SKEPTIC

Reinforcing Application Security through User Behavioural Analysis

Olivier THONNARD
Amadeus IT Group, Sophia Antipolis, France
olivier.thonnard@amadeus.com

Zayani DABBABI
Amadeus IT Group, Sophia Antipolis, France
zayani.dabbabi@amadeus.com

Miruna MIRONESCU
Amadeus IT Group, Nice, France
miruna-mihaela.mironescu@amadeus.com

Damien FONTANES
Thales Group, Sophia Antipolis, France
damien.fontanes@mythalesgroup.io

Abstract — Rules and signature-based security concepts have shown their limitations in detecting and preventing sophisticated fraud in online systems due to ever-changing attacker's behaviors. To address this problem, we present a *User Behavior Analytics* (UBA) approach that reinforces application security by profiling users' activities and evaluating incoming user sessions against previously learnt models of "normal" (i.e. usual) behavior. Our SKEPTIC approach is based on an unsupervised learning approach that models the expected users' behaviors based on past activities. After the initial learning phase, the system uses a novel scoring approach to evaluate new incoming sessions and identify "anomalies" highlighting significant deviations from the previously learnt baseline models. These statistical anomalies can be identified on various types of user session features (e.g., infrequent sequences of actions, unusual connection times or abnormal connection origins) and are combined using a multi-criteria aggregation method, which continually adjusts a global risk score for each evaluated session. The risk score can be fed to an intervention module that can be configured to trigger different responses based on the severity level (e.g. trigger identity proofing or locking the user account). We have evaluated our approach on two different datasets: a synthetic dataset simulating a typical scenario of spear-phishing attack in which the attacker is impersonating a legitimate user to gain access to an enterprise network, and a dataset of application logs from a real-world ecommerce application used in the global travel industry. In both cases, our results show that our SKEPTIC approach can successfully detect suspicious events and can prevent common fraud scenarios that involve user impersonation, account take-over or credential compromise.

Keywords — User Behavior Analytics, fraud prevention, machine learning, cybersecurity, anomaly detection, artificial intelligence, credential compromise

1. Introduction

Online fraud has been growing both in volume and complexity in the recent years [1], [2]. Fraudsters have managed to continually adapt to fraud prevention solutions relying on basic rules or predefined signatures. Rule-based detection methods have thus shown their limitations and these cannot deal any more with ever-changing attacker behaviors. Fraudsters now rely more and more on sophisticated attack scenarios that involve user impersonation and account take-over, which enables them to disguise as a legitimate user and mimic his activities. This kind of advanced fraud scenario has been facilitated by the growth of *automated attacks*, such as credential stuffing or credential compromise performed at large scale [2] [3] [4].

Online fraud prevention systems must thus evolve in order to deal with this dynamic threat landscape. We have to leverage novel machine learning techniques that can help in corroborating a user's identity when connecting to an online application as well as validate his/her session activity *beyond* the login phase. This can be achieved by leveraging so-called *User & Entity Behavior Analytics* (UBA or UEBA), which is an analytics led threat detection technology that brings user profiling and anomaly detection together [5] [6]. The idea is to continuously learn users' behaviors based on their past activities and create baseline models of "expected" (e.g. usual) behaviors, which can then be used to evaluate new incoming sessions by comparing them against previously learnt models.

In this paper we present a novel UBA approach called SKEPTIC that reinforces application security by profiling users' activities thanks to an unsupervised learning method and by scoring new users' sessions according to baseline user behaviour models or UBM's (Fig. 1). The outcome of our SKEPTIC UBA system is a global *risk score* calculated for each user session, which combines a number of anomalies identified on the different user session features extracted from the application logs (e.g., infrequent sequences of actions, unusual connection times or connection origins, different characteristics of the client machine). The risk score of a session accumulates evidences of suspicious user activity and effectively reflects the abnormal degree of a user session. This can be used as input to an intervention module in order to trigger different responses at the application according to the severity level and predefined business rules.

