Network-wide intrusion detection supported by multivariate analysis and interactive visualization

Roberto Theron*
Department of Computer Science and Automation
University of Salamanca - Spain

Roberto Magán-Carrión†
Dpt. of Signal Theory, Telematics and Communications - CITIC
University of Granada - Spain

José Camacho ‡
Dpt. of Signal Theory, Telematics and Communications - CITIC
University of Granada - Spain

Gabriel Maciá Fernndez §
Dpt. of Signal Theory, Telematics and Communications - CITIC
University of Granada - Spain

Abstract

In this paper, we introduce a new visualization tool for network-wide intrusion detection. It is based in multivariate anomaly detection with a combination between Principal Component Analysis (PCA) and a new variant called Group-wise PCA (GPCA). Combining these methodologies with the capabilities of interactive visualization, the resulting tool is a highly flexible and intuitive interface that allows the user to navigate through the enormous amount of data collected in the network, in order to find anomalous or unexpected behaviors. We use a real case study to illustrate the capability of the tool to unveil the complex mixture of information that can be found in network security/traffic data and identify and diagnose anomalies in it.

Index Terms: [Networks]: Networks security [Human-centered computing]: Visual analytics [Human-centered computing]: Visualization systems and tools

1 Introduction

The problem of anomaly detection in network traffic has been recurrently studied in the literature. In particular, the use of Principal Component Analysis (PCA) for anomaly detection was proposed more than a decade ago [13, 20]. PCA is a matrix decomposition technique that separates the structural part of the data from the residuals in terms of variance. The common trend for PCA anomaly detection [16] is to build the PCA model from the complete data under monitoring, and look for anomalies in the residuals. The underlying assumption is that the structural correlation captured by PCA represents the normal, free of anomalies, traffic behavior. This assumption in fact leads to the main shortcoming of the approach: anomalies of large magnitude, and therefore of large variance, can pollute the normality model. This, in turn, makes the approach very sensitive to calibration settings [18].

In this paper, we propose an anomaly detection tool based on the combination of interactive visualization and multivariate analysis. The considered multivariate analysis technique is based in the combination of PCA and a recently proposed modification of PCA, referred to as Group-wise PCA (GPCA) [5]. With GPCA, we can identify anomalies in the structural part of the model following a straightforward methodology, simple to use and understand by security professionals not trained in multivariate tools. Since anomalies are detected in the structural part of GPCA, this approach does not suffer from the same shortcomings of the detection in residuals.

1.1 Related Work

There are several works in the literature that integrate computational intelligence and visualization to detect and/or react when an anomalous behavior occurs in a network. For example, in [1], a visualization tool for situation awareness of system/network, both for proactive and reactive approaches, was proposed. The proactive part helps the security analyst to understand the system configuration and their vulnerabilities, while the reactive part offers an overall and dynamical view of what is currently happening in the system and how the ongoing attacks evolve. The dataset used for testing the approach was synthetically generated to represent a critical infrastructure for power and water purification. The authors in [9] proposed an ensemble visualization where a network ensemble data consists of related and time-dependent alerts and traffic flows. They used a dissimilarity matrix of the ensemble to perform agglomerative clustering, organizing the members of the ensemble in groups according to their similarity. Afterwards, the security analyst could inspect similar groups and find anomalous behaviors in the dataset.

A visual tool that integrates visualization and machine learning techniques for network forensic in critical infrastructures is proposed in [7]. The authors were able to identify system and process-related threats at network-level and application-level, respectively. To that end, they used a probabilistic based IDS for network messages classification and derived new message attributes from the domain knowledge, and used the Pearson's Chi-Square test for the detection of contextual anomalies.

Most malware networking behavior is periodically repeated. For that reason, in [10] a Fourier transform-based periodic detector was proposed. The authors also developed an interactive visualization for the verification of the detector's results. The tool was tested using the DARPA 1999 dataset together with botnet related traffic.

In [8], Engle and Whalen used so called hive panels, which consist of a small-multiples approach where several hive plots [15] are used to convey different network properties and datasets. Engle and Whalen argue that hive plots are better suited for the discovery of visual patterns than common force-directed network layout algorithms, which are not reproducible and cannot be used to compare across networks or their statistical properties. Also, this work is also related to our approach in the sense that we do not integrate visualization in anomaly detection as a post-hoc analysis.

However, as in previous works [22] [17] dealing with the design of intrusion detection systems, or more recent ones [6] aimed at detecting anomalous behavior, the visualization suggested in [8] is used as a guide in the feature engineering step. We do not focus on feature selection as a means for reducing the dimensions of the dataset but on the the interaction of the expert with the visualizations leading to dimensionality reduction operations [19].

In comparison with previous works, two are the principal advantages of our approach:

1. We consider the interactive customization of the underlying

*e-mail: theron@usal.es
†e-mail:rmagan@ugr.es
‡e-mail:josecamacho@ugr.es
§e-mail:gmacia@ugr.es

978-1-5386-2693-1/17/$31.00 ©2017 IEEE

model in the visualization, so as to make the most of the integration of the data analysis methodology with the visual interaction capabilities of the tool.

2. We integrate detection and data understanding/diagnosis into the visualization, so as to reduce the number and impact of false positives in the data analysis approach.

Network security is a complex domain that requires, on the one hand, expertise in order to make sense of the data, and on the other hand, automated mechanisms capable of dealing with massive amounts of data. Our aim was to design an interactive tool that takes both sides into consideration.

The remaining of the paper is organized as follows. Sections 2 and 3 introduce PCA and GPCA, respectively. Section 4 introduces iGPCA, a novel interactive tool for cybersecurity that makes use of GPCA. In Section 5 we explain the workflow that a cybersecurity analyst would follow with a real case study. Section 6 presents some concluding remarks.

2 Principal Component Analysis

Principal Component Analysis (PCA) is probably the most popular multivariate analysis tool. PCA provides a factorization of data sets where M variables (or features) are measured/computed for N observations (or objects). The goal of PCA is to find the subspace of maximum variance in the M-dimensional variable space. This is done by finding linear transformations of the original variables, called principal components (PCs), which are orthogonal and explain decreasing amounts of variance in the original data. The PCA factorization follows the expression:

$$\mathbf{X} = \mathbf{T}_A \mathbf{P}_A^{\mathrm{T}} + \mathbf{E}_A, \tag{1}$$

where $\mathbf{X}$ is a $N \times M$ data matrix, $\mathbf{T}_A$ is the $N \times A$ score matrix containing the projection of the objects onto the A principal components (PCs) sub-space, $\mathbf{P}_A$ is the $M \times A$ loading matrix containing the linear combination of the variables represented in each of the PCs, and $\mathbf{E}_A$ is the $N \times M$ matrix of residuals.

