Characterization of Attackers' Activities in Honeypot Traffic Using Principal Component Analysis

S. Almotairi, A. Clark, G. Mohay, and J. Zimmermann
Information Security Institute, Queensland University of Technology
Brisbane, Queensland, Australia
{s.almotairi,a.clark,g.mohay,j.zimmermann}@isi.qut.edu.au

Abstract

Monitoring Internet traffic is critical in order to acquire a good understanding of threats and in designing efficient security systems. While honeypots are flexible security tools for gathering intelligence of Internet attacks, traffic collected by honeypots is of high dimensionality that makes it difficult to characterize. In this paper, we propose the use of principal component analysis, a multivariate analysis technique, for characterizing honeypot traffic and separating latent groups of activities. In addition, we show the usefulness of principal component plots in visualizing the interrelationships between the detected groups of activities and in finding outliers. This work is demonstrated through the use of low interaction honeypot traffic data from the Leurrè.com project, a world wide deployment of low interaction honeypots.

1. Introduction

Characterizing attackers' activities present in honeypot traffic data can be challenging due to the high dimensionality of the data and the amount of traffic collected. The high amount of background noise, such as scans and backscatter, add to the challenge by hiding interesting abnormal activities that require immediate attention from security personnel. Detecting these outlying activities can potentially be of high value and give early signs of the discovery of new vulnerabilities or breakouts of new automated malicious codes, such as worms. In this work, we propose the use of principal component analysis (PCA), in the characterization of attacker activities present in low-interaction honeypot traffic data. While PCA has been used to characterize network traffic in the past, as far as we are aware this is the first time it has been used to characterize honeypot traffic.

The use of PCA in this study is motivated by the popularity of PCA as an exploratory technique [1] that is easy to implement and requires less computational power than other linear methods, such as projection pursuit, and produces results that are easy to interpret. In this paper, the effectiveness of PCA in detecting the structures of attackers' activities in honeypot traffic is demonstrated through the characterization of the attackers' activities into dominant groups, visualization of some the interrelationships between the extracted groups, and the ability to detect different types of outliers. Consequently, characterizing honeypot traffic will improve our understanding of attacker behaviors, optimization of honeypot design, and the identification of interesting activities.

Pouget et al. [2] applied clustering techniques to low-interaction honeypots, with the port sequence of a 'large session' as a main clustering feature, to group traffic that shares similar activity fingerprints, or attack tools. This study uses data from the same project, but with a different time span. In this paper, the raw honeypot data is processed based on a well-known traffic flow technique [3] without the notion of sessions used in the Leurré.com project: large and tiny sessions. Moreover, our aim is to characterize attackers' activities using principal component analysis, while their study was to characterize the root causes of attacks using association rules. In our previous work [4], we have used the cliquing algorithm to extract different groups of activities, that exhibit similarities, from low-interaction honeypot clusters. Packet inter-arrival time distributions were used as the main clustering feature with the objective of identifying the repeated use of attack tools and attack processes. The cliquing algorithm was applied to pre-clustered honeypot data for extracting refined activities. This work differs in that we apply PCA directly to the honeypot data.

Lakhina et al. [5] use principal component analysis to decompose the structure of Origin-Destination flows, from two backbone networks, into three main constituents, namely periodic trends, bursts and noise. Labib et al. [6] proposed a method of detecting two

978-0-7695-3354-4/08 $25.00 © 2008 IEEE
DOI 10.1109/NPC.2008.82

IEEE computer society

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

classes of attacks, Denial-of-Service and Network-Probe, present in the 1998 DARPA data set; they utilized PCA in reducing the dimensionality of the traffic vector and identifying attacks. Our work differs by applying PCA to data from low-interaction honeypots for finding dominant groups of attacker's activities and finding outliers.

The structure of the paper is as follows. Section 2 provides a brief summary of principal component analysis. The dataset used in this study and the preprocessing is described in Section 3. Section 4, details the process of applying PCA to the preprocessed honeypot data. Interpretations of the principal components (PCs) are presented in Section 5 while the interrelationships between the components are discussed in Section 6. Detection of outliers based on plots of the PCs is discussed in Section 7. Finally, the results are discussed and the paper is concluded in Section 8.