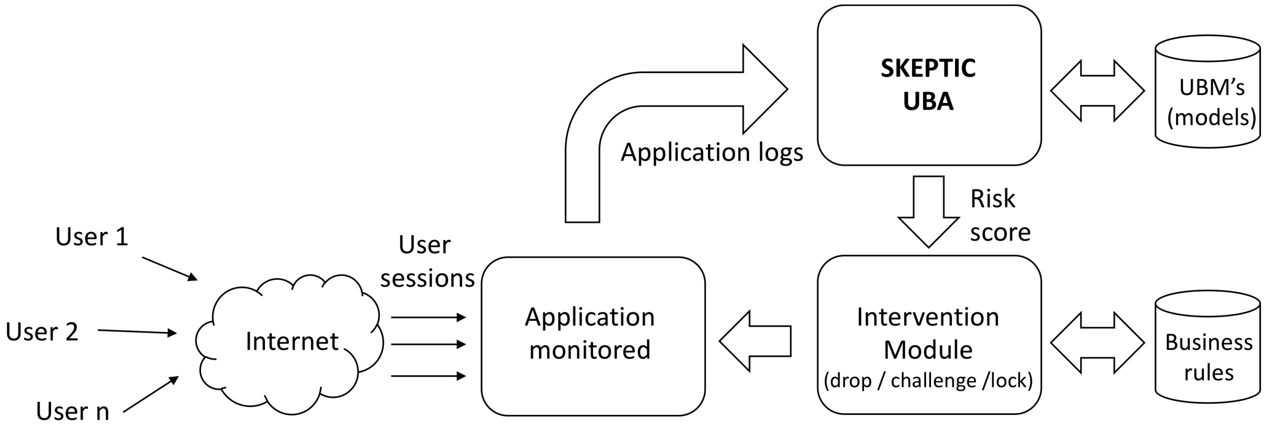

Fig. 1. Overview of SKEPTIC UBA flow and how it integrates with the application to be monitored.

In order to assess the efficacy of our approach, we performed an experimental validation on two different datasets:

- a synthetic dataset simulating a typical scenario of spear-phishing attack in which the attacker is impersonating a legitimate user to gain access to an enterprise network (this dataset was provided by a large technology provider from the Aviation industry)
- a real dataset of application logs obtained from an ecommerce application used by thousands of travel agents in the global travel industry

Our experimental results are very encouraging as they demonstrate that our SKEPTIC approach can successfully detect suspicious events and can be used to prevent online fraud scenarios that involve user impersonation, account take-over or credential compromise.

While security vendors started to include UBA components in certain commercial SIEM solutions, we believe that our SKEPTIC framework presents several advantages compared to commercial *off-the-shelf* products: (i) our approach is *generic* and *extensible*, and can be easily tuned to any application that provides online services to end-users (or entities); (ii) the approach is *non-intrusive* for the monitored application, as it only requires application logs to be fed to the system (in batch or streaming fashion); (iii) different statistical models can be integrated to fit the requirements of the monitored application in terms of attributes or session features (e.g. numerical, categorical or textual features, list or sequences of user actions, time series, geographical data, IP addresses, etc); (iv) the global *risk score* provided to analysts can be explained and easily understood by breaking it down by individual anomaly scores, which in turn highlights the reasons why a session was tagged as suspicious ("white-box" approach).

The rest of the paper is structured as follows: Section 2 presents a brief state-of-the-art and some related work; Section 3 describes the details architecture of our SKEPTIC UBA framework; in

Section 4 we present the results of our validation performed on two different datasets; Section 5 concludes the paper and give some directions for future work.

