboratory [8]. Additionally, a total of 151 different attacks against HTTP and FTP services have been recorded and intermixed with the normal traffic. These attacks have been generated using common hacking tool, such as METASPLOIT [7], and for the sake of security, have been targeted against virtual copies of the original services.

To assess the detection of unknown attacks, the network traffic is randomly split into a *known* and *unknown* partition during multiple experiments. Models for anomaly detection are trained on the payloads of the known partition only, whereas detection results are only reported for the unknown partition. In this setting, SANDY significantly outperforms related methods and identifies 80–97% of the attacks in the unknown set with less than 0.002% false positives. Neither the popular intrusion detection system SNORT nor state-of-the-art methods for payload-based anomaly detection [5; 6; 16] attain a similar accurate detection of unknown attacks. Moreover, SANDY attains a throughput rate between 26–60 Mbit/s, although common speed-up techniques such as multi-processing and hardware accelerations have not been integrated yet.

As an example of this evaluation, Fig. 3 shows the detection performance of SANDY and SNORT as a ROC curve, where the false-positive rate is given on the x-axis and the detection rate on the y-axis for varying sensitivity of the methods. SNORT attains a flat curve which saturates at 80% detection, despite a database of recent detection patterns. In contrast, SANDY enables an almost perfect identification of attacks with a false-positive rate

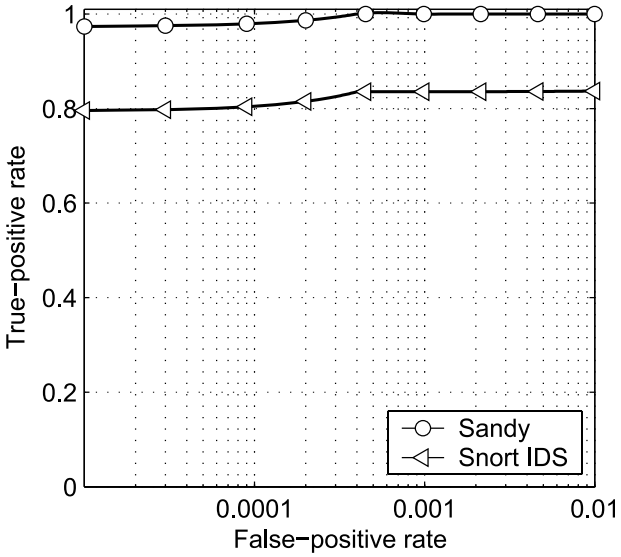


Figure 3 Detection performance (HTTP).

of 0.002%, although all attacks have been *unknown* during the learning phase.

A comparison with methods for payload-based anomaly detection [5; 6; 16] is presented in Fig. 4. The detection performance is shown as $AUC_{0.01}$ (the area under the ROC curve from Fig. 3) for increasing fractions of unknown attacks in the training data. On clean data, several anomaly detection methods perform similarly to SANDY. However, if only 0.5% of the training data contains unknown attacks (one out of 200 normal requests) the performance drastically drops for all methods except SANDY. This compensation of unknown attacks can be credited to the "soft margin" employed in the prototype, which compensates attacks when learning the hypersphere of normality. Although the sanitization of training data may lessen the impact of unknown attacks [1], these results demonstrate that robustness is a critical prerequisite for learning-based intrusion detection.

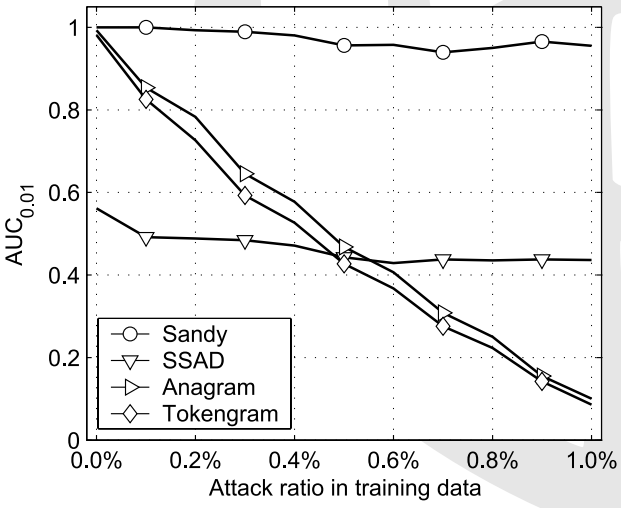


Figure 4 Robustness on contaminated data (HTTP).

