Poster: Combining Data Loss Prevention and Detection

Elisa Costante
SecurityMatters
Email: elisa.costante@secmatters.com

Davide Fauri, Sandro Etalle, Jerry den Hartog, Nicola Zannone
Eindhoven University of Technology
Email: {d.fauri, s.etalle, j.d.hartog, n.zannone}@tue.nl

Abstract—**In this paper, we propose a hybrid Data Loss Protection framework that combines signature-based and anomaly-based solutions, enabling *both* detection and prevention of unauthorized disclosures of data. The framework uses an anomaly-based engine that automatically learns a model of normal user behavior, allowing it to flag when insiders carry out anomalous transactions. Typically, anomaly-based solutions stop at this stage. Our framework goes further in that it exploits an operator's feedback on alerts to automatically build and update signatures of attacks which are used to timely block undesired transactions before they can cause any damage.**

I. Introduction

Data loss, i.e. the unauthorized disclosure of sensitive information from a corporate network or a database, is an increasing threat. A recent study [1] shows that over 502 million records, including credit card numbers, access credentials and other personal information, were leaked in the first half of 2014. Organizations can lose their competitive advantage if confidential information is stolen. Moreover, data breaches can affect customers' perception towards a company's image by decreasing its reputation, especially if sensitive personal information is leaked. Unsurprisingly, data leakages are typically propagated by Insider Threats [2].

To minimize the risk of data breaches, organizations often employ Data Loss Protection (DLP) solutions that monitor the access and exchange of confidential data to identify unauthorized disclosure or improper usage [3]. To distinguish allowed from malicious transactions, DLP systems maintain a model of either allowed (whitelisting) or malicious (blacklisting) behaviour. This model can either be specified based on an expert's knowledge or learned from past transactions.

A blacklist with signatures describing well-known attacks hardly produces any false positive, allowing it to be used for *prevention* by blocking attacks before they are executed. However, such an approach cannot detect unknown attacks. In particular, it is often easy for insiders to avoid blacklisting-based detection. The insider has (privileged) access to systems, and can usually carry out actions that qualify as data leakage without breaking the system's rules and/or using leakage paths that are specific to the target system, and which cannot be considered in a general-purpose signature.

Anomaly-based solutions, which learn a model of normal behavior and flag any deviation from the model as a suspicious activity, can find unknown attacks but may have a high false positive rate. As such, anomaly-based systems are typically used only for *detection*; they raise an alert upon detecting a suspicious activity but do not block the activity. Alerts typically have to be manually analyzed to determine whether they are false positive or they correspond to an actual attack. This, however, has high operational costs and a lengthy response time to security incidents.

MacDonald [4] states that the main problem is that enterprises have long prioritized threat prevention over detection and response. Since it is impossible to have a signature available before an attack, it is necessary to have the ability to define new signatures as soon as new attacks are identified.

We address the problem of identifying and reacting to insider threats by monitoring user activities and detecting anomalous behavior. Moreover, once the security operator flags an anomaly as suspicious, we create on-the fly rules that are able to block any further transactions that match the suspicious pattern.

To block new attacks, we design and integrate a prevention system with a white-box anomaly detection system in the style of [5]. The key characteristic of a white-box approach is that it provides an operator with the root causes of alerts. This allows the operator to interpret an alert and determine whether the alert is a false positive (i.e., a legitimate transaction marked as a suspicious activity) or a true positive (i.e., an actual attack). This feedback is used to improve the model for detection (false positive) or for prevention (true positive). In the latter case, the root causes of alerts are used to create and maintain blocking (or warning) rules that are used to prevent (or signal) the execution of the flagged activities in the future.

II. Framework

Existing DLP solutions are not fully able to cope with the problem of data breaches. To overcome their limitations, we propose a hybrid framework that combines signature-based and anomaly-based approaches. In particular, it provides capabilities to immediately respond to an alert by automatically creating rules that are used to block similar queries. An overview of the framework is presented in Fig. 1. It consists of five main phases: *(i)* learning; *(ii)* prevention, *(iii)* detection, *(iv)* alert analysis and *(v)* rule management.

During the learning phase, transactions are analyzed by a learning engine to create profiles of normal behavior. Here, we consider profiles created using the white-box anomaly-based solution presented in [5]. This solution specifies profiles of normal behavior in terms of feature histograms: specifically,

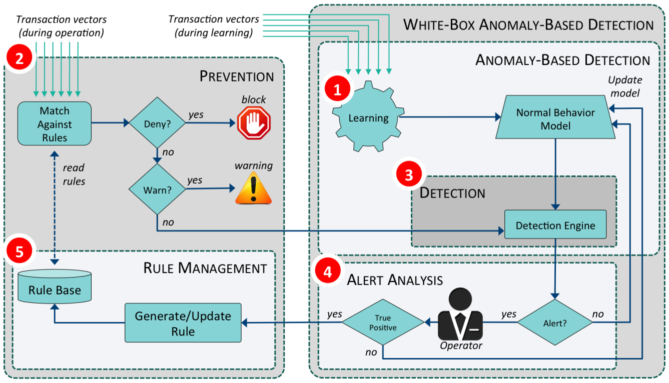

Figure 1: Framework for Data Loss Prevention and Detection

for every feature, a histogram is learned from a given set of transactions by analyzing the frequency of feature values. However, the framework is general enough to be extended to any detection tool able to generate white-box alerts indicating the root causes of anomalous transactions.