2. Related work

Anomaly detection is an open research area that considers the problem of finding *outliers* in data, i.e., observations or data points that do not conform to expected behavior [7]. Many techniques in the literature have been used to detect anomalies for different use cases and applications. Some techniques use a scoring system to rank anomalies based on the degree to which the evaluated instance deviates from the expectation. Other techniques use a labelling system to denote whether the instance is normal or anomalous. The latter is often dynamic in nature (e.g., new types of anomalies can be identified) making it difficult to associate the training data to a particular label. Many anomaly detection techniques have been proposed in the literature, some are designed and applied on certain application domains, while others are more generic. Some researchers performed surveys [8] [9] [10] on the existing anomaly detection techniques in order to provide a comprehensive overview and a taxonomy of the techniques used to solve the anomaly detection problem.

In this section, we only present some brief overview of research directions that tackle the problem of anomaly detection by leveraging supervised and unsupervised methods.

Based on the information used and the techniques employed we propose a classification of anomaly detection techniques into six major groups: (i) Statistical Methods, (ii) Knowledge Based Methods, (iii) Distance based Methods, (iv) Model based approaches and (v) Graph based Methods and (vi) Ensemble Techniques. However, anomaly detection algorithms are quite diverse in nature and thus one technique may fit into more than one category.

i. *Statistical anomaly detection techniques* work by fitting a statistical model to the data at hand and then applying a statistical test on the unseen data instance to check whether it belongs to the model. Statistical anomaly detection techniques assume that normal data instances occur in the high probability zone of a statistical model and anomalies occur in the low probability spectrum.

ii. *Knowledge Based Methods* search for instances of known attacks, by attempting to match with pre-determined attack representations.

iii. *Distance Based Methods* detect outliers by calculating different distances between points. More explicitly, they compute the full dimensional distances of points from one another using various features enhanced by the densities of local neighborhoods.

iv. *Model Based Anomaly* detection techniques are based on building learning data models using artificial learning techniques [9].

v. *Graph-Based Methods* is based on the long-range correlations properties of the graphs. Graphs or other structured data, such as the sequential data, are used by dedicated algorithms in the process of machine learning. This has the role to identify the anomalies in the graphs [11].

vi. *Ensemble Techniques Methods* have the advantage of combining multiple anomaly detection algorithms in order to boost their joint anomaly detection performance [10] [12].

As it will become clear in the following section, our SKEPTIC framework is based on a *hybrid* approach that combines several techniques from categories described here above (i.e., Ensemble, statistical and knowledge-based).

3. SKEPTIC Framework Architecture

3.1 Overview

The enhanced application monitoring extension we propose can be seen as a generic anomaly and misuse detection framework that uses different detection techniques. As introduced previously, our primary goal is to design and develop a generic, non-intrusive solution that can prevent online fraud scenarios that involve user impersonation, account take-over or credential compromise. Unlike most of the network anomaly detection approaches proposed in the literature, our approach uses *application logs* as input instead of network packets. The framework is designed to be tuned both manually and programmatically to adapt to the applications to be monitored. By adapting the models and detection algorithms used for each application to monitor, we believe the framework will perform better (in terms of accuracy and false positives) since more context and functional knowledge can be harnessed when focusing on a specific application.

The framework is based on a hybrid *ensemble* approach that combines the advantages of statistical methods with knowledge- and model-based techniques for outlier detection. The motivations behind this decision are:

- anomaly detection models are usually constructed using a subjective and heuristic process based on functional expert knowledge
- assumptions and hypotheses made in the modelling phase can be imperfect
- models may work better on some parts of the data than other

The *ensemble* analysis approach is used in order to reduce the dependence of the models on the specific data set or data locality, and increase the robustness and performance of anomaly detection process. The underlying idea is simple: combining the results from different statistical models will create a more robust model (as demonstrated by majority voting approaches such as AdaBoost [12]).

As depicted in Fig. 2, the SKEPTIC framework consists of two main components or "layers": (i) a Behavioral Engine ("modeling layer"), which is in charge of learning user behaviors and creating appropriate models (i.e. user profiles) representing normal expected activities of the users; (ii) a Multi Criteria Scoring Engine ("detection layer") that leverages the previously learnt user behavior models to evaluate new incoming session and assign anomaly scores (normalized in [0, 1]) representing the (un)usual character of certain data points extracted from the evaluated session features. In the rest of this section, we will describe the characteristics and the underlying techniques of these two components.