The model and decomposition in eq. (1) can be used to perform anomaly detection. Matrices $\mathbf{T}$ and $\mathbf{E}$ in eq. (1) are employed to construct a normality model for both model and residual sub-spaces, respectively. For that, the following statistics, the D-st and Q-st, are computed:

$$D_n = \mathbf{t}_c \lambda^{-1} \mathbf{t}_n^t \tag{2}$$

$$Q_n = \mathbf{e}_n \, \mathbf{e}_n^t \tag{3}$$

where $\mathbf{t}_n$ and $\mathbf{e}_n$ are row vectors that represent the scores and residuals corresponding to a given observation and λ represents the covariance matrix of $\mathbf{T}$. Control limits on both statistics can be defined following [4].

One of the advantages of PCA anomaly detection is that the PCA model can be interpreted and the anomalies diagnosed [4]. However, PCA has several limitations for interpretation, in particular *i*) it does not distinguish between unique and shared variance [11], and *ii*) the principal components are a linear combination of all the variables simultaneously [12]. Hence, every component typically compresses variance for several and inter-independent groups of related variables. This greatly complicates interpretation, especially for high dimensional data [2].

The limitation of PCA for data interpretation is illustrated in Figure 1 with a simulated example. A set of 10 observations on 20 variables are synthetically generated so that groups of correlated variables can be easily identified. Figure 1(a) shows a heat map corresponding to the correlation matrix of the data. Each group corresponds to a different source of variance in the data that should be interpreted separately. For instance, a group of variables may identify a set of features related to normal or anomalous behaviors in a network. Now, let us imagine that the biggest group of variables corresponds to unwanted traffic features, like an excess of alarms security sensors, while the rest correspond to normal features. Since the first PC is a linear combination of all the variables according to the coefficients (loadings) in Figure 1(b), the scores in Figure 1(c) cannot be interpreted as a consequence of any of the sources of variance, but as an obscure mixture of those. Thus, these scores cannot be directly associated to anomalous or normal behaviors.

3 Group-wise Principal Component Analysis

GPCA is a sparse variant of PCA where, unlike in other sparse PCA algorithms, the model meta-parameters can be identified from visual inspection of the data. It sequentially solves both problems in the application of PCA to data interpretation. First, a set of groups of correlation variables are found in the data by focusing on the shared variance. Then, these groups are used to perform group-wise PCA, so that each loading vector is constrained to have non-zero values in a single group of variables. Following this approach, interpretation is simplified to a large extent, since each Group-wise PC (GPC) can be safely interpreted on an individual basis. The visualization tool proposed in this paper is based on this property.

Let us come back to the previous example. The first GPC is illustrated in Figure 2. The loading vector captures the largest group of variables, representing unwanted traffic, and the corresponding score vector represents the distribution of the observations according to it. We can see that in that score vector the observations are grouped, so that the first five observations take positive scores while the last five take negative values[1]. Then, the observations with higher scores should be inspected for these unwanted characteristics.

The GPCA framework is based on the following methods:

- The MEDA algorithm [2], which is used to compute correlation maps with minimum noise level from a PCA model.
- The Group Identification Algorithm (GIA) [5] for the identification of (possibly overlapping) groups of variables in the MEDA map.
- The group-wise PCA (GPCA) algorithm [5] to identify the GPCs.

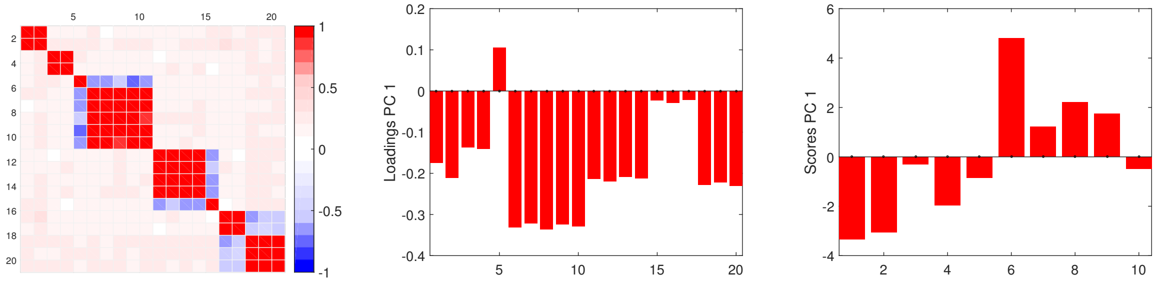
To calibrate a GPCA model, we need to set values for two meta-parameters. On the one hand, the number of PCs in the PCA model to compute MEDA, which can be obtained from the visual inspection of the variance captured per number of PCs. On the other hand, a threshold for groups selection, the γ parameter, that can be set from visual inspection of the MEDA map. Both selection procedures are integrated into the proposed tool. For more information on GPCA, please refer to [5].

4 iGPCA: interactive GPCA analysis

As already mentioned, combining computational intelligence and information visualization techniques helps the analyst to discover new insights and knowledge in datasets. Embracing this idea of an effective integration of both parts, we have designed and developed the tool called iGPCA (interactive GPCA analysis). iGPCA fosters the security analyst to dive into the data for finding suspicious behaviors. Moreover, iGPCA includes the main steps/phases of a dataset analysis task: from the preliminary inspection of the original data to the anomaly diagnostic. Furthermore, iGPCA is also aligned with the so-called visual analytics mantra (”Analyze first, show the important, zoom, filter and analyze further details on demand [14]).

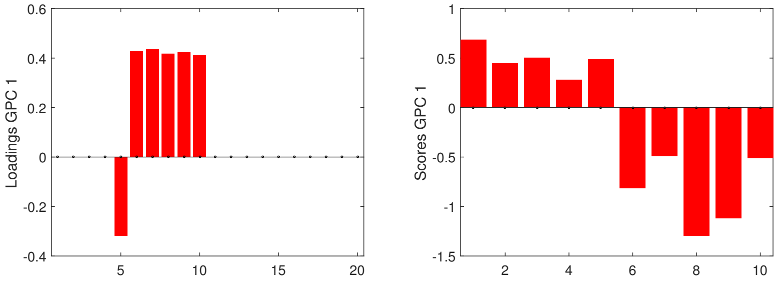
It should be noted that the information captured from a network is usually presented in the form of system logs or network traces, and

[1] Notice the data is mean-centered, so that scores are centered in zero.



