softserve experience matters

Anomaly Detection – Unsupervised Approach

www.softserveinc.com

white paper

Anomaly Detection – Unsupervised Approach

As a rule, the problem of detecting anomalies is mostly encountered in the context of different fields of application, including intrusion detection, fraud detection, failure detection, monitoring of system status, event detection in sensor networks, and eco-system disorder indicators.

Fig.1. Anomaly Detection

The urgent need to solve the problem of **anomaly detection** lies in the fact that any deviation from a general picture showing the current state of the system may carry important information about the existing issue. Ignoring deviations may lead to undesirable outcomes: for example, an unusual blackout on an X-ray image may serve as evidence of cancer. Prepare and prevent, they say.

In today's data driven world, Information Security attracts special attention when it comes to pattern changes detection. The continuous development and complexity of information processing automation presupposes the decisive role of security in information technology. **2015 was especially rich in cyber-attacks** with companies such as T-Mobile, Kaspersky, and Anthem having had their security compromised and all sorts of personal information about users exposed.

Having a close look at the informational environment of any organization, it's not hard to see that applying traditional safety measures is a rather prohibitive approach, and often ineffective. Since there is a whole range of possible scenarios in a user's workflow within the security system, basic rules face multiple exceptions, reducing preventive protection and making the regular analysis of inner threats identification more complicated. Detecting external attacks is also becoming more and more problematic since attackers are aware of typical intrusion detection means and apply covert agents for the attack. Here are main types of the network security breach:

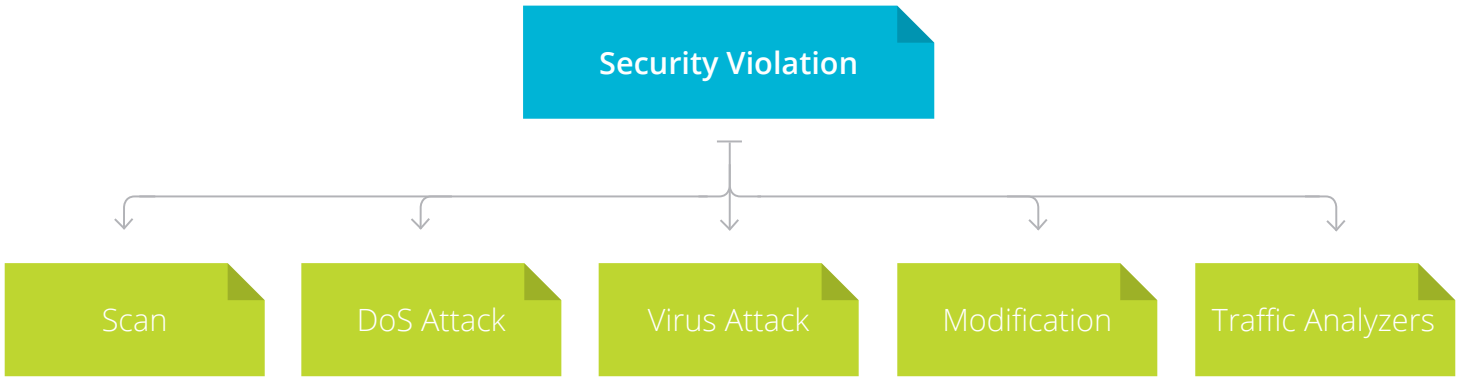


Fig. 2. Network Security

These processes may be spotted, for instance, due to the increased activity of certain ports, new unusual services, changes in a user's work with network resources, etc.

One possible solution to this problem is the development of systems that identify unusual user network behavior, based on analysis of network

activity logs. Using data mining techniques, these systems reveal indicative behavior patterns and draw conclusions about behavior that differs from what's considered conventional. The systems may though be self-adaptive, minimizing **human involvement** in configuring the system. Without taking into account an organization's specifics, such systems are of particular interest to specialists in the field of machine learning and data mining.

In this paper our Data Science Group (DSG) describes informational security risk identification by detecting deviations from the typical pattern of network activity.

Anomaly Detection – What's in a Name

Following its etymology, an anomaly is any deviation from a rule. Consequently any violation of the standard behavior traced in historical data can be interpreted as an anomaly. However, pattern violation may be both known in advance and established as a result of analysis. In this context, any problem related to the detection of non-standard behavior leads to a search of the underlying condition (baseline) and the classification of each event is recorded in the system as corresponding to or inconsistent with the found prototype.

Talking about information security, this task may be accomplished by means of analyzing user activity and network equipment to detect irregular behavior in the external and internal network traffic (analysis of **Netflow** logs) which may serve as a signal of both internal activity and an attempted external attack. So, an anomaly is *any event which may be estimated as statistically impossible in accordance with analysis results of the network activity protocol*. Here are three types of detected anomalies:

1. Significant deviation of the observed values from the expected value
2. Failure in the process reflecting the change of the measured parameter within the surveillance area
3. An atypical set of observed values in the measured parameters

Standard **Netflow** logs contain the following set parameters:

1. Ingress interface (SNMP ifIndex)
2. Source IP address
3. Destination IP address
4. IP protocol
5. Source port for UDP or TCP, 0 for other protocols
6. Destination port for UDP or TCP, type and code for ICMP, or 0 for other protocols
7. IP Type of Service

Interpretation of the network protocol is not available here, so we don't have information on which recorded events can be described as the norm, and which events are considered as a deviation. Therefore, to detect anomalies, the network activity protocol analysis should possess the following features:

1. Collection and storage of network activity results data
2. Representation of network activity in the form of numerical characteristics series, supplemented by non-numeric attributes (qualifying factors), including time markers
3. Identification of hidden patterns in data which provide the basis for formation of the behavior pattern
4. Evaluation of new observations for pattern matching