3.2 Behavioral Engine

The Behavioral modelling layer (*upper layer* in Figure 1) is designed to let the system continuously learn – in an unsupervised manner – users' behaviors by applying statistical analysis on historical data. Users' sessions are processed by extracting specific features from all requests made to the application once a user has logged in and has initiated a new session. Statistical learning algorithms are then used to build User Behavior Models (UBM) that will be leveraged afterwards by the detection (scoring) layer to evaluate new incoming users' actions and detect anomaly or significant deviations.

An important aspect is to adopt a *user-centric* modelling, as different (types of) users may have very different behaviors, and hence may not be subject to the same types of anomalies. The learning process is auto-adaptive: UBM's are updated dynamically based on new incoming data, since users' behaviors may change over time. It supports different types of attributes used as input to the learning modules (categorical, numerical, binary, or textual). Examples of such attributes include sequences of actions, session length, connection times, client information (IP address, geolocalization, browser fingerprints) or request/transaction rate. The system is extensible, in the sense that new features can easily be integrated later on, should we discover that these features can help to refine UBM's and may be relevant in some cases for detecting anomalies or fraudulent behaviors.

Example models that are already used in the Behavioral Engine are: probability distributions, Mean Standard Deviation, Gaussian Mixture Models, Regression Models, Markov Chain Mixture Models, and Bag of Words Model. The choice of one or another model depends on the monitored application and its features. While default models can be automatically attributed based on basic data type identification, these are best configured with the support of a functional expert who will ensure that the most appropriate models are selected for the application.

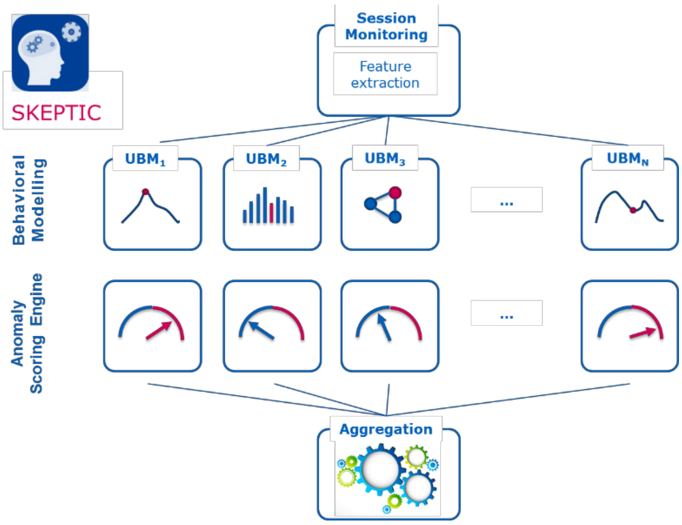

Fig. 2. SKEPTIC UBA Framework – Detailed Architecture. The upper layer ("Behavioral modelling") creates various User Behavior Models (UBM_X) which are leveraged in the lower layer ("Scoring Engine") to find anomalies in user sessions (normalized in [0,1]). All anomaly scores are then combined in the Aggregation layer.

3.3 Multi Criteria Scoring Engine

The anomaly scoring engine (SCE) – depicted in Figure 1 as the *lower layer* – leverages the behavioral models (UBM's) previously created for every feature that was extracted from the application session. The SCE associates a specific anomaly scoring function to every model. As for statistical models, rare or infrequent values will result in higher anomaly scores, whereas frequently observed values will be mapped to very low anomaly scores. For more complex features like sequences of actions, more sophisticated techniques must be used to accurately model the observed data patterns, such as Markov chains or Gaussian mixtures.