2. Principal component analysis

Principal component analysis (PCA) is a multivariate statistical technique that has been widely used in multi-disciplinary research areas such as internet traffic analysis, economics, image processing, and genetics, to name only a few. PCA is mainly used to reduce the dimensionality of a data set into a few uncorrelated variables, principal components (PCs), which retain most of the variation in the original data [1, 7-9]. The resulting principal components are a linear combination of the original variables, are orthogonal, and ordered with the first principal component having the largest variance. Although the number of resulting principal components is equal to the number of original variables, much of the variance in the original set of p variables can be retained by the first k PCs, where $k< p$. Thus, the original p variables can be replaced by the new k principal components.

Given the p-dimensional random variables $X=(X_1, ..,X_p)^T$ with a sample mean $\overline{X}_i$ and a sample covariance matrix $\sum$, we seek to find a lower dimension vector $A=(A_1, ..,A_k)^T$ of $\sum$ that has the maximum variance of the original data with all the Eigenvalues being greater than zero. Thus, the first linear function Z_1 of X having maximum variance:

$$\begin{aligned} Z_1 &= a_{11}X_1 + a_{12}X_2 + a_{13}X_3 + ...+ a_{1p}X_p \\ Z_2 &= a_{21}X_1 + a_{22}X_2 + a_{23}X_3 + ...+ a_{2p}X_p \\ &\vdots \\ Z_k &= a_{k1}X_1 + a_{k2}X_2 + a_{k3}X_3 + ...+ a_{kp}X_p \end{aligned} \quad (1)$$

The second linear function Z_2 is uncorrelated with Z_1 and having the second largest variance and so on until the kth function, Z_k, is found which is uncorrelated with $Z_1,\ldots,Z_{k-1}$. For the interested reader, full discussion of principal component analysis can be found in [1, 7].

Principal component analysis has the following advantages:

- It does not require any distributional assumptions and can be used with many types of data.
- The extracted principal components are uncorrelated.
- The first few principal components retain most of the variation in the original data.

In this work, PCA is utilized to search for groups of activities found in the honeypot traffic, without assumptions about these groups or interrelationships between them. In addition, some of our objectives in this study are to explore the usefulness of PCA in visualizing honeypot traffic, finding interrelationships between groups of activities, and detecting outliers.

3. Dataset and preprocessing

In this section we describe the dataset used in this study and the preprocessing that has been applied to the data.

3.1. Dataset

The honeypot traffic data used in this analysis comes from the Leurrè.com project [8]. The Leurrè.com project was launched in 2004 for collecting malicious traffic using globally distributed, identical honeypot environments; currently 50 platforms are deployed in 30 different countries. The Leurré.com honeypot sensor is based on the open source low-interaction honeyd [9]. Each sensor runs on a single host and emulates three operating systems at the same time (on different IP addresses): Windows 2003 Professional; Windows 2003 Server; and Linux Red Hat. For the purpose of this study, only one low-interaction honeypot sensor's data is used due to the availability of log files. Traffic data for the period of September 15 until November 31, 2007 for two of the honeypot environments were included, namely Windows 2003 Professional and Windows 2003 Server. Both environments are identical in terms of open ports, TCP and UDP. The traffic traces consist of 839663 packets which were the result of attacks from over 5400 different IP addresses.

3.2. Preprocessing

Before applying the PCA to the traffic data, the following steps were performed to process the raw traffic data. First, raw tcpdump [10] files of daily honeypot

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

data were collected and merged into a single traffic file. Then packets were grouped together (according to the notation of flow [3]) into *basic flows*; our basic flow conforms to the standard definition of an IP flow, namely the five-tuple containing the: source IP address, destination IP address, source port, destination port, and protocol type. If a packet differs from another packet by any key field, it is considered to belong to another flow [11, 12]. For the purpose of this study, we set the timeout of basic flows to a maximum of five minutes. The five-minute timeout parameter was selected based on our experiments and the nature of low-interaction honeypots where the majority of flows are less than 300 seconds; a higher value of time out has little influence in the final results. The second step was to group the basic flows again into what we call *activity flow*, where the newly generated flows were combined based upon the source IP address of the attacker with a maximum of sixty minutes inter-arrival time between basic flows. Finally, the data is filtered to remove internet noise, such as backscatter [13].