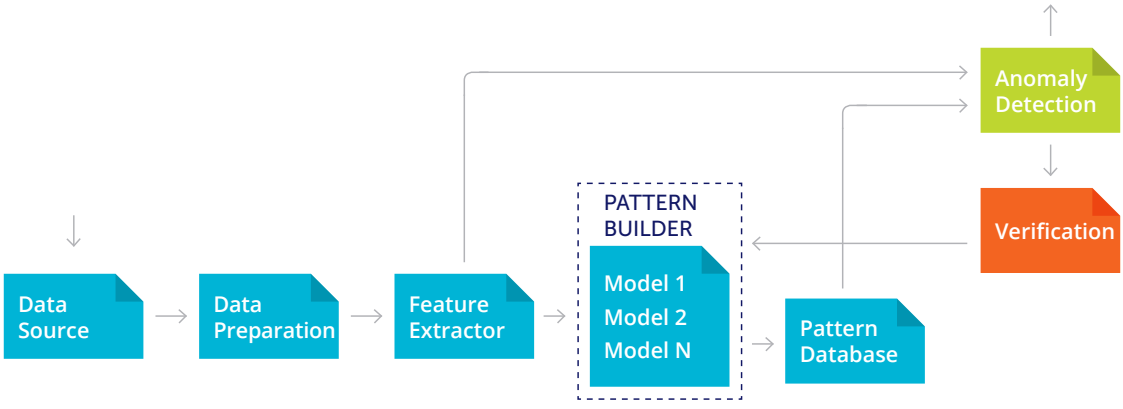
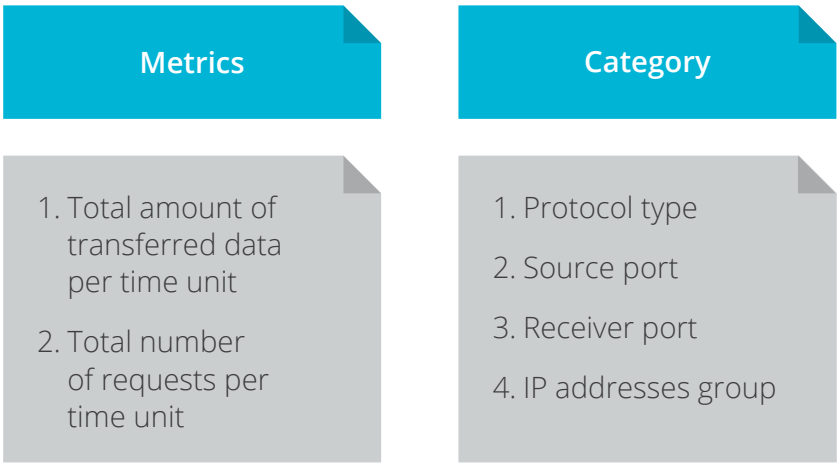


Fig. 3. Anomaly Detection - Infrastructure

The core of the proposed system lies in the ensemble of models, each of which allows estimation of the average statistical activity of users (or groups of users) and classifies the observations recorded in the system as corresponding to normal or abnormal. The following metrics and measurement categories were used as the data source for constructing models:



According to the anomaly types described above, we tried to combine three different models: on one hand, to evaluate the nature of the metrics change within different categories considering the accumulated data as a process extended in time; on the other hand, to analyze the value of metrics in different categories together for the isolated period of time.

1. **Dynamic Threshold Model** allows an accepted level of assignment (measured value) to be set for specific time intervals (time of a day, weeks, months, etc.). Subsequently, each observed value is evaluated for compliance with the threshold, making it possible to identify cases of the anomalies appearance. This model is a solution used to detect the first type of anomalies.

2. **Association Rules Based Approach** allows the activity of the network to be described as an unrelated collection of events, and as a consequence, presenting it in the form of a stochastic process. Any event that is characterized by low probability can be interpreted as an anomaly.

3. **Time Series Clustering** allows the identification of common regularities in the time series structure, capturing any deviation from the established pattern at the same time (fitting the second type of anomalies).

Dynamic Threshold Model

What lies at the heart of a Dynamic Threshold Model is representation of a user's activity expressed as a set of measured parameter values (metrics) for a certain observation category through a time series. It allows the establishment of presence or absence of general tendencies that can be expressed in the form of a general trend or have a seasonal character. For example, we can consider a time series that reflects a week of network activity of group members through the total amount of data transmitted in one minute (Fig. 4).