The scoring engine will evaluate new users' sessions against the previously learnt models. A global anomaly score is calculated by aggregating all individual anomaly scores using multi-criteria decision analysis (MCDA) techniques. The use of MCDA ensures that the SCE can deal with any combination of significant anomalies – even unforeseen ones, and thus allows to overcome the limitations of traditional rule-based systems which are based on static, predefined rulesets. Indeed, the SCE can be configured so as to implement complex decision schemes, whereby a set of fuzzy conditions need to be satisfied in order for the global score to lie in a certain range of values. For example, a domain expert may configure the system to accept "*at least k*", "*some*", "*many*" or "*most of*" anomaly scores having a significant value (e.g. >0.5) for the final score to be significant (i.e. between 0.5 and 1). Moreover, individual anomalies can be weighted differently to assign more or less relative importance to specific session features. In practice, this multi criteria scoring can be realised by using special aggregation functions such as Ordered Weighted Average (OWA) and Weighted OWA [13], [14].

The output of this aggregation is a global *risk score* (again in [0, 1]) which accumulates evidences of suspicious user activity within a session and can be used as input to an intervention module in order to trigger different responses at the application according to the severity of the score. Typical responses that can be triggered are: user identity proofing (e.g. via sms/email), suspending the transaction, challenging the user (e.g. secret question), or temporarily locking the user account.

4. Validation and Testing

The first dataset on which we evaluated our SKEPTIC approach was synthetic and provided by an industrial partner from the aviation industry. It is composed of two-month network-based data reflecting a typical scenario of spear-phishing attack targeting an enterprise network. An external attacker impersonates some trusted employee (e.g. an HR representative) within the organization and sends an email containing a malicious attachment to a targeted employee who unwittingly opens the weaponized document and infects his laptop with a malware. The compromised laptop of the employee is then used to scan the internal network and find vulnerable systems (i.e. lateral movement). In the last step of the attack, one of the vulnerable systems is in turn compromised and the hacker can finally connect remotely to the internal network and exfiltrate sensitive data. The targeted incongruities to be identified by SKEPTIC are then as follow:

- **Incongruity 1** : Reception of the weaponized document by an internal employee
- **Incongruity 2** : Equipment infection by the malware of the corrupted employee's laptop
- **Incongruity 3** : Use of the compromised equipment to scan and damage internal systems

While SKEPTIC was originally not designed with this kind of scenario in mind, we could still fine-tune it for this type of dataset (primarily made of network layer data and not application logs):

Features	Description
Working hours	Working time (*in hour*) of the user
Weekday	Working day (*from Monday to Sunday*) of the user
TRX	Number of daily transactions (*integer*)
Destination	Destination of the transaction

Protocol	Type of protocol used within the network
Meta	Type of meta elements (protocol dependency)
Att size	Attachment size (*bits*)

Fig. 3. Use Case 1 – Detailed features list as inputs for both behavior engine and multi criteria scoring engine

By fine-tuning our Anomaly Scoring Engine and building 1-month-based models for the Behavioral Modeling, our framework successfully detected 2 out of 3 behavioral incongruities that were expected to be found – as confirmed by the industrial partner who provided the dataset. The incongruities were detected thanks to unexpected system or employee behavior within the enterprise network (e.g. abnormal communication patterns between systems or between two employees):

Incongruity to detect	Overall score	Behavior detected - associated features raised
1	0.702	Medium change of trends – Meta
2	1.000	unknown equipment – all features
3	0.901	Pic of abnormal activities – TRX, protocol, META

Fig. 4. Use Case 1 – Incongruities scores ; behavior detected with associated features raised

The second dataset used for evaluation involves real application logs coming from a booking management platform used in the global travel industry. The goal was to test whether Skeptic was able to detect fraudulent bookings and functional misuse based on user sessions. A three months training dataset (Jan-March 2018) was used to let the system learn about user behaviors and create appropriate models.