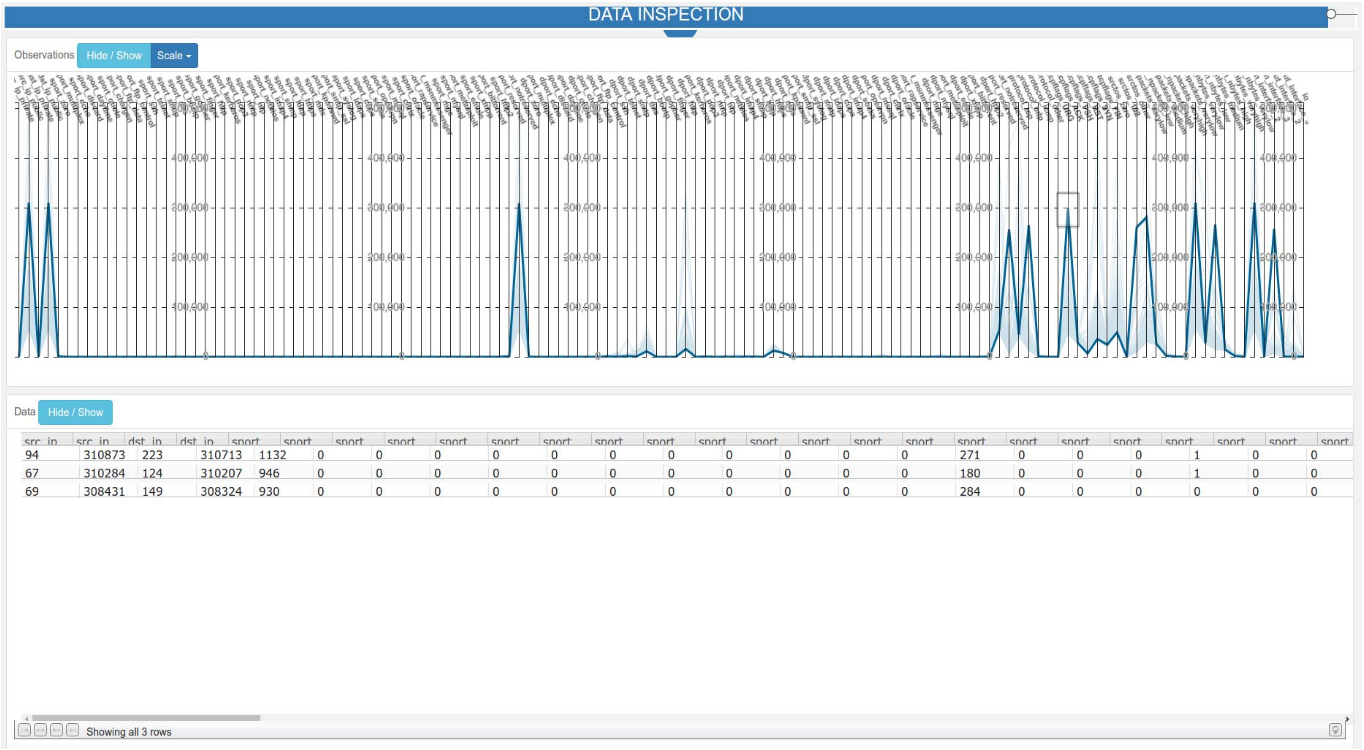
(a) (b) (c)

Figure 1: Illustrative example: (a) correlation matrix of multivariate data set, where a set of groups of variables is apparent, (b) first loading vector of PCA, with weights from all variables and (c) corresponding score vector.



(a) (b)

Figure 2: Illustrative example: the loading vectors in GPCA (a) and corresponding score vectors (b).



src_in	src_in	dst_in	dst_in	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport	sport
94	310873	223	310713	1132	0	0	0	0	0	0	0	0	0	0	0	271	0	0	0	1	0	0			
67	310284	124	310207	946	0	0	0	0	0	0	0	0	0	0	0	180	0	0	0	1	0	0			
69	308431	149	308324	930	0	0	0	0	0	0	0	0	0	0	0	284	0	0	0	0	0	0			

Figure 3: Data inspection. The user can see all the data used for the GPCA model and can interact with it and identify suspicious observations that are different from the rest.

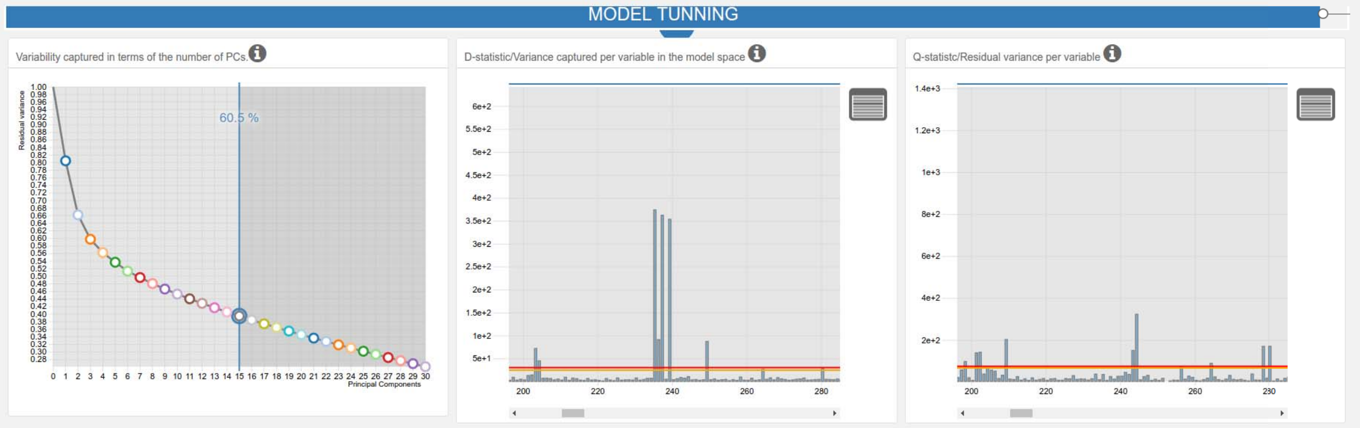


Figure 4: PCA analysis. PCs selection and preliminary detection of the anomalous observations.

cannot be directly used to feed a typical tool for anomaly detection. Therefore, some sort of parsing and feature engineering needs to be done in order to generate quantitative features that can be used for data modeling. In the context of anomaly detection with PCA, Lakhina et *al.* [16] proposed the definition of counters obtained from *Netflow* records as quantitative features. In [3], we generalized this definition to consider several sources of data, proposing the feature-as-a-counter approach. Each feature contains the number of times a given event –*e.g.* the apparition of a word in a log or of a traffic flow involving a given port in a *Netflow* file– takes place during a given time window. This general definition makes it possible to handle, in a suitable way, most sources of information in anomaly detection. Both raw and parsed data conform the input to iGPCA.