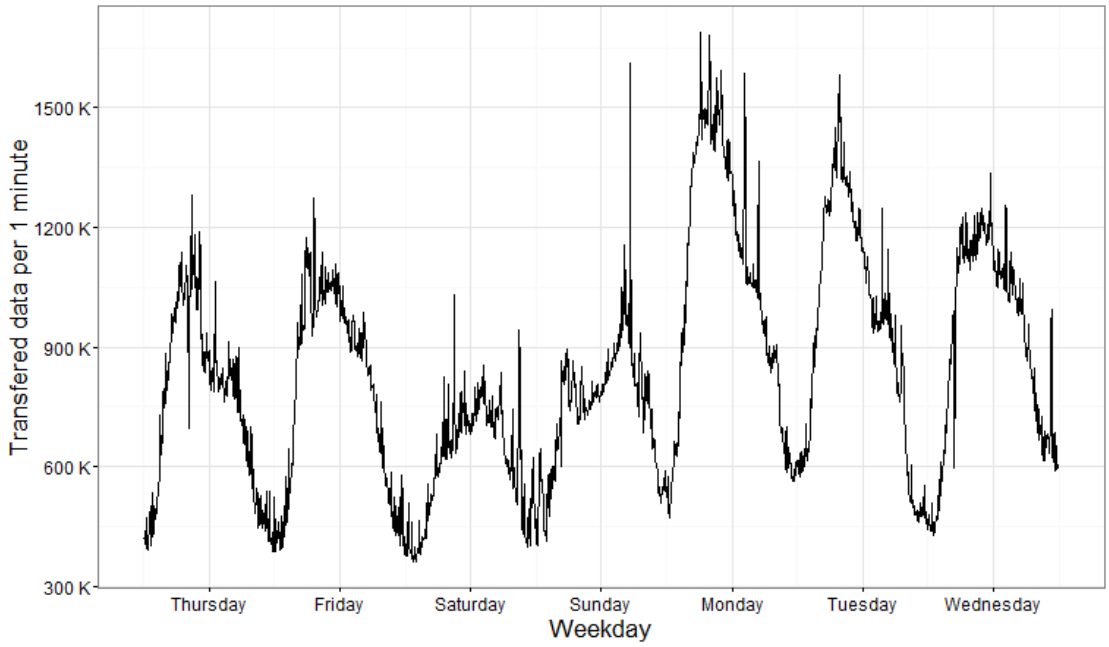


Fig. 4. Time Series Example

There is a common trend within this time section: through the week the average user activity decreases. At the same time, the process cannot be considered as proportional: there is a periodic sequence related to the fact that the maximum and minimum activity fits in the middle of the employees' work day and night-time correspondingly. In this way each day's activity can be described by a common pattern with minor amendments from the primary trend. Then, user activity may be described in the form of a discrete time series (signal quantized in both time and amplitude), along with evaluating a potential sampling error (allowed deviation threshold) (fig. 5). If the replacement analogue signal with a discrete error exceeds the permitted deviation threshold, the situation can be evaluated as an anomaly.

The next step is setting the maximum network load threshold for each part of the time series. The main reasoning behind this is to have average user activity reflected in both series trending and seasonal components, with the changes in load behavior taken into account.

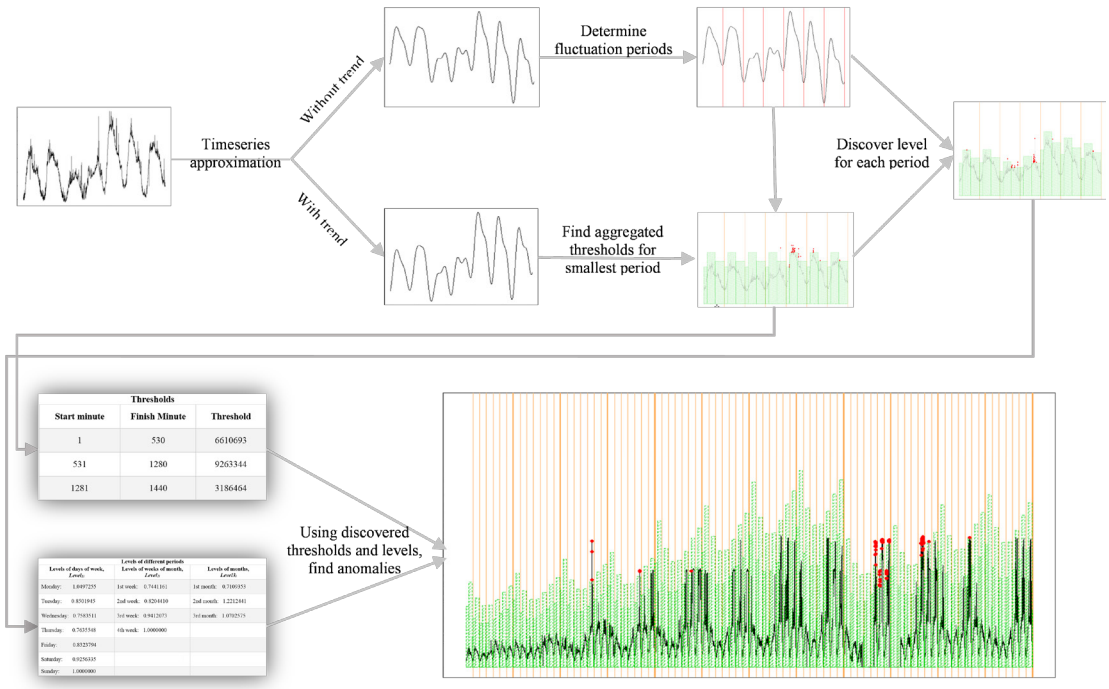


Fig. 5. Flow of Anomaly Detection based on user activity

The model shown allows the description of user activity in the network through a limited set of constants. This set can be obtained at the initial stage of system adjustment and doesn't require the entire set of historical data analysis "on the fly" to operate.

In addition, the model may be easily adapted to the changing process. If significant deviations of the discrete time series from the continuous one become more frequent, it is enough to merely adjust the threshold values and/or coefficients of the relative load.

Association Rules based Model

As mentioned before, the process being monitored can be represented as a set of linked events. Speaking about the activity of the user and/or group of users in the network, consider a single metric one-time value calculated for the category and/or a several categorical variables combination. For example, for sampling step time in 1 minute, estimate a total number of transmitted packets by a department for 1 minute on a TCP/IP protocol in the first half of the work day and at the same time estimate the total amount of packets via UDP protocol for the same user group.

We are not really interested in the absolute value of each of these categories, but rather in the relative number of cases when this activity was recorded.