The detection models were built based on a set of predefined session features that allow to monitor user activities. In the table below, we summarize the session features into categories and the associated behavioral models used for this use case:

Features	Description	Associated Models
Working Time	Working time of the user: Hour and weekday based.	Probability Histogram / Gaussian Mixture

TRX	Transaction Rate over 10 minutes, 1 hour and 1 day time windows.	Mean – Standard Deviation
Session Length	Time elapsed between the login and logoff actions	Probability Histogram
Action Categories	Categories of user performed actions. E.g. : Booking, Ticketing, Passenger details display, etc.	Probability Histogram
Action Sequences	The sequence of actions performed over time.	Probability Histogram / N-gram
Connection Office	The office from which the user initiated the session	Probability Histogram
Connection Origin	User IP address and Geo IP lookup information	Probability Histogram

Fig. 5. USE CASE 2 – Session feature categories and associated models

The detection and scoring module was then activated on the data of the following month (April 2018). The analysis was conducted on data presenting 6k requests/day, 3k users, 1.6k travel agencies, 30k IP addresses. About 10 suspicious sessions were identified per day. Among these, three out of four confirmed fraud cases (True Positives) were successfully detected thanks to abnormal connection time/origin and unusual sequence of actions. In all three cases, we could validate that the legitimate user account had been taken over by a fraudster, either via spear-phishing or some other form of malware infection (e.g. remote access trojan or key logger).

In the table below we summarize the details of the fraudulent sessions detected:

Incongruity to detect	Overall score	Detection Reason
1	0.92	Unusual Connection Origin and Action categories
2	0.87	Unusual Connection Time and Transaction rate

3	0.82	Unusual Connection origin

Fig. 6. USE CASE 2 – Detected fraudulent sessions details

A close inspection of the 4th undetected fraudulent session (False Negative) revealed that the lack of sufficient user history for building the training models is behind the failure to detect the fraudulent user activity.

The anomalous sessions that could not be associated to confirmed frauds cases (False Positives) were inspected by security analysts and data scientists. The analysis results showed that false negatives origins fall into three categories. In the table below we categorize these false positive and provide some recommendations to address them:

Category	**Description**	**Recommendations**
Unusual legitimate sessions	This include legitimate user sessions that are unusual. As example, this can happen when the user is travelling and connecting from unusual locations and connection times, when a user connects on behalf of another user, etc.	• No changes are needed to apply to modeling and scoring models. • Filter sessions from future training model updates.
Lack of sufficient training data	This include the suspicious sessions that are due to insufficient user history used in the training phase.	• Increase the span of the training period • Include user profiling in the detection models • Assign default models that are the closest to the untrained user profile
Aggressive Scoring parameters	This include suspicious sessions due to aggressive scoring and aggregation parameters.	• Tweak scoring parameters by leveraging functional expertise.

Fig. 7. USE CASE 2 – False Negative Categories

The current detection results show the high potential of our SKEPTIC UBA system at detecting and preventing online fraud in real complex applications. Moreover, improvements are continuously made to fine tune the SKEPTIC performance and implement the recommendations from Security Analysts.

5. Conclusions and Future Work

We have presented in this paper a novel *User Behavior Analytics* (UBA) approach that reinforces application security by profiling users' activities thanks to an unsupervised learning method and by scoring new users' sessions according to baseline user behaviour models. Thanks to a multi-criteria aggregation method, our system can provide a global *risk score* for each user session by combining anomalies identified on different session features. This risk score can be used as input to an automated intervention module to trigger different responses at the application layer according to the severity level and predefined business rules. An experimental validation on two different datasets (one synthetic and one real dataset from the global travel industry) have demonstrated that our system can successfully detect real suspicious events or abnormal sessions, and can be used to prevent online fraud scenarios that involve user impersonation, account take-over or credential compromise.

Further extensions to our approach include integrating an expert knowledge-based detection module (e.g. rule engine), which will be used to detect some predefined application misuse scenarios and known malicious activities. Such rule-based detection engine is suitable for integrating expert knowledge of the application monitored, and is particularly useful for harnessing community rules for known attacks (e.g. [13] OWASP Core Rule Set). As part of this future rule engine, we will also provide support for *fuzzy logic* rules, which can be useful for expressing uncertainty in dynamic rules that are evaluated using continuous functions and serve as input to an inference engine.