3.3. Candidate feature selection

As described above, this study deals with two types of flows: basic and activity flows. The basic flow conforms to the definition of the standard flow [3, 12], which is a unidirectional series of IP packets with the same source IP and destination IP, source port and destination port and protocol number, with a timeout of five minutes. The second type of flow is the activity flow; an aggregation of basic flows based on the source IP address only with a timeout of sixty minutes, or inter-arrival times between two basic flows is less than sixty minutes.

Traffic features computed from the activity flows include: the total number of basic flows generated by individual IP and aggregated based on sixty minutes; total number of open TCP ports targeted; total number of distinct open TCP ports targeted; total number of open UDP ports targeted; total number of distinct open UDP ports targeted; total number of closed UDP ports targeted; total number of distinct closed UDP ports targeted; total number of ICMP flows; number of machines targeted per attack; total duration of basic flows; total number of source packets sent per IP; total number of source bytes sent; total source rates which is the sum of the source rates of all the basic flows, where a source rate is number of source packets in a basic flow divided by the duration of that flow; sum of the average packet size per basic flow; total activities as the summation of source and destination rates; and summation of inter-arrival times between basic flows. Table 1 lists the selected 18 variables and their descriptions.

Table 1. Variables used in the analysis

No.	Variable	Description
1	TF	Total number of basic flows
2	TCP_O	Total number of open TCP ports targeted
3	D_TCP_O	Total number of distinct open TCP ports targeted
4	TCP_C	Total number of closed TCP ports targeted
5	D_TCP_C	Total number of distinct closed TCP ports targeted
6	UDP_O	Total number of open UDP ports targeted
7	D_UDP_O	Total number of distinct open UDP ports targeted
8	UDP_C	Total number of closed UDP ports targeted
9	D_UDP_C	Total number of distinct closed UDP ports targeted
10	ICMP	Total number of ICMP flows
11	TM	Total number of machines targeted
12	DUR	Total duration of basic flows
13	SPKTS	Total number of source packets
14	SBYTES	Total number of source bytes
15	SRATE	Total of source rates of basic flows
16	AVG_PK_SIZE	Sum of average packet size
17	T_ACT	Total Activities
18	IAT	Total inter-arrival times between basic flows

These traffic features were selected as being representative of the behavior of the three transport layer protocols that are monitored by the honeypot, namely TCP, UDP and ICMP. We expect these variables to have some correlations between them which will be identified and removed during the principal component analysis.

4. PCA on the honeypot data set

Principal component analysis can be calculated using either the covariance matrix or the correlation matrix. However, PCs defined using the covariance matrix are very sensitive to the unit of measurement of variables. In addition, when the variance of the variables differs widely, which is the case for the honeypot data, the first few PCs will be dominated by variables with high variances, as they contribute little information to the structure of the data set. Moreover, one drawback of PCs on covariance matrices, with different units of measurement, is the difficulty in interpreting the PC scores. Thus, the use of correlation rather than the covariance matrix for deriving the PCs was preferred in our analysis.

To calculate the PCs from the correlation matrix, the p-dimensional vector $X=(X_1, ..,X_p)^T$ is standardized by:

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

$$C_i = \frac{X_i - \overline{X}_i}{\sqrt{s_i}} \quad (2)$$

for $i=1,.,p$, where $\overline{X}_i$ is the sample mean and s_i is the sample variance for X_i. Let R be the sample correlation matrix of C with Eigenvalue vector $l=(l_1,\ldots,l_p)$, then the principal component analysis

$$Z = A^T C \quad (3)$$

where $A= (A_1, ..,A_k)^T$ is the Eigenvector of R, with the first component equals to:

$$Z_1 = A_1^T C = a_{11}C_1 + a_{12}C_2 + a_{13}C_3 + ... + a_{1p}C_p \quad (4)$$

In PCA, components are in decreasing order where the most important component, which is listed first, has the highest variance value. Consequently, only the first few PCs are retained as they explain most the variance in the data.