Every new transaction is analyzed by the prevention module. This module matches the transaction against a *rule base*, which comprises deny (or blocking) rules and warning rules. If the rule base contains a rule matching the transaction, the transaction is blocked or a warning is raised according to the type of rule that is fired; otherwise, the transaction is passed through to the detection engine. The detection engine aims to detect unknown attacks. In particular, this engine verifies whether the transaction matches the previously learned profiles of normal behavior. In case of a match, the transaction is used to update the current profiles; otherwise, an alert is raised.

Alerts are analyzed by an operator, who leveraging his domain knowledge can flag them as true/false positives. If an alert is marked as a false positive (i.e., it corresponds to a legitimate activity), it is used to update the model of normal behavior. Otherwise, if the alert corresponds to a malicious activity, the operator can decide to *enforce* it, namely to automatically create some rules (devised from the alert) to be added to the rule base. In particular, the operator can decide to enforce the alert by creating a blocking rule or a warning rule. This way, when a similar transaction arrives, it can be blocked or signaled before its execution without further intervention from the operator, hence providing prevention capabilities.

III. Validation

To validate our approach, we performed a number of experiments. It is trivial to prove the correctness of our approach in blocking all similar repeated attacks, i.e. those with the same root causes, as soon as the first true positive is identified. Thus, in the experiments we focused on evaluating how *effective* our approach is, in blocking new 'unwanted' behavior and thus reducing the effort for the operator. In the experiments, we henceforth assumed that every alert raised by the detection module is a true positive: upon receiving an alert, the preventive module creates a blocking rule for every root cause, and updates the rule base accordingly.

To assess the impact of our solution we measured the number of alerts raised over time, with and without the prevention module in place. This gives a measure of the transactions that were blocked, thus providing an indication of the reduced effort for the operator and, indirectly, of how much data loss was prevented. We did not analyze the effectiveness of the detection engine in terms of detection rate and false positive rate as this has already been evaluated in a previous work (see [5]): we note that a reduction in the absolute amount of alerts implies a proportional reduction in the absolute amount of false positives.

For the experiments we used both synthetic and real-life logs. In particular the real-life log was taken from an (Oracle) operational database of a large IT company. The dataset was created by enabling the DBMS auditing facility, and it contains a total of 12,040,910 transactions from about 100 users. We used 3,612 and 361,227 transactions (approximately 0.03% and 3% of the whole dataset resp.) to learn the normal behavior, while we used 70% of the dataset for validation. We observed a proportional reduction of the number of alerts of approximately 18% when the small dataset was used for the training of the detection engine, and 15% when the large dataset was used.

Finally, although our implementation is not optimized with respect to performance, the time to match a transaction against a rule base containing over a thousand rules is around 0.4 ms, which makes our approach suitable for real time applications.

IV. Conclusion

In this paper, we overcome the limitations of previous DLP techniques by combining a white-box anomaly-based detection technique able to raise alerts for any previously unseen transaction, with a prevention technique that blocks transactions through a rule that is automatically created when an operator flags an anomaly as an attack. Experiments show that our approach achieves promising results, reducing the response time to alerts, the amount of data leaked, and operational costs for handling suspicious activities in that alerts for similar activities do not have to be reexamined. Finally, the use of a different feature set to characterize transactions allows the easy adaption of our solution to different domains, like web applications, firewalls or network-based intrusion detection systems.

Acknowledgments: This work has been partially supported by the European Commission through project FP7-SEC-607093-PREEMPTIVE, by the Dutch program COMMIT through the THeCS project and by NWO through the IDEA ICS project.

References

[1] Open Security Foundation, "Data breach trends during the first half of 2014," Report, 2014.
[2] C. L. Huth, D. W. Chadwick, W. R. Claycomb, and I. You, "Guest editorial: A brief overview of data leakage and insider threats," *Information Systems Frontiers*, vol. 15, no. 1, p. 1, 2013.
[3] T. Wuchner and A. Pretschner, "Data loss prevention based on data-driven usage control," in *Proc. of ISSRE*. IEEE, 2012, pp. 151–160.
[4] N. MacDonald, "Architecting a new approach for continous advanced threat protection," in *Gartner Security & Risk Manag. Summit*, 2014.
[5] E. Costante, J. den Hartog, M. Petkovic, S. Etalle, and M. Pechenizkiy, "Hunting the unknown - white-box database leakage detection," in *Data and Applications Security and Privacy XXVIII*, ser. LNCS 8566. Springer, 2014, pp. 243–259.