6. Acknowledgment

The research conducted in this paper was partially funded through the European Union project DiSIEM [15] which aims at enhancing existing SIEM systems with diversity-related technologies. More specifically, the project aims to (i) enhance the quality of events collected using a diverse set of sensors and novel anomaly detectors, (ii) add support for collecting infrastructure-related information from open source intelligence data available on diverse sources from the internet and, (iii) create new ways for visualizing the information collected in the SIEM.

7. References

[1] CyberSource, "2017 Online Fraud Benchmark Report," 2017. [Online]. Available: https://www.cybersource.com/content/dam/cybersource/2017_Fraud_Benchmark_Report.pdf. [Accessed July 2018].

[2] Gartner, "Market Guide to Online Fraud Detection," 2018. [Online]. Available: https://www.gartner.com/doc/3849295/market-guide-online-fraud-detection. [Accessed July 2018].

[3] SecurityWeek, "Credential Stuffing: a Successful and Growing Attack Methodology," 17 January 2017. [Online]. Available: https://www.securityweek.com/credential-stuffing-successful-and-growing-attack-methodology. [Accessed July 2018].

[4] DarkReading, "Credential-Stuffing Attacks Take Enterprise Systems By Storm," January 2017. [Online]. Available: https://www.darkreading.com/attacks-breaches/credential-stuffing-attacks-take-enterprise-systems-by-storm/d/d-id/1327908. [Accessed July 2018].

[5] Gartner, "Market Guide for User and Entity Behavior Analytics," 2018. [Online]. Available: https://www.gartner.com/doc/3872885/market-guide-user-entity-behavior. [Accessed July 2018].

[6] NuDataSecurity, "The Future of Fraud Prevention - User Behavior Analytics," 12 May 2015. [Online]. Available: https://www.bankinfosecurity.com/whitepapers/future-fraud-prevention-user-behavior-w-1628. [Accessed July 2018].

[7] V. Chandola, A. Banerjee and V. Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys (CSUR),* vol. 41, no. 3, 2009.

[8] A. A. Ghorbani, L. Wei and M. Tavallaee, Network Intrusion Detection and Prevention: Concept and Techniques, Springer, 2010.

[9] S. Omar, A. Ngadi and H. Jebur, "Machine Learning Techniques for Anomaly Detection," *International Journal of Computer Applications,* vol. 79, no. 2, pp. 33-41, 2013.

[10] M. Goldstein and S. Uchida, "A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data," *PLoS ONE ,* vol. 11, no. 4, 2016.

[11] A. Leman, T. Hanghang and K. Danai, "Graph-based Anomaly Detection and Description: A Survey," *Data Mining and Knowledge Discovery,* vol. 29, no. 3, pp. 626-688, 2015.

[12] Y. Freund and R. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences,* vol. 55, no. 1, pp. 119-139, 1997.

[13] V. Torra and Y. Narukawa, Modeling Decisions, Springer, 2007.

[14] G. Beliakov, A. Pradera and T. Calvo, Aggregation Functions: A Guide for Practitioners, Springer, 2007.

[15] FFCUL, "DiSIEM Project," [Online]. Available: http://disiem-project.eu/. [Accessed June 2018].

[16] I. G. a. F. J. Anscombe, "Rejection of Outlier," *Technometrics,* vol. 2, pp. 123,147, 1960.

[17] P. G.-T. a. J. E. D.-V. J. M. Estevez-Tapiadpor, "Stochastic propocol modelling for anomaly based netwrok intrusion detection," *Information Assurance,* 2003.

[18] J. Antti and S. Tuomo, "Anomaly Detection Framework Using Rule Extraction for Efficient Intrusion Detection," *CoRR,* 2014.