The Kaisers' rule [14] for eliminating PCs with Eigenvalue less than one suggests retaining the first six components (see Table 2 for the Eigenvalues). The cumulative total variance of these components is 80% of the total variance of the original data.

However, the extracted *communalities* of variables, an amount of variance within each variable accounted for by the components, indicate that one of the variables, namely total number of distinct open TCP ports targeted, has a low extraction value. This suggests the inclusion of more components. After the inclusion of the seventh component, all the communalities are high, which indicates that the extracted components represent the variables well.

Table 2. Extracted principal components

Principal Component	Eigenvalues	% of Variance	Cumulative %
1	5.410	30.054	30.054
2	2.374	13.190	43.244
3	2.153	11.959	55.204
4	1.681	9.339	64.543
5	1.432	7.954	72.497
6	1.362	7.567	80.064
7	.959	5.329	85.393

The Scree plot of energy contributed by each component is summarized in Figure 1. This plot suggests that between six and seven components can be retained as the sharp drop occurs between the sixth and seventh component where the Eigenvalues are greater than or equal to 1. The sharp drop in the curve indicates a typical cutoff for selecting the correct number of components to be considered in the analysis.

All of the above supports our decision to retain seven components with the rest of the components being eliminated. Table 2 shows the accumulated percentages of the total variances of the 18 extracted components. The first seven components contribute over 85% of the total variance in the original data, which suggest that the extracted components are very representative of the data.

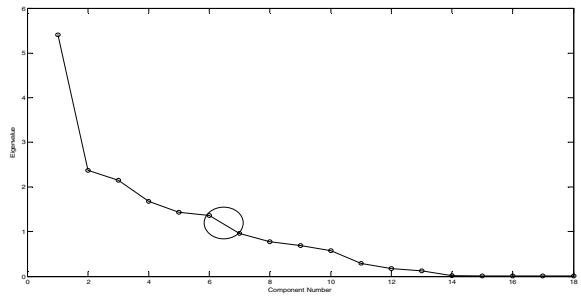

Figure 1. Scree plot of Eigenvalues

5. Interpretation of the results

As mentioned earlier, components are in decreasing order of importance, where components with larger variances are more important and give more information about the data. The components were rotated to simplify the analysis and make the interpretation easier.

Interpretation of the components is achieved by examining the loading of the variables for each component as variables with high loading are of high significance in the interpretation. Then, each PC's interpretation was validated by inspecting sample traffic against the original data. For this study, we have selected variables with a loading value over 0.6 as they are the most significant in the analysis.

Interpretation of the first seven PCs (PC1-PC7) for the honeypot data is summarized in Table 3. The first component (PC1) is highly correlated with the total number of basic flows, total number of TCP ports targeted, total duration of basic flows, total number of source packets, and total number of source bytes. The first component indicates high interactions between attackers and the honeypot on open ports and as the variance suggests, is the most important component. PC2 is highly correlated with closed TCP ports. This component suggests vertical and horizontal scan activities which focus on very specific ports. In PC3, activities target closed UDP ports and could be interpreted as spam, worm activities, or mis-configured servers. PC4 is related to repeated activities over a short period of time; this is explained by the high correlations between the total activities and variables in the first PC's variables, such as SPKTS, SBYTES, DUR, TCP_O, and TF. PC5 is represented by the total machines targeted and ICMP traffic. It can be inferred that these activities are of IPs sweeping the globe seeking live machines. PC6 represents activities that target open UDP ports. PC7 is a subset of the first component and represents

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

short attacks against specific open ports, mainly port 80, 139, and 445.

Table 3. Interpretations of the first 7 components

Component	Percentage of variation	Interpretation
1	30.054 %	Targeted attacks against open ports
2	13.190 %	Scans activities
3	11.959 %	Spam or miss-configuration
4	9.339 %	Repeated short activities
5	7.954 %	Detection activities
6	7.567 %	Targeted attacks against open UDP ports
7	5.329 %	Short attacks

The principal component analysis of the data shows that there are at least seven clusters of activities represented in the data. These clusters of activities can be separated and then PCA can be applied further to find new sub-clusters of activities and the process repeated.