Bearing the previous steps in mind, iGPCA is divided in four interactive parts:

1. **Data inspection**. A preliminary approximation to the parsed data may help the security analyst to find out suspicious behaviors on certain observations and/or features. For example, observations whose features values are higher or lower from the rest, can be indicative of an unusual/anomalous behavior.

2. **PCA Analysis**. Building a model from the parsed data is a key step that could compromise the efficacy and accuracy of the data analysis. For instance, selecting a small number of PCs in PCA and GPCA models could leave out relevant information from the data. Moreover, outliers detection and removal could increase the model accuracy. For that, we use PCA and D-st and Q-st control charts.

3. **GPCA Analysis**. Commonly, an unusual behavior in the data involves more than one feature. Aiming at identifying the groups and relevance of these variables, MEDA and GIA tools are used over the parsed data. The former computes the correlation among variables, while the latter is in charge of establishing groups of variables in MEDA and their security relevance. Additionally, as a contribution to the determination of the behavior of a group of suspicious variables, GPCA obtains the corresponding set of scores and loadings for future inspection. For instance, an anomaly occurs when a score lies over or below the control limits.

4. **De-parsing**. The last step of the approach is the selection of the specific registers or logs in the raw data that are associated with the anomaly. For that, we use both the scores and loadings from GPCA. The scores provide the timestamp for the anomaly, which can be one or a set of consecutive sampling intervals. The loadings provide the features associated with the anomaly. Using the same regular expressions employed during parsing of the selected features in the selected interval, we can identify the related raw logs. This information is then presented to the security analysts. We call this step de-parsing because, from the observations and features of the parsed data, we recover the associated raw data, just the opposite operation to parsing. Once the raw information is gathered, it is visualized as a means of anomaly verification. Obviously, which visualization technique is appropriate depends on the type of the data source and its characteristics, *i.e.*, whether it is structured data or not, the data source itself, among others.

Although the aforementioned steps in iGPCA may offer useful tools for anomaly detection and diagnosis, the strengths of iGPCA rely on the interactive approach, in which the human capabilities and expertise of the analyst can make a significant difference.

5 Towards anomaly detection with iGPCA

In this section we present a real case study for illustrating the proposed visual methodology. First, we introduce and describe the used dataset and second, we present an example of use of iGPCA in order to investigate, analyze and diagnose a specific anomaly.

5.1 Dataset description

The dataset used in this example of visual analysis is the UGR'16 dataset[2]. It consists of a trace with a duration longer than 4 months (between March 2016 and August 2016). The trace contains Netflow information of traffic captured in the border routers of a tier-3 ISP. The total number of flows in the dataset is above 16,900M, and the number of external IPs observed in the trace is higher than 600M, corresponding to around 10M different (sub)networks. Data are parsed into M-dimensional vectors (observations) representing time intervals of 1 minute. In particular, we defined a set of M=138 network-related features, corresponding to 11 different *Netflow* variables.

There are several reasons for the selection of this dataset. First, the data are recent and, thus, reflect the real trends in current network traffic. Second, the trace contains anomalies that allow us to evaluate if the presented visual methodology is efficient to uncover them. Actually, there are two types of security incidents in the dataset: one group of anomalies were generated by the researchers in a controlled way, while the others are anomalies that appear in the real background traffic of the network. One final reason for choosing this

[2] Available to download at: https: //nesg.ugr.es/nesg-ugr16/

dataset is the high rate of traffic in the trace, which makes identifying and analyzing anomalies similar to finding a needle in a haystack.

5.2 Case Study

As the methodology implies an interactive visualization, we first describe the anomaly that is selected for our case analysis, and then we explain step by step the different phases in the visual analysis workflow and their interactions. For a demonstration of the system in practice and for a better understanding of the example of use of iGPCA we refer to the supplementary video[3].

The anomaly selected from the dataset takes place in the short interval from 04:10:00-08/01/16 to 04:14-08/01/16, though interval 04:00-08/01/16 to 04:23:59-08/01/16 has been selected to compute the GPCA model. The anomaly is taken from the set of anomalies that are not generated in a controlled way, so we test our methodology with anomalies that happen in real background traffic. During this period in the trace we first made a manual analysis to find the ground truth of the anomaly, that we explain now.

The anomaly is signaled as an increase of ACK packets and a big amount of very short UDP connections (small number of bytes). In Fig. 7, we can see the different bursts of traffic connections that happen during the anomaly interval. Inspecting the Netflow records for this time period we can find a single IP from Germany creating 867,405 connections from only four origin ports (5061, 5062, 5066 and 5068). The destinations are 4,097 different hosts distributed in 16 different subnets (/24 mask). Depending on the source port of the connection, each victim host is scanned through a specific range of 60 ports (*e.g.*, from source port 5068, ports 5000-5059 were scanned). Due to this pattern of connections, we conclude that it seems to be a malware driven scanning for a specific vulnerability.

Following, we walk the reader trough the use of the tool –according to the visual interactive methodology described in Section 4– for finding and diagnosing the previously mentioned anomaly as example of use.

5.2.1 Data Inspection

In its first view, iGPCA shows the whole dataset. Fig. 3 shows all the data composed by observations (rows) and features (columns). The top part of the figure depicts values of the variables per observation while the bottom one shows the numeric values. The user can interact with the parallel coordinates plot by selecting a range of observations that will be filtered in the data table just below it. From this visualization we can do a preliminary analysis where values different from the rest might suggest that the involved observations are prone to be anomalous.

5.2.2 PCA analysis

As described in Section 3, GPCA depends on the number of PCs used to build the initial PCA model. Fig. 4 shows how the analyst can adjust and select the number of PCs interactively, obtaining an immediate feedback on how the residual variance varies. Changing the number of PCs also changes the underlying model so the user can see the effects on the D-st and Q-st statistics. These charts enable the analyst to leave out data outliers (by vertically moving the blue line below the height of the bar) and determine potential anomalous events that correspond with observations passing the control limits. Note that leaving out observations in turn change the model, yielding full interaction of any user action with the modeling. The figure shows that we decided to use 15 PCs from the inspection of the model's captured variance. The bars corresponding to observations number 236-238 and 240 in the D-st chart, above the control limit, suggest that something unusual is happening in them. We will inspect these observations in more detail.

5.2.3 GPCA analysis

Once the number of PCs is selected, we are ready to start the GPCA analysis. To that end, Fig. 6 depicts the MEDA graph (heatmap on the left) that shows the variables relationships. In the same graph we can see the groups of variables obtained by GIA (colored rectangles). The user can adjust the *gamma* (γ) and *groups threshold* sliders to change the number of groups of variables obtained by GIA and show only the groups with a certain intra-correlation, respectively. The MEDA graph is necessarily connected with the previous ones, described in the PCA analysis section, because model modifications affect the number of groups and relationship among the variables. Consequently, this graph is linked and is refreshed when the user changes the model.