5 Conclusions

In this article, we have studied a framework for self-learning intrusion detection that enables *effective* and *efficient* identification of unknown attacks. The framework rests on two key concepts: the embedding of network payloads in a vector space which provides a universal link to machine learning and the geometric anomaly detection that enables identifying attacks as deviations from normal communication independent of manual crafted detection patterns. While self-learning intrusion detection does not generally eliminate the threat of network attacks, it considerably raises the bar for adversaries to get their attacks through network defenses. In combination with existing security measures, such as firewalls and intrusion prevention systems, it safeguards today's networks against future threats.

Moreover, the concept of geometrically analysing data also proves beneficial in other areas of computer security. For example, techniques presented in this article have also been successfully applied for analysing the behavior of malicious software [12] and detecting attacks against web browsers [11]. With the increasing automatisation of attacks and malicious software, these learning-based approaches provide a promising ground for effective and automatic defenses.

References

[1] G. Cretu, A. Stavrou, M. Locasto, S. Stolfo, and A. Keromytis. Casting out demons: Sanitizing training data for anomaly sensors. In: *Proc. of IEEE Symp. on Security and Privacy*, 2008.

[2] R. Duda, P. E. Hart, and D. G. Stork. *Pattern classification*. John Wiley & Sons, second edition, 2001.

[3] J. Franklin, V. Paxson, A. Perrig, and S. Savage. An Inquiry Into the Nature and Causes of the Wealth of Internet Miscreants. In: *Proc. of Conf. on Computer and Communications Security (CCS)*, pp. 375–388, 2007.

[4] C. Gates and C. Taylor. Challenging the anomaly detection paradigm: A provocative discussion. In: *Proc. of New Security Paradigms Workshop (NSPW)*, pp. 21–29, 2006.

[5] K. L. Ingham and H. Inoue. Comparing anomaly detection techniques for HTTP. In: *Recent Adances in Intrusion Detection (RAID)*, pp. 42–62, 2007.

[6] C. Kruegel, T. Toth, and E. Kirda. Service specific anomaly detection for network intrusion detection. In: *Proc. of ACM Symp. on Applied Computing (SAC)*, pp. 201–208, 2002.

[7] K. Maynor, K. Mookhey, J. F. R. Cervini, and K. Beaver. *Metasploit Toolkit*. Syngress, 2007.

[8] V. Paxson and R. Pang. A high-level programming environment for packet trace anonymization and transformation. In: *Proc. of Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pp. 339–351, 2003.

[9] K. Rieck. *Machine Learning for Application-Layer Intrusion Detection*. Doctoral thesis, Berlin Institute of Technology (TU Berlin), 2009.

[10] K. Rieck and P. Laskov. Linear-time computation of similarity measures for sequential data. In: *Journal of Machine Learning Research*, vol. 9, pp. 23–48, June 2008.

[11] K. Rieck, T. Krueger, and A. Dewald. Cujo: Efficient detection and prevention of drive-by-download attacks. In: *Proc. of 26th Annual Computer Security Applications Conf. (ACSAC)*, pp. 31–39, 2010.
[12] K. Rieck, P. Trinius, C. Willems, and T. Holz. Automatic analysis of malware behavior using machine learning. In: *Journal of Computer Security*, vol. 19(3), 2011.
[13] J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge University Press, 2004.
[14] R. Sommer and V. Paxson. Outside the closed world: On using machine learning for network intrusion detection. In: *Proc. of IEEE Symp. on Security and Privacy*, pp. 305–316, 2010.
[15] Symantec. Symantec Global Internet Security Threat Report: Trends for 2009. Vol. XIV, Symantec, Inc., 2010.
[16] K. Wang, J. Parekh, and S. Stolfo. Anagram: A content anomaly detector resistant to mimicry attack. In: *Recent Adances in Intrusion Detection (RAID)*, pp. 226–248, 2006.

Received: April 4, 2011

Dr. Konrad Rieck, Laureate of the CAST/GI Dissertation Award IT-Security 2010, is a postdoctoral researcher at Technische Universität Berlin. He graduated from Freie Universität Berlin and received a Doctorate in Computer Science from Technische Universität Berlin. He is soon to take up a position as Assistant Professor for Computer Security at the University of Göttingen.

Address: Technische Universität Berlin, Franklinstrasse 28/29, 10587 Berlin, Germany, Tel.: +49-30-314-78630, Fax: +49-30-314-78622,
e-mail: konrad.rieck@tu-berlin.de