6. Interrelations between components

Plots of PCs can serve two main purposes: to define the interrelations between components and to identify outliers. As discussed in Section 5 (interpretation of the results), the two components PC2 and PC5 represent two types of activities: TCP scanning and live machine detection respectively. The interrelationships between these two components are presented in Figure 2.

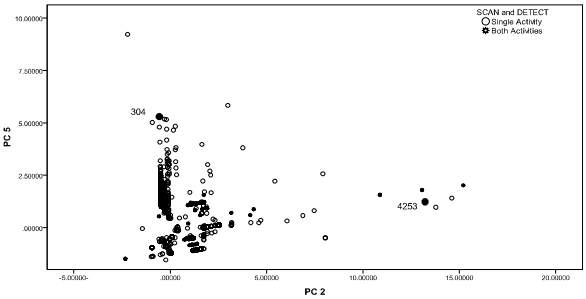

Figure 2. Scatter plot of TCP scan (PC2) vs. live machine detection (PC5)

The figure shows that there are at least two clusters of activities: detection with very few scans, on the upper left side of the figure along the PC5 axes; scans with very few machine detection activities at the bottom of the figure along the PC2 axes. Mixed activities, moderate rate of scans with moderate live machine detection activities are located in the middle part of the figure. Extreme activities of scanning and live machine detection activities are also visible as far points along both PC axes.

An example of scan only activities is observation 4253, which originated in Germany. The IP scanned all machines for closed port 2967 and then two weeks later scanned closed port 5904. Observation 304 is an example of the second type, live machine detector. The IP originated in Japan and was only involved in detection activities.

7. Identification of extreme activities in honeypot traffic

Detecting extreme activities in honeypot traffic is analogous to outlier detection in multivariate statistics. Outliers, in statistics, can be defined as observations that deviate largely from the rest of the data [15]. In honeypot traffic, outliers are extreme activities that are distanced from the p-dimensional hyperspace defined by the variables. The aim of detecting these extreme activities is to help in searching for the root causes of variations in patterns of the defined structures, and take measures to protect production networks against them. Outliers in honeypot might arise from many malicious network activities, such as releases of newly automated codes (worms) or discovery of new vulnerabilities; or even mis-configured servers.

One of the challenges in detecting outliers in high dimension data, such as honeypot traffic, is the difficulties of inspecting large numbers of variables in the data set simultaneously. In addition, inspecting each variable by itself or even inspecting plots of pairs of variables might not reveal any extreme behavior when the combination of multiple variables is considered an outlier. This study provides a preliminary investigation of utilizing principal component analysis in detecting extreme observations, through: graphical inspection of the first few and last few principal components' Plots; and the statistics of squares of the weighted principal component scores against the squared Mahalanobis distance.

Inspecting two and three dimensional scatter plots of the first few and last few PCs for detecting outlying observations was suggested by Gnanadesikan [16]. This was justified since, the first few PCs are good in detecting outliers that inflate the correlations, while the last few PCs are useful in detecting outliers that add unimportant dimensions to the data and are hard to distinguish from the original variables.