In addition, the security analyst can modify the subjective relevance of each variable by changing their corresponding weight in the interactive graph shown in Fig. 5. These weights are initially set up to 1 meaning that all of them are equally relevant. Note that changing the weight does not change the inherent model, as this weight should not be confused with the dispersion of the variables that do has an influence on PCA and GPCA. These weights are only used by the analyst to put more emphasis on a variable –thus embedding the available expert knowledge– in the MEDA visualization using a color code. For instance, the group of maximum weight is shown in dark orange in Fig. 6. Clicking on this group, the corresponding scores and loadings charts by GPCA are shown. It can be easily seen on these graphs that the anomalous observations are below the limits. In fact, the anomaly we are interested in corresponds to the observations 236-238 and 240 (around 4 AM in the graph), which in turn are part of the intervals of the anomaly under evaluation.

Once the anomaly is detected, a more detailed exploration should be performed to obtain some relevant and useful details for subsequent response or mitigation actions.

5.2.4 De-parsing

De-parsing refers to the procedure of, first, retrieving the involved raw data that motivated the anomaly and, second, presenting these data for a more detailed inspection.

When user clicks on an anomalous score, the de-parsing procedure is launched. In the current case of study, observations 236-238 and 240, corresponding to 04:10-08/01/16, 04:11-08/01/16, 04:12-08/01/16 and 04:14-08/01/16 timestamps, can be used as triggers of the de-parsing procedure. Once the timestamps are identified, it is possible to retrieve the associated data from the corresponding intervals. Afterward, the data will be visualized in a proper way (depending on the type of the data source). In our case, hive plots [15] can successfully convey the relationship among network flows obtained from Netflow data sources.

Fig. 8 shows the de-parsing visualization of the selected anomaly time interval. In our case such time interval goes from 04:14-08/01/16 to 04:15-08/01/16, as can be seen in the bar chart through the temporal evolution of the number of connections in this period of time (this in turn corresponds with the fifth traffic burst shown in Fig. 7). For a better understanding of this evolution, animated hive plots are used, thus enabling the discovery of patterns that, in turn, can be inspected in detail. This approach allows the visualization of a simulation of the recorded traffic in such a way that the visual identification of anomalies can easily be discovered in a preattentive manner [21]. The hive plot represents the set of connections corresponding to the previous time interval. In the figure, the different IPs observed in the connections are represented as circles in the axes, such that the size of the circle depends on the number of connections in which that IP is involved. In this case, we have split the representation in 3 different axes, where we have selected the most active IPs in the ISP range (x-axis), other IPs in Spain (y-axis) and the rest of the world IPs (z-axis). The lines connecting IPs from different or same axes represent the connections. The color of

[3] Available to download at: https: //nesg.ugr.es/nesg-ugr16/VizSec17.mp4

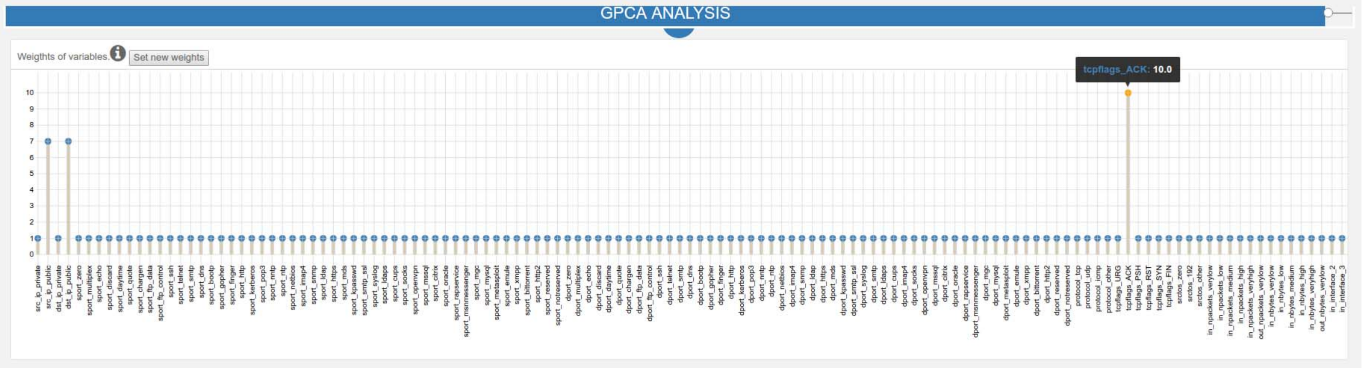


Figure 5: GPCA analysis. Different variable weights to set up the color in the GIA groups where it belongs to.

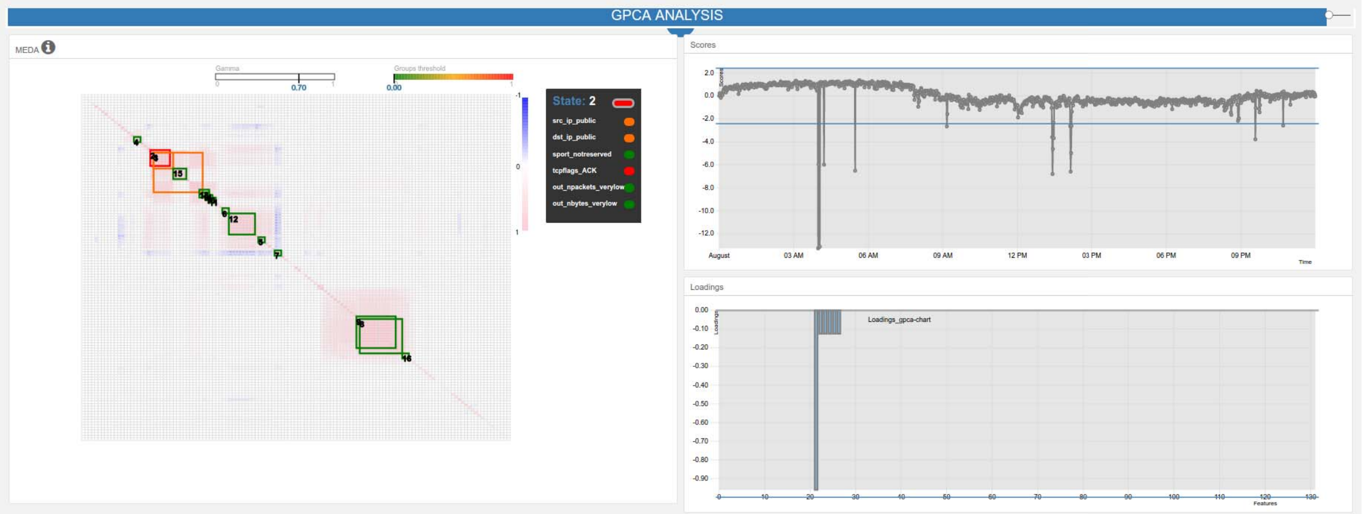
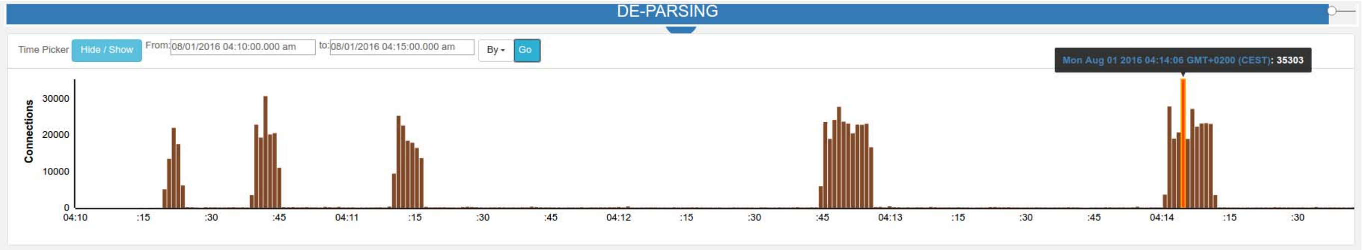