For example, let's take a time series which reflects a needed parameter

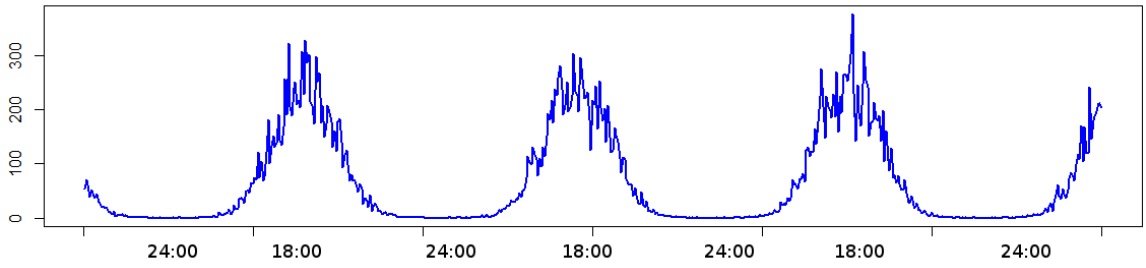


Fig. 6. Metric representation example

change in time. In most cases, the results of this measurement will be presented by a continuous variable, i.e. it takes an infinite number of values. Each value is unique within a quantization scale, which makes it possible to estimate the contribution of each individual observation in the common behavior pattern formation.

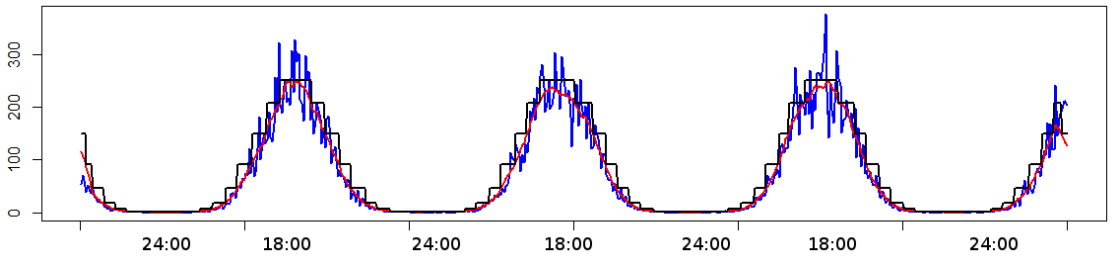


Fig. 7. Signal discretization

However, it's not difficult to see that the measured value lies within certain discrete ranges, the sequence of which can be traced over time. Thanks to quantization, this fact facilitates shifting from a continuous representation to a discrete one. Limit the number of measured values by a finite set breaking the entire values range to the equal density areas:

Taking the time factor (T) into account, it becomes possible to allocate those discrete random variable values that are highly unlikely for a specific time line. So, in our example, the beginning and end of the working day are characterized by discrete random variable values from the set {6, 18, 46}, while the range from 12:00 to 18:00 is characterized by a greater load:

	0.2	0.5	2	6	18	46	92	150	208	250
0-6	0,42	0,17	0,17	0,17	0,07					
6-9					0,3	0,35	0,35			
9-12							0,14	0,43	0,38	0,05
12-18							0,1	0,2	0,2	0,5
18-21					0,35	0,45	0,2			
21-24	0,14	0,43	0,38	0,05						

With this approach, the discrete random variable value (the analyzed parameter) defined by a small occurrence possibility within any time axis area (time slot) means getting out of the overall pattern, and consequently – the anomaly. Thus, for the time interval from 9:00 to 12:00 we get:

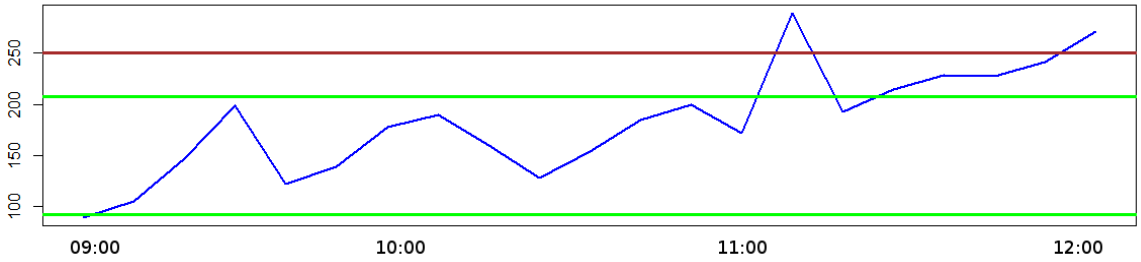


Fig 8. Possible value boundaries

Such interpretation makes it possible to simultaneously analyze measured parameter values for different categories; according to experimental data it may be established that the value of one category suits for adjusting metric values for the other**.** This means that there are common combinations of values in both categories. In this case, the information enhances certain pattern value. The more categories a frequent combination combines, the higher the detecting pattern probability in the observed process is, and the higher the deviation from it appears.

Let's say, as based on the results of observations within one month, the following activity was recorded in two different settings (M1 and M2):

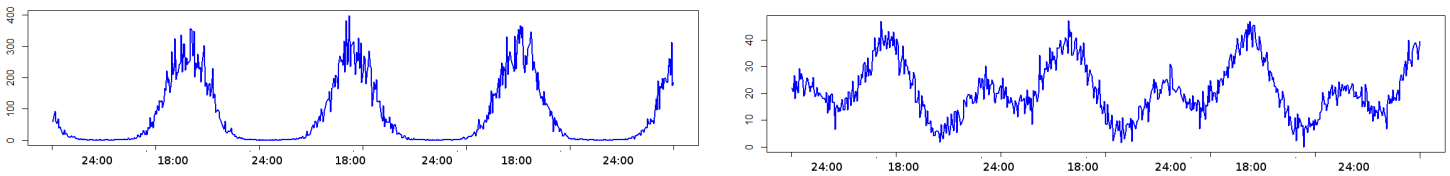


Fig 9. Observation results for two metrics

Dependence between individual values of the observed value can be represented as a graph, where nodes are random events (system's state). Each event corresponds to a certain metric value or combination of several values, and link weights reflect their conditional probabilities (Fig. 10 -left). As seen in the graph, the system behavior is described by a pattern set that can be found in each observation (system's state). This graph can be simplified by means of small weight connection leveling and path reduction with a low total weight (Fig. 10 -right). As a result, we get a set of rules that define normal system behavior.