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

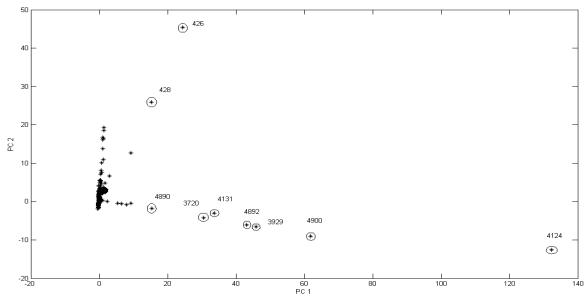

Figure 3. Scatter plot of the first two principal components

The scatter plot of the first two principal components is illustrated in Figure 3. These two components, PC1 and PC2, account for 43% of the total variance in the data. The first component has high loading values on multiple variables: total number of basic flows, total number of open TCP ports targeted, total durations of basic flows, total number of source packets, and total number of source bytes. The second component has two variables with high loadings on total number of closed TCP ports and distinct closed TCP ports targeted. Outlying observations (circled on the plot) can be spotted, in Figure 3, as points that have extreme values along the principal component axes near the edges, far from the body of data. Observations 4124, 4900, 3929, 4892, 4131, 3720, 426, 4890 and 428 are extreme on the first principal component (PC1) while observations 426 and 428 are extreme on the second principal component (PC2). Two observations are in common, namely: 428 and 426. These observations are possible outliers that require further investigation.

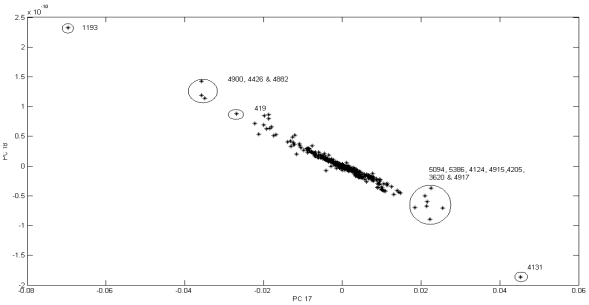

Figure 4. Scatter plot of the last two components

The scatter plot of the last two principal components, PC17 and PC18, which account for less than 1% of the total variance, is illustrated in Figure 4. There are two observations, 4131 and 1193 that are extreme for PC17 and PC18 near the edges of the graph.

Although, scatter plots of principal components are very useful for spotting outlying observations visually, automatic detection of outlying observations can be achieved through construction of a control ellipse. As the contours of constant probability for p-dimensional normal distribution are ellipsoids [7], the ellipsoid defined by random vectors $\boldsymbol{x}$ has the following characteristics:

Constant probability contour for the distribution of x is defined by

$$\sum_{k=1}^{p} \frac{z_{ik}^2}{l_k} = const \quad (5)$$

where Z_{ik} is the score of kth PC of ith observation and l_k is the kth Eigenvalue.

The ellipsoid is centered at the mean and its axes lie along the principal components where half the square root of the Eigenvalues ($l_1^{1/2}$, $l_2^{1/2}$, …, $l_p^{1/2}$) are the lengths of its semi-major and semi-minor axes.

The ellipsoid of p-dimensional space of x value satisfies

$$\sum_{k=1}^{p} \frac{z_k^2}{l_k} \le x_p^2(\propto) \quad (6)$$

where $x^2_p(\alpha)$ is the percentile of a chi-square distribution with p degrees of freedom.

Setting a threshold for detecting outlying observations based on x^2_p (α) requires the distribution of x to be multivariate normal. However, since we do not make any assumptions about the distribution of our data, the population ellipsoid

$$\sum_{k=1}^{p} \frac{z_{ik}^2}{l_k} = const \quad (7)$$

is still valid despite any normality assumption, but the ellipsoid loses its interpretation as contours of constant probability [1]. Based on the empirical distribution of the first two components PC1 and PC2, Equation 7 becomes:

$$\frac{z_{i1}^2}{l1} + \frac{z_{i2}^2}{l2} \le 5.8 \quad (8)$$

Figure 5 provides a zoom into Figure 3 omitting the very clear outliers and a sketch of the control ellipse for the first two principal components.

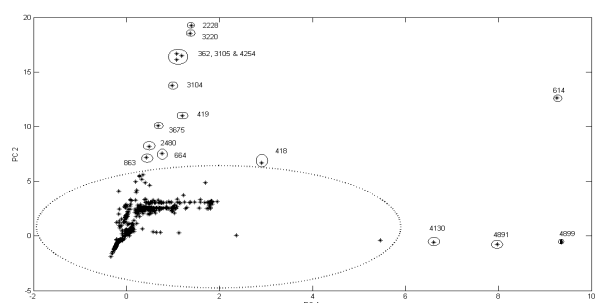

Figure 5. Ellipse of a constant distance based on the first two components

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

Jollife [1] discussed the uses of the sum of the squares of the weighted principal component scores of the last q principal components in detecting outliers that are hard to distinguish from the original variables, which is given by:

$$D_i = \sum_{k=p-q+1}^{p} \frac{z_{ik}^2}{l_k} \quad (9)$$

where $q<p$ and z_{ik} is the score of kth PC of ith observation and l_k is the kth Eigenvalue. When $q=p$, the equation represents the squared Mahalanobis distance of the ith observation from the mean of the data, which is given by:

$$M^2(x_i) = (x_i - \bar{x})^T S^{-1} (x_i - \bar{x}) \quad (10)$$

Figure 6 provides a scatter plot of the statistics D_i vs. (M^2-D_i) for detecting outliers that are different from the first p-q [17].

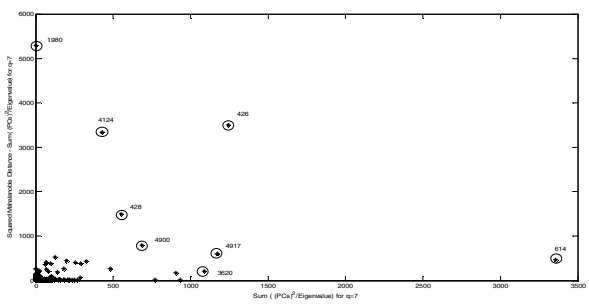

Figure 6. Scatter plot of the statistics Di vs. (M2-Di)

Finally, most of the detected outlying observations were identified by more than one statistic, but with different ordering. Table 4 lists the top 5 outliers ordered according to their significance (from high to low).

Table 4. Top five extreme observations

PC1	PC2	PC17	PC18	M2	D2	M2-D2
4124	426	4131	1193	1980	614	1980
4900	428	5094	4900	426	426	426
3929	2228	5386	4426	614	4917	4124
4892	3222	4124	4882	4124	3620	428
4131	362	4915	419	428	4131	4900

7.1. Evaluation of the detected outliers

To evaluate our methodology and judge the significance of the detected outliers, sample points of the outlying observations were manually inspected against the original data set to explain the reasons these points were selected as outliers.

Observation 4124, which is extreme in PC1, Figure 3, was a result of an attack from an IP in the USA targeting one machine on a single open TCP port, port 80. The attack started on Wednesday, 21 November 2007 at 06:18:44 GMT and ended on Friday, 23 November 2007 at 08:01:08 GMT. The attack generated over 150,062 packets. Observation 4124 was also extreme on M2, (M^2-D_i), and PC17.

Observation 2228 is extreme on both PC2 and (M^2-D_i) statistics. The attacking IP originated in China and lasted for less than 10 seconds. It was a combination of ICMPs and moderate scans of seven unusual closed TCP ports and one TCP open port, port 80. The attacker targeted all machines on the honeypot environment.

Observation 3105 is extreme in PC2 only. The IP address originated in China and was also responsible for another outlying observation, 3104. The IP address was involved in scanning activities. The first two attacks were detected as outliers, on 14/11/2007 and on 22/11/2007. However, the last attack that was launched on 22/11/2007 against open ports, which went undetected as it was not extreme on any dimension.

Observation 1193 is extreme on both PC17 and PC18. The IP address originated in Thailand and lasted for 40 minutes. It was mainly alternating connections to two open TCP ports 445 and 139 and one TCP closed port 9988. This observation has large value on TCP_O variable, moderate values on TF, DUR, and SPKTS variables, and low values on TC_C, and ICMP variables.

Observation 614 is an outlier on M^2 and D_i statistics. It was caused by an attack from an IP address that originated in Romania and lasted for 40 minutes. It was mainly connections to two open TCP ports (445 and 139) and one closed TCP port (9988). This observation shows similar behaviors to observation 1193 with the same duration, but with different IP from different country a week later.

Observation 1980 generated a large amount of UDP traffic (two packets every 30 minutes) against port 137. The attack took place between Thursday, 18 Oct 2007 and Wednesday, 24 Oct 2007 and has large UDP_O and IAT values. This observation is on the top lists of outliers on both M^2 and (M^2-D_i).

The main source of difference between the two statistics M2 and (M2-Di) was due to the value of q in Di statistics. More experiments are needed to select an appropriate value for the current data set. Moreover, setting a higher value for activity flow time-out, currently 60 minutes, would improve the detection of attacks that propagate slowly over an extended period, such as observations 1980 and 3105.

The detection of outliers serves as a first step for an online model for monitoring honeypots. As detailed

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.

before, these outlying observations could be eliminated from the data and PCs could be recomputed from a robust version of the correlation matrix [16].

8. Conclusions and future work

In this paper, we have proposed the use of principal component analysis (PCA) on the traffic flows of low-interaction honeypots. PCA proves to be very powerful tool in detecting the structure of attackers' activities and the decomposition of the traffic into seven dominant clusters. Moreover, scatter plots of the PCs are very efficient in looking at the interrelationships between components or groups of activities and in identifying any extreme traffic. Our experimental results on real traffic data show that principal component analysis could provide security administrators with a very simple and efficient way of summarizing honeypot traffic and monitoring activities.

Although our study was done off-line, it serves as a seed for a future real time model for monitoring honeypot traffic and providing security personnel prompt alerts of internet threats. Areas for future research include the implementation of the proposed model as a real time monitoring system of honeypots, experimentation with different data from different honeypot environments and different time periods.

9. Acknowledgements

The work described in this paper has been made possible by virtue of the Leurrè.com honeypot project led by Marc Dacier and that contribution is gratefully acknowledged.

10. References

[1] I.T.Jollif, *Principal Component Analysis*, 2nd ed., Springer, New York, 2002.
[2] F. Pouget and M. Dacier, "Honeypot-based Forensics", in *AusCERT Asia Pacific Information technology Security Conference*, Australia, 2004.
[3] K. C. Claffy, H.-W. Braun, and G. C. Polyzos, "A Parameterizable Methodology for Internet traffic Flow Profiling", *EEE Journal of Selected Areas in Communications,* vol. 13, October 1995, pp. 481-1494.
[4] S. Almotairi, A. Clark, M. Dacier, C. Leita, G. Mohay, V. H. Pham, O. Thonnard, and J. Zimmermann, "Extracting Inter-arrival Time Based Behaviour from Honeypot Traffic using Cliques", in *The 5th Australian Digital Forensics Conference*, Perth, Australia, 2007.
[5] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. Kolaczyk, and N. Taft, "Structural Analysis of Network Traffic Flows", in *ACM SIGMETRICS*, June 2004.
[6] K. Labib and V. R. Vemuri, "An Application of Principal Component Analysis to the Detection and Visualization of Computer Network Attacks", *Annals of Telecommunications,* 2005 Nov/Dec Issue, pp. 218-234.
[7] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, 6th ed., Prentice Hall, April, 2007.
[8] "The Leurrè.com Project Home Page", *http://www.leurrecom.org*, February 2008.
[9] N. Provos, "A Virtual Honeypot Framework", in *13th USENIX Security Symosium*, San Diego,CA, USA.
[10] "TCPDump Home Page", *http://www.tcpdump.org/*, November 2007.
[11] "Argus-Client 2.0.6", *http://www.qosient.com/argus/*, November 2007.
[12] "Cisco IOS NetFlow", *http://www.cisco.com/*, December 2007.
[13] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring Internet Denial-of-Service Activity", *ACM Transactions on Computer Systems* vol. 24, 2006, pp. 115-139.
[14] A. Rencher, *Methods of Multivariate Analysis*, 2nd ed., March 2002.
[15] V. Barnett and T. Lewis, *Outliers in Statistical Data*, 3rd ed., Wiley, April 1994.
[16] R. Gnanadesikan, *Methods for Statistical Data Analysis of Multivariate Observations*, 2nd ed., Wiley-Interscience Publication, New York, January 1997.
[17] B. Flury, *A First Course in Multivariate Statistics*, Springer, New York, 1997.

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on May 20, 2009 at 06:50 from IEEE Xplore. Restrictions apply.