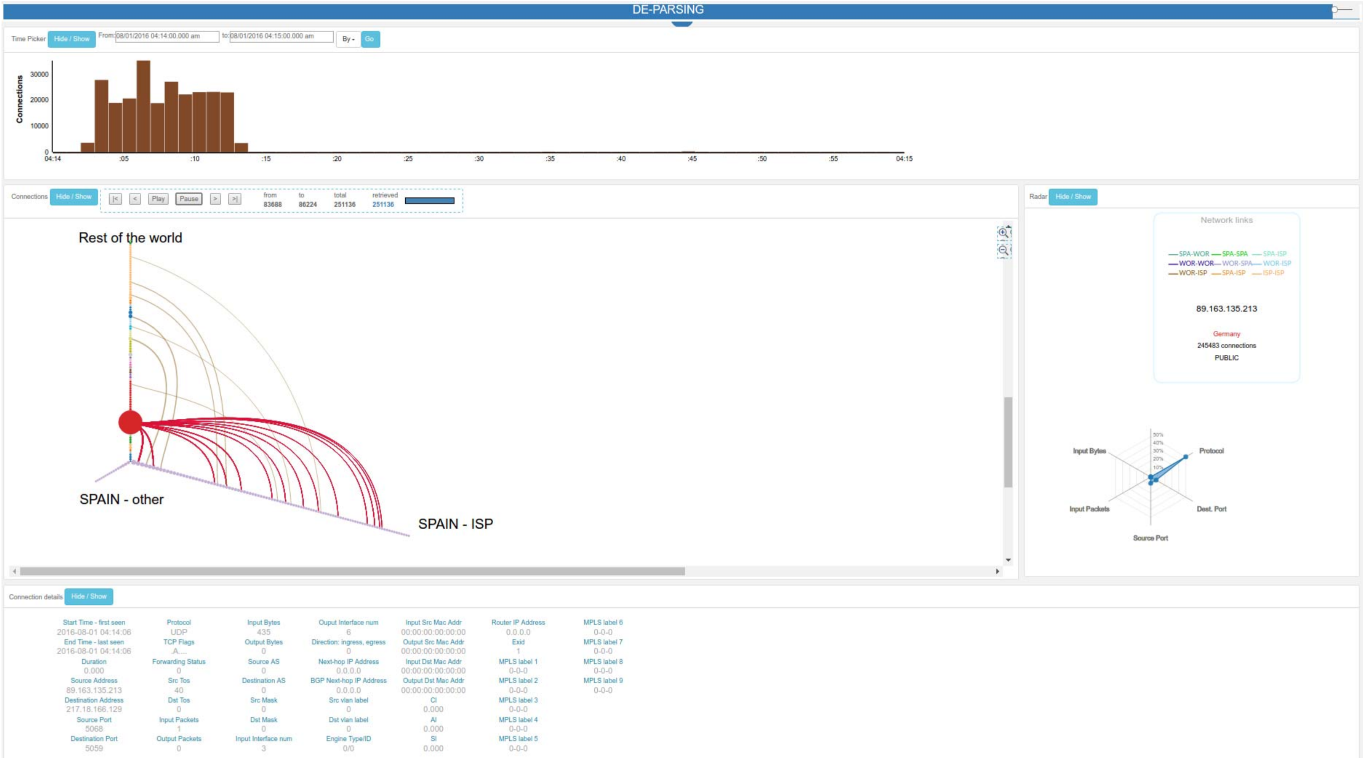
Figure 6: GPCA analysis. On the left, MEDA graph and GIA groups are shown while the scores and loadings GPCA plots of the group 2 are depicted on the right.



Time Picker From: 08/01/2016 04:10:00.000 am to: 08/01/2016 04:15:00.000 am

Mon Aug 01 2016 04:14:06 GMT+0200 (CEST): 35303

Figure 7: De-parsing. Time bar chart for the complete anomaly time interval. Traffic bursts suggest the occurrence of an anomalous behavior.



From: 08/01/2016 04:14:00.000 am to: 08/01/2016 04:15:00.000 am

from 83688 to 86224 total 251136 retrieved 251136

89.163.135.213

Germany

245483 connections

PUBLIC

Start Time - first seen: 2016-08-01 04:14:06
End Time - last seen: 2016-08-01 04:14:06
Duration: 0.000
Source Address: 89.163.135.213
Destination Address: 217.18.166.129
Source Port: 5068
Destination Port: 5059
Protocol: UDP
TCP Flags: .A....
Forwarding Status: 0
Src Tos: 40
Dst Tos: 0
Input Packets: 1
Output Packets: 0
Input Bytes: 435
Output Bytes: 0
Source AS: 0
Destination AS: 0
Src Mask: 0
Dst Mask: 0
Input Interface num: 3
Ouput Interface num: 6
Direction: ingress, egress: 0
Next-hop IP Address: 0.0.0.0
BGP Next-hop IP Address: 0.0.0.0
Src vlan label: 0
Dst vlan label: 0
Engine Type/ID: 0/0
Input Src Mac Addr: 00:00:00:00:00:00
Output Src Mac Addr: 00:00:00:00:00:00
Input Dst Mac Addr: 00:00:00:00:00:00
Output Dst Mac Addr: 00:00:00:00:00:00
CI: 0.000
AI: 0.000
SI: 0.000
Router IP Address: 0.0.0.0
Exid: 1
MPLS label 1: 0-0-0
MPLS label 2: 0-0-0
MPLS label 3: 0-0-0
MPLS label 4: 0-0-0
MPLS label 5: 0-0-0
MPLS label 6: 0-0-0
MPLS label 7: 0-0-0
MPLS label 8: 0-0-0
MPLS label 9: 0-0-0