The problem of obtaining user activity patterns in the network faces the problem of detecting associative links between individual states of the system described above, which form associative rules that describe the required patterns. This approach makes it possible to use known methods of mining association rules in order to recreate the average user activity, in particular the Apriory algorithm.

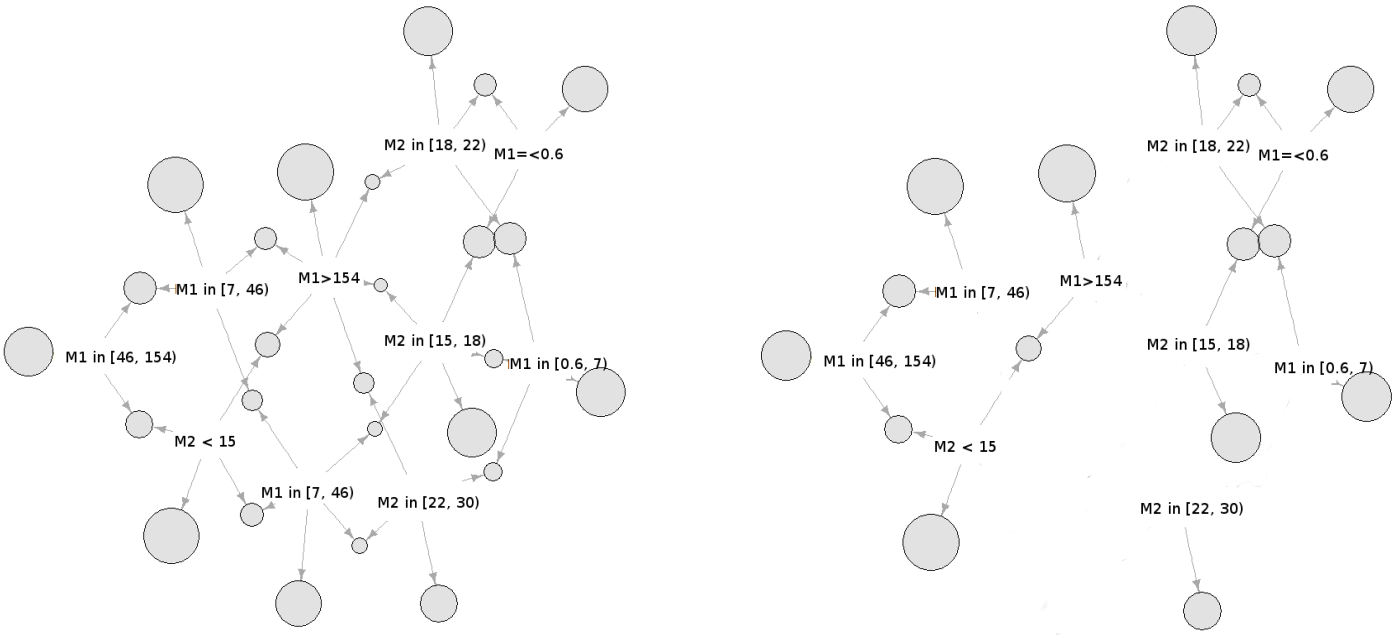


Fig. 10. Stochastic Model

Each of the rules selected is a component of the required user behavior pattern: the bigger the number of rules for observation gets, the more significant the pattern may be considered, and the more likely it will characterize the expected behavior standard. As the observation rank, we used a normalized meaning of the weighted rules taking into account their lift, rule's performance at predicting cases measured against a random choice. So for this example, we get:

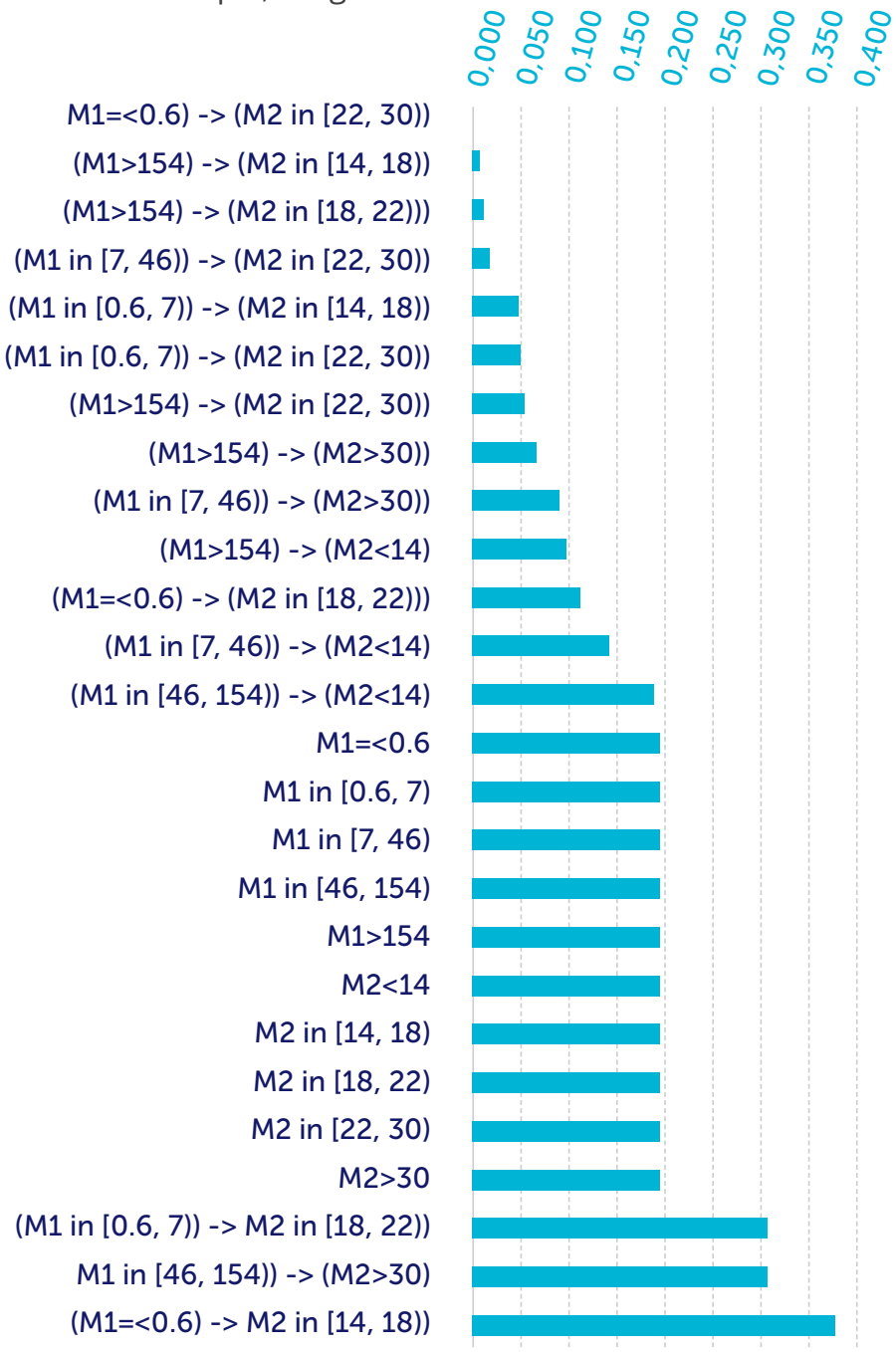


Fig. 11. Rules importance

The lower the observation rank, the more reasons there are to believe that it is covered by the definition of anomaly.

The developed model allows several factors to be taken into account in assessing user activity, as well as describing it as a finite rules set, each of which is characterized by a certain level of significance in the found patterns system. Similar to the previous model, it can be used to analyze the process online, and adapted to the changing process due to the revaluation of the detected rules weight and/or by means of adding new ones.

Time Series Clustering

The last model is based on the time series partition method which allows a set to be presented as a discrete segments ensemble that forms separated groups (clusters) defined by the fact that time series components from the same cluster are close to each other, while the fragments from different clusters are distinct. Here's an example of time series analysis:

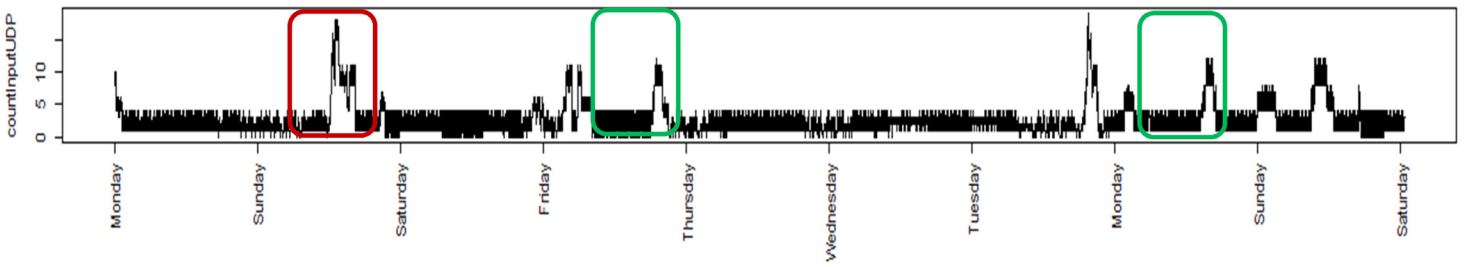


Fig 12. Time series fragments similarity

Line fragments highlighted in green have more in common than a red-contoured one. In this case, they may be referenced to one group. To estimate the time series proximity degree (similarity), it is possible to use a timeline dynamic transformation algorithm (DTW) which facilitates the finding of optimal conformance by using time series fragments. This method was used in speech recognition. The process of time series segmentation presupposes formation of fragments plurality from this line by passing it with a floating window of a given size and with a certain sampling interval.

In general terms, Fig. 13 depicts the basic process of the model in question, i.e. detection of the time series abnormal dynamics.

The result of the pairwise comparison of the obtained fragments through a DW-distance evaluation forms the adjacency matrix which reflects the proximity degree between the fragments. The obtained matrix may be used as a basis for solving the issue of fragments clusterization.

The clusterization result of the first 13,000 minutes (80%) for three metrics (countInputUDP, countInputTCP and averagePacketSizeOutputUDP) is shown in the figure below. Fragments of the time series are displayed in the space of two principal components, while different colors distinguish components

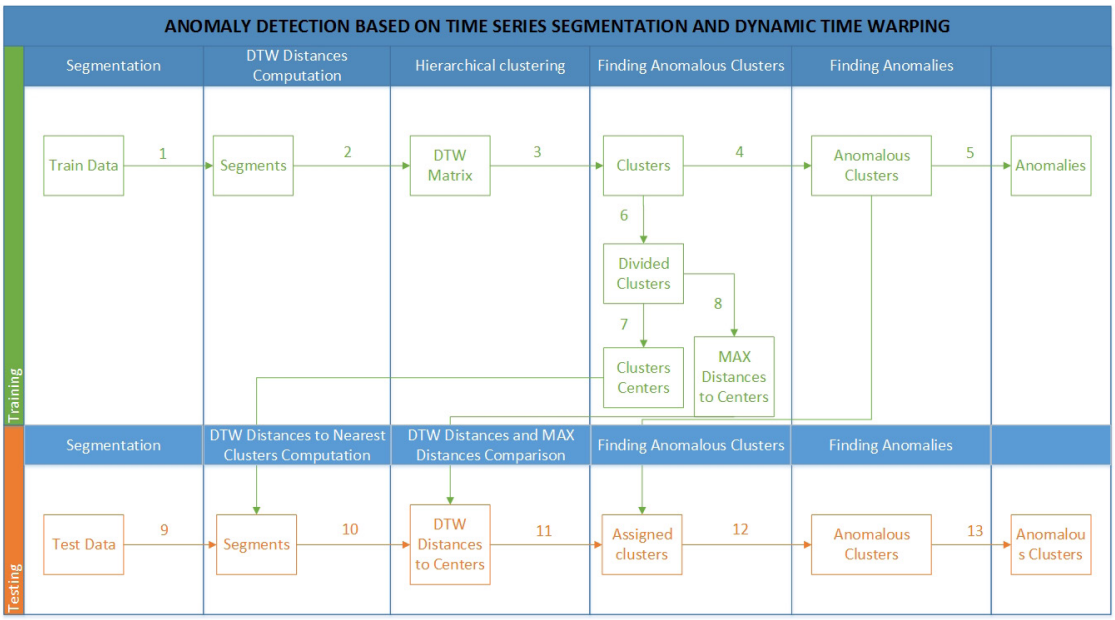


Fig 13. Abnormal dynamics detecting process

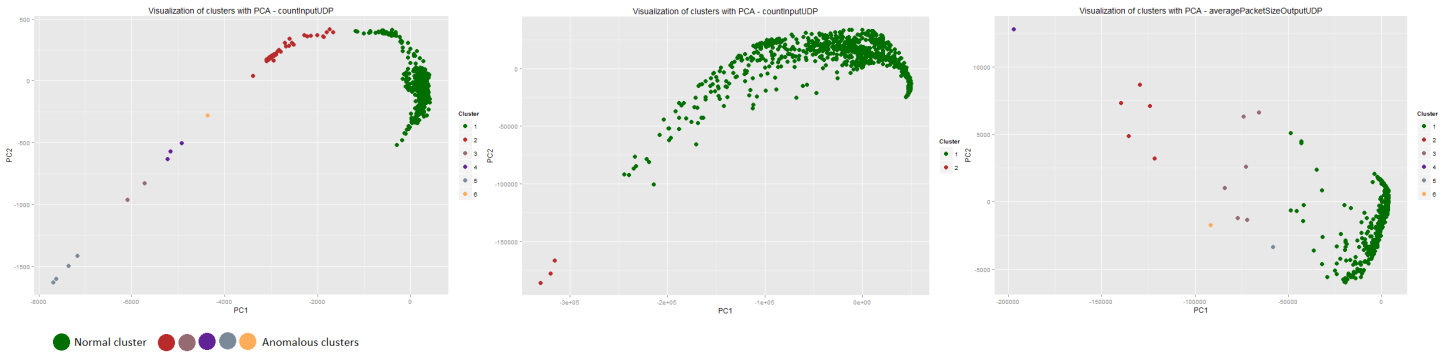


Fig 14. Clusters representation

from different clusters. In each case a single large cluster is formed out of the time series fragments that constitute a common behavior pattern. Clusters of small size can be interpreted as an anomaly.

Clusters with small amounts of components are related to dynamics changes, which isn't typical for the majority of cases. The level of these deviations abnormality depends on a relevant set power. At the same time, the significance of anomaly in a certain period of time can be determined by the number of metrics that were analyzed to identify the period.

The obtained model allows network activity to be estimated through the dynamics of the measured parameter changes and/or measured parameters system, which may help identify abnormality even if the absolute values lie within acceptable ranges. The only drawback it poses is the complexity of usage "on the fly" because its implementation requires calculation of the similarity degree of the new observations with the existing clusters elements.

Models Ensemble

To maximize the effectiveness of the models used for the information security violations detection, our Data Science Group decided to unite them into one ensemble (Fig. 15). As seen from the diagram, the first two models (Dynamic Thresholds and Association Rules) use the same data set related to measurements according to N categories received in real time. This approach may be referred to a begging model where the role of arbiter is performed by an *Anomaly Confidence Level* unit reestablishing the level of certainty that abnormal network activity takes place in the given time period. Meanwhile, the Time Series Clustering model works with the pool of historical data which presupposes its inclusion in the model according to the boosting method.

Because of the differing usage modes between the models (online and offline), the model-based time series segmentation is the most applicable in non-business hours to clarify the situation when an anomaly hasn't been detected. According to the offline verification results and further verification, the models are to be adjusted to the

new conditions, which leads to a change in a range of normal behavior patterns and, as a consequence, allows the observed process to be described with more accuracy.

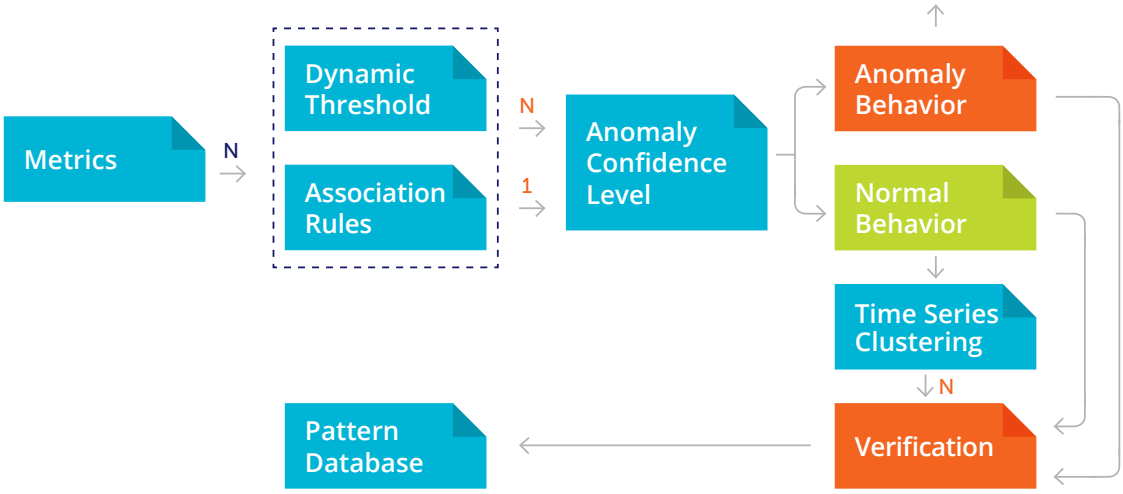


Fig 15. Ensemble of models

Conclusions

Within the framework of the conducted research, our Data Science Group came up with a comprehensive solution to detect differing network activity of users or group of users as opposed to a well-known pattern, which in its turn may indicate attempts of an information security breach. The offered solution is an ensemble of three models that facilitate analysis of three anomaly types:

1. Significant deviation of the observed values from the expected – **Dynamic Threshold Model**. Simplicity in implementation and its ease of adaptation are the main advantages of the model; on the other hand, the isolation of the analysis results for each individual metric from observations in other categories appears to be its downside, which makes it difficult to search for event patterns.

2. Unusual set of the observed values of the measured parameters – **Association Rules Based Model**. The main advantage of this model is its ability to describe the observed process as a set of related events; insensitivity to weak process dynamic changes is its main drawback.

3. Unusual dynamics in the observed process – **Time Series Clustering Model**. Even though the found patterns in this model reflect the internal dynamics of the observed process, it doesn't allow detection of event patterns and its application "on the fly" is extremely difficult.

This is why we suggest using an ensemble model, since combined they neutralize all the disadvantages and facilitate decision-making regarding adaptation to the modified conditions. As a result, this solution allows both typical abnormal network activity manifestations to be identified and unusual and new elements of the network anomalies to be detected, while the self-adjusting capability lets the solution adapt to the "legal" changes in the network processes.

About the Authors

Tetiana Gladkikh is a Data Scientist at **SoftServe.** She has 19 years of experience in Research, with 15 years being dedicated to Data Mining and Computational Intelligence. The main areas of Tetiana's scientific interests are Data Mining, Artificial Intelligence (Genetic Algorithms, Neural Network, Fuzzy Logic), Mathematical Statistic, Computer Vision, and Methods of the verification computer devices. She holds a PhD in Technical Sciences (Elements and Devices of Computer Technic).

Taras Hnot is a Data Analyst at **SoftServe.** He has a Master's Degree in Economic Cybernetics with a strong knowledge of statistical modeling and data mining algorithms. His main interests are Statistical Learning, Predictive Analytics, Time Series Analysis, Artificial Intelligence and Recommender Systems. Taras has experience in development of anomaly detection systems, analyzing and detecting patterns of huge payment networks, implementing different types of algorithms in order to build computer vision systems.

Volodymyr Solskyy is a Data Scientist at **SoftServe.** He has 9 years' experience in commercial software development, with 5 years being dedicated to Scalable Architectures, Cloud Technologies and Machine Learning. His main areas of interest are Distributed Systems, Network Analysis, Information Theory and Knowledge Extraction. Volodymyr obtained his Master's degree in Engineering (Theoretical Mechanics) at Ivan Franko National University of Lviv, Ukraine.

About SoftServe

SoftServe is a leading technology solutions company specializing in software development and consultancy services. Since 1993 we've been partnering with organizations from start-ups to large enterprises to help them accelerate growth and innovation, transform operational efficiency, and deliver new products to market.

To achieve this we've built a strong team of the brightest, most inquiring minds in the industry, and we form close, collaborative relationships with our clients so we can really understand their needs and deliver intuitive software that exceeds their expectations.

Our experience stretches from Big Data/Analytics, Cloud, Security and UX Design to the Internet of Things, Digital Health and Digital Transformation, we have offices across the globe and development centers across Eastern Europe. For more information please visit **www.softserveinc.com**.

USA HQ
Toll Free: 866-687-3588
Tel: +1-512-516-8880

Ukraine HQ
Tel: +380-32-240-9090

Bulgaria
Tel: +359-2-902-3760

Germany
Tel: +49-69-2602-5857

Netherlands
Tel: +31-20-262-33-23

Poland
Tel: +48-71-382-2800

UK
Tel: +44-207-544-8414

EMAIL
info@softserveinc.com

WEBSITE:
www.softserveinc.com

softserve experience matters