Figure 8: De-parsing. Hive plot showing the connections for the selected anomaly interval. A more detailed connection information is shown at the bottom and on the right by hovering over the selected connection. On the right a radar graph showing the percentage of certain variables is shown.

lines indicates the direction of the connections. On the other hand, the radar graph (on the right of Fig. 8) conveys, for each selected variable, the percentage of the maximum value accounted for among the flows in the selected time interval.

From the hive plot in Fig. 8 we can clearly identify a pattern related to the fact that abnormal traffic is being generated. First, a single IP is generating most of the connections (see the big circle in the figure). Second, we see that the same source port is used (this can be seen in the repeated radar pattern when hovering over the lines). Both the radar, and the details panel at the bottom of the figure indicate that almost all connections in the interval have the same origin port. The user can interact with the tool through selecting different time intervals which change the connections involved and the hive plot. Additionally, a detailed information of the selected connections can be seen under the radar plot.

In order to show the differences between anomalous and normal traffic, Fig. 9 illustrates the visual pattern conveyed by the tool for a normal traffic behavior time interval (see this period in Fig. 7 from third to fourth bursts). It can be seen by inspecting the hive plot, that the connections observed exhibit a pattern that is very widespread among the different IPs. The variety of colors in the lines is also an evidence that there are no common patterns (the analyst can hover over the lines of a particular color, and the radar plot will not show a similar pattern). Moreover, the evolution of the connections illustrated in the bar chart does not show any specific or differentiated pattern as well as as the number of connections is much lower in comparison with the anomalous traffic burst.

Finally, for exploratory purposes, at the top of the hive plot the analyst can use a player panel to watch an animation of the connections evolution during the selected time interval. The users can play, pause, go backward or forward step by step. Also, an attack evolution study may provide a predictive analysis of its future behavior that could be used to apply the corresponding response actions to mitigate or prevent the attack.

6 Conclusion

In this paper, we introduce a new visualization tool for network-wide intrusion detection based on multivariate anomaly detection with Principal Component Analysis (PCA) and Group-wise PCA (GPCA). We use a real case study to illustrate the capability of the tool to unveil the complex mixture of information that can be found in network traffic data and identify and diagnose anomalies in it. In this example, the tool leads to the most relevant anomalies in a way that the user does not need any training in multivariate analysis.

We propose a workflow for the interactive visual analysis that hides the mathematical complexity of the models behind. Furthermore, the expertise and human capabilities of the security analyst can be exploited in the workflow and contribute to promptly and more accurate detection and diagnosis of abnormal network behavior.

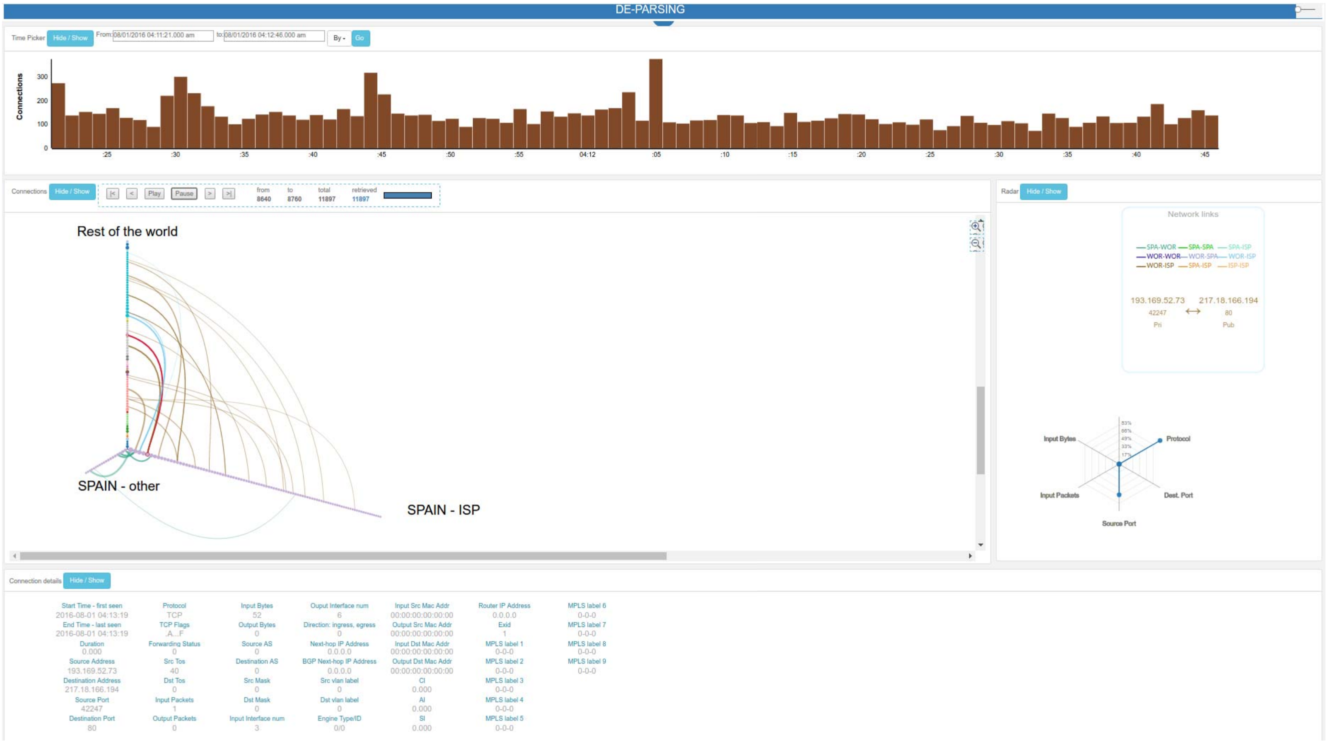
Future work is mainly aimed at connecting the user interface with the features, so that the user can modify or re-define the features that are considered in the detection system to explore the data in more detail. Other ancillary visualizations will be studied for other types of sources. We are also working on the comparison between GPCA and other machine learning methods for anomaly detection in cibersecurity. Finally, we plan to conduct formal usability studies in order to discover and overcome usability issues that may arise.

Acknowledgments

This work has been partially supported by Spanish Government-MINECO (Ministerio de Economa y Competitividad) and FEDER funds, through project TIN2014-60346-R.

References

[1] M. Angelini, N. Prigent, and G. Santucci. PERCIVAL: proactive and reactive attack and response assessment for cyber incidents using visual



From: 08/01/2016 04:11:21.000 am to: 08/01/2016 04:12:46.000 am

from	to	total	retrieved
8640	8760	11897	11897

Start Time - first seen 2016-08-01 04:13:19	Protocol TCP	Input Bytes 52	Ouput Interface num 6	Input Src Mac Addr 00:00:00:00:00:00	Router IP Address 0.0.0.0	MPLS label 6 0-0-0
End Time - last seen 2016-08-01 04:13:19	TCP Flags .A...F	Output Bytes 0	Direction: ingress, egress 0	Output Src Mac Addr 00:00:00:00:00:00	Exid 1	MPLS label 7 0-0-0
Duration 0.000	Forwarding Status 0	Source AS 0	Next-hop IP Address 0.0.0.0	Input Dst Mac Addr 00:00:00:00:00:00	MPLS label 1 0-0-0	MPLS label 8 0-0-0
Source Address 193.169.52.73	Src Tos 40	Destination AS 0	BGP Next-hop IP Address 0.0.0.0	Output Dst Mac Addr 00:00:00:00:00:00	MPLS label 2 0-0-0	MPLS label 9 0-0-0
Destination Address 217.18.166.194	Dst Tos 0	Src Mask 0	Src vlan label 0	CI 0.000	MPLS label 3 0-0-0	
Source Port 42247	Input Packets 1	Dst Mask 0	Dst vlan label 0	AI 0.000	MPLS label 4 0-0-0	
Destination Port 80	Output Packets 0	Input Interface num 3	Engine Type/ID 0/0	SI 0.000	MPLS label 5 0-0-0	

Figure 9: De-parsing. Hive plot showing the connections found from the normal traffic interval between burst third and fourth in Fig. 7

analytics. In *2015 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, 2015. doi: 10.1109/VIZSEC.2015.7312764

[2] J. Camacho. Missing-data theory in the context of exploratory data analysis. *Chemometrics and Intelligent Laboratory Systems*, 103:8–18, 2010.

[3] J. Camacho, G. Maciá-Fernández, J. Díaz-Verdejo, and P. García-Teodoro. Tackling the big data 4 vs for anomaly detection. *Proceedings - IEEE INFOCOM*, (1):500–505, 2014. doi: 10.1109/INFCOMW.2014.6849282

[4] J. Camacho, A. Pérez-Villegas, P. García-Teodoro, and G. Maciá-Fernndez. Pca-based multivariate statistical network monitoring for anomaly detection. *Computers & Security*, 59:118–137, 2016.

[5] J. Camacho, R. A. Rodríguez-Gómez, and E. Saccenti. Group-wise Principal Component Analysis for Exploratory Data Analysis. *Journal of Computational and Graphical Statistics*, 2017.

[6] N. Cao, C. Shi, S. Lin, J. Lu, Y. R. Lin, and C. Y. Lin. Targetvue: Visual analysis of anomalous user behaviors in online communication systems. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):280–289, Jan 2016. doi: 10.1109/TVCG.2015.2467196

[7] B. C. M. Cappers and J. J. v. Wijk. Understanding the context of network traffic alerts. In *2016 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, Oct. 2016. doi: 10.1109/VIZSEC.2016.7739579

[8] S. Engle and S. Whalen. Visualizing distributed memory computations with hive plots. In *Proceedings of the Ninth International Symposium on Visualization for Cyber Security*, pp. 56–63. ACM, 2012.

[9] L. Hao, C. G. Healey, and S. E. Hutchinson. Ensemble visualization for cyber situation awareness of network security data. In *2015 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, 2015. doi: 10.1109/VIZSEC.2015.7312766

[10] N. A. Huynh, W. K. Ng, A. Ulmer, and J. Kohlhammer. Uncovering periodic network signals of cyber attacks. In *2016 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pp. 1–8, Oct. 2016. doi: 10.1109/VIZSEC.2016.7739581

[11] I. Jolliffe. *Principal component analysis*. Springer Verlag Inc., EEUU, 2002.

[12] I. Jolliffe, N. Trendafilov, and M. Uddin. A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics*, 2003. doi: 10.1198/1061860032148

[13] A. Kanaoka and E. Okamoto. Multivariate statistical analysis of network traffic for intrusion detection. In *14th. International Workshop on Database and Expert Systems Applications (DEXA'03)*, pp. 1–5, 2003.

[14] D. A. Keim, F. Mansmann, and J. Thomas. Visual analytics: how much visualization and how much analytics? *ACM SIGKDD Explorations Newsletter*, 11(2):5–8, 2010.

[15] M. Krzywinski, I. Birol, S. J. Jones, and M. A. Marra. Hive plotsrational approach to visualizing networks. *Briefings in bioinformatics*, 13(5):627–644, 2011.

[16] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft. Structural analysis of network traffic flows. *ACM SIGMETRICS/Performance*, pp. 61–72, 2004. doi: 10.1145/1012888.1005697

[17] P. Laskov, K. Rieck, C. Schäfer, and K.-R. Müller. Visualization of anomaly detection using prediction sensitivity. In *Sicherheit*, vol. 2, pp. 197–208, 2005.

[18] H. Ringberg, A. Soule, J. Rexford, and C. Diot. Sensitivity of PCA for traffic anomaly detection. *ACM SIGMETRICS Performance Evaluation Review*, 35(1):109, jun 2007. doi: 10.1145/1269899.1254895

[19] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim. Visual interaction with dimensionality reduction: A structured literature analysis. *IEEE transactions on visualization and computer graphics*, 23(1):241–250, 2017.

[20] M. Shyu, S. Chen, K. Sarinnapakorn, and L. Chang. A novel anomaly detection scheme based on principal component classifier. In *IEEE Foundations and New Directions of Data Mining Workshop (ICDM'03)*, pp. 171–179, 2003.

[21] A. Treisman. Preattentive processing in vision. *Computer vision, graphics, and image processing*, 31(2):156–177, 1985.

[22] J. Xin, J. Dickerson, and J. A. Dickerson. Fuzzy feature extraction and visualization for intrusion detection. In *Fuzzy Systems, 2003. FUZZ'03. The 12th IEEE International Conference on*, vol. 2, pp. 1249–1254. IEEE